IMPLEMENTACIÓN DE WEB SCRAPING COM PYTHON PARA PÁGINAS DEL SECTOR TURÍSTICO EN BUENAVENTURA

INVESTIGACIÓN: Análisis Estratégicos de los Servicios Tecnológicos Asociados al Sector Turísticos de Alojamiento y Gastronomía en Internet

---

Junio, 2023

Participante: *Kevin Arley Andrades Mondragon, CC:1148954816, Andradeskevin31@gmail.com*

Informe de semillero presentado como requisito para optar al título de Ingeniero de sistemas
Grupo de investigación Virtual
Línea de Investigación Tecnología, Sociedad y Región
Sublínea de Logística
Semillero de investigación Tics de Ecoturismo adscrito al proyecto macro "Análisis estratégicos de los servicios tecnológicos asociados al sector turísticos en el Distrito de Buenaventura"

Director: *Mg. Ferney Osma Mejía*

Programa de ingeniería de Sistemas
Universidad del Pacífico

RESUMEN

En el siguiente informe describe y muestra muy detalladamente cómo fue el proceso de elaboración de un prototipo de web scraping para la extracción de información en formato texto de sitios o páginas web de forma automatizada, todo con la intención de poder analizar qué tanto se conoce o que tanta información hay del turismo, la gastronomía, el transporte y el alojamiento en la ciudad de Buenaventura. Este prototipo se desarrolló en el lenguaje de programación Python, dado que este cuenta con un conjunto de librerías o bibliotecas que hacen el Web Scraping mucho más eficaz y fácil comparado con otros lenguajes de programación como PHP, JS, etc. Además, mostrar como fue el proceso de configuración de sitios web en la plataforma ya finalizada y por último, la finalidad del desarrollo de este prototipo se hace con el fin de entregar un producto 100% funcional al semillero liderado por el Mg. Ferney Hurtado Mejía.

Palabras Claves: *Web Scraping, Python y librerías*

ABSTRACT

In the following report, he describes and shows in great detail how the process of developing a web scraping prototype was for the extraction of information in text format from websites or web pages in an automated way, all with the intention of being able to analyze how much is known or that there is so much information about tourism, gastronomy, transportation and accommodation in the city of Buenaventura. This prototype was developed in the Python programming language, since it has a set of libraries that make Web Scraping much more efficient and easier compared to other programming languages such as PHP, JS, etc. In addition, to show how the process of configuring websites on the already completed platform was, and finally, the purpose of developing this prototype is to deliver a 100% functional product to the seedbed led by Mg. Ferney Hurtado Mejia.

Key Words: Web scraping, Python and libraries

INTRODUCCIÓN

Para comenzar, este escrito se centra en el proceso de creación de un prototipo para la extracción de información en forma de texto de diferentes páginas web extrayéndola de forma mecánica usando la herramienta o técnica de web scraping. Para la creación del prototipo se implementa en el lenguaje de programación Python, dado que este ofrece muchas ventajas por la facilidad para extraer información de forma rápida y precisa, es por esta razón que se escoge este lenguaje en comparación a otro tales como: PHP, JS y RUBI. Por otro lado, si se hiciera de forma manual sería un proceso muy lento y tedioso por la cantidad de información a buscar y esto no es lo más conveniente porque uno de los objetos del semillero es la optimización de tiempo y de los procesos de recolección de información. Además, el proceso de construcción de la página web es la unificación de todos los prototipos desarrollados por el equipo de programación del semillero.

ASPECTOS METODOLÓGICOS

La importancia del desarrollo de este proyecto es que permite recopilar información sobre turismo, alojamiento, gastronomía, eventos especiales en los destinos turísticos y transporte, de manera más eficiente y automatizada, porque de lo contrario sería muy tenue, costoso y abrumador ir de un sitio web a otro sitio web buscando información interesante y que tenga relación con lo que se está buscando.

1. Análisis de requerimientos

Tabla 1. Requerimiento funcional N°1

Número:	1
Nombre:	Insertar página
Descripción	Cualquier persona (programador o persona común) puede registrar sitios web, para extracción de información.
Entrada	URL de la página.
Procesos	Guardar URL dentro del documento .json
Salida	Documento .json actualizado con las URLs de las páginas web.

Fuente: Colaboración con grupo semillero (2022).

Tabla 2. Requerimiento funcional N°2

Número:	2
Nombre:	Ejecutar programa en Python
Descripción	Luego de tener el documento con las páginas web, el código debe poder extraer la información y mostrarla.
Entrada	Ejecutar el archivo json
Procesos	Se cargan las páginas web en el programa y este extrae la información y luego muestra los resultados en consola.
Salida	Muestra en consola el resultado final

Fuente: Colaboración con grupo semillero (2022).

2. Requerimientos no funcionales

Tabla 3. Requerimientos no funcionales

Requerimientos no funcionales	
Req.1	El sistema de ser eficiente en los tiempos de extracción
Req.2	El programa debe ser capaz de mantener grandes cantidades de información

Fuente: elaboración propia (2023)

3. Elaboración de prototipo(Desarrollo)

Se presentó un prototipo desarrollado en Python, ya que este cuenta con un conjunto de librerías que ayudan a facilitar la realización del web scraping.

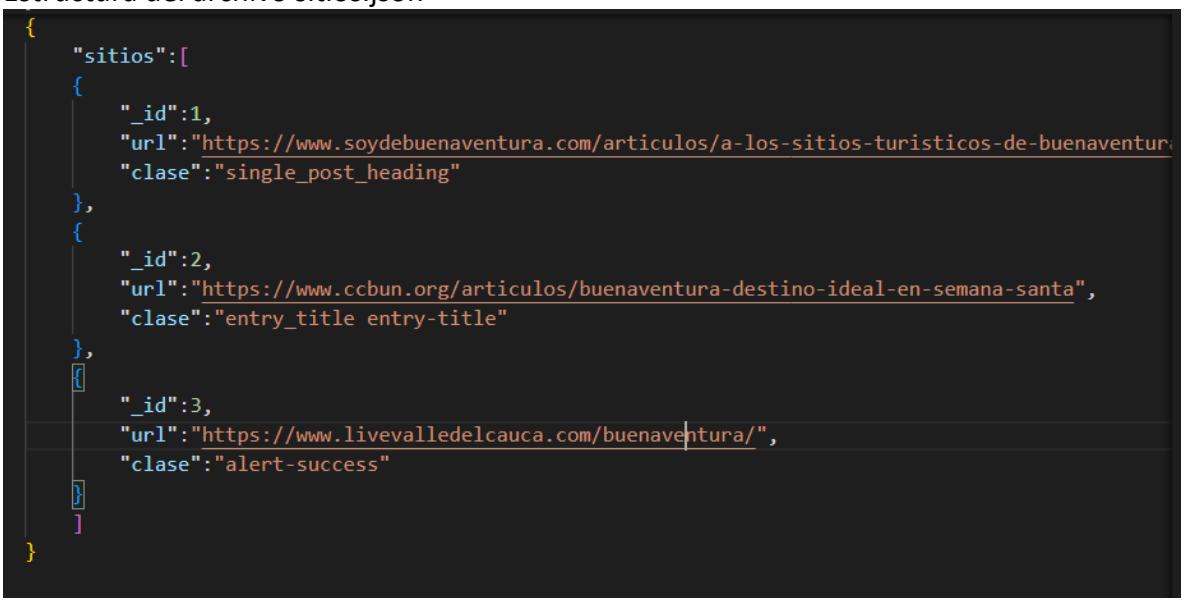
A continuación se detallará más la elaboración del prototipo:

Librerías usadas:

1. Requests: Esta librería me permite realizar las solicitudes HTTP.
2. BeautifulSoup: Está extraer información HTML Y XML, es decir esta permite extraer la información que estoy buscando de las diferentes páginas web.
3. Json: Este es un formato que me permite estructurar mejor la recolección de las páginas web.

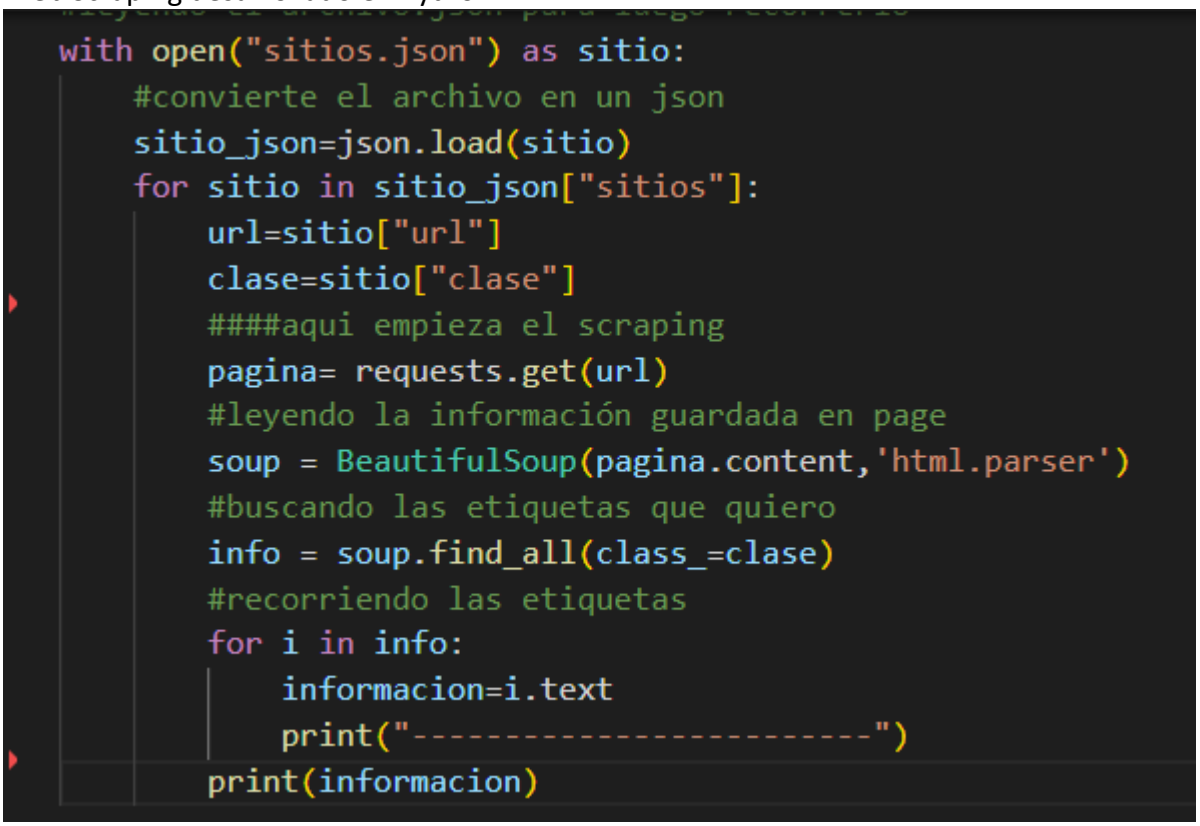
Para la construcción del web scraping se utilizó dos archivos uno para guardar los enlaces con su respectiva clase(sitios.json) y el tipo Python para ejecución del scraping, a continuación se podrá ver imágenes de los archivos y el enlace al repositorio el prototipo en GitHub.

Estructura del archivo sitios.json



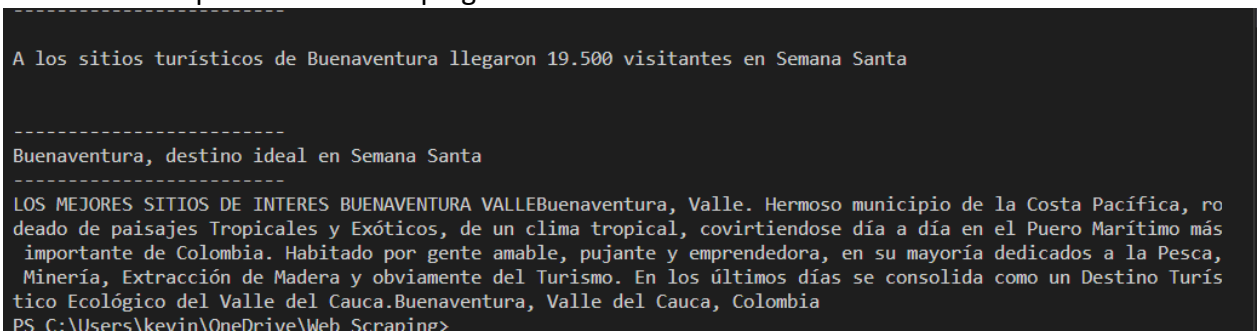
Fuente: elaboración propia.

Web Scraping desarrollado en Python.



Fuente: elaboración propia (2023).

Resultado de aplicar el web scraping



Fuente: elaboración propia(2023) .

1. Inicialmente se debe cargar el archivo sitios.json para obtener la información del sitio web.
2. Segundo, con la ayuda el método json.load() transforma el archivo a un objeto Python;
3. Recorrer el objeto y guardar las urls y la clase del objeto en las variables “url” y “clase”.
4. En la variable “pagina” se guarda el resultado de la petición.
5. Posteriormente de realizar la petición en la variable “soup” se guarda todo el contenido extraído del sitio web con la ayuda de la librería BeautifulSoup.
6. La variable “info” lo que hace es guardar solo la parte de la página que interesa extraer
7. Por último, lo que se hace el limpiar o quitarle las etiqueta a la información extraída para que el texto sea leíble.

4. Pruebas

Luego de ejecutar el prototipo, este realiza de manera exitosa la extracción de información de las páginas seleccionadas, además se puede especificar qué información de los sitios se quiere extraer y este lo hace con una mayor precisión, pero hay que tener en cuenta que entre más sitios web tenga el archivo sitios.json más se demora en terminar el scraping.

Adjunto el enlace del repositorio: https://github.com/kevinsiinho/web-scraping-python.git

5. Implementación

5.1. Consolidación del lenguaje

El web scraping se puede desarrollar en muchos lenguajes de programación, a continuación se presentará un cuadro de comparación con los 3 lenguajes que se tuvieron en cuenta y el por qué se eligió Python como la mejor opción:

Tabla N°4 .Comparación de lenguajes

Lenguaje	Conocimiento sobre el lenguaje	Ventajas	Desventajas
Python	Medio	Cuenta con muchas librerías para facilitar la extracción con muy pocas líneas de código	Puede ser más lento que otros lenguajes en determinadas acciones.
JS	Alto	Cuenta con muchas librerías y buena comunicación con otros lenguajes	De las 3 es la que cuenta con menos librerías.
PHP	Medio	Cuenta con muchas librerías y franmworks para extracción.	Su estructura es la más compleja de las 3 y necesita de más codificación para la extracción.

Fuente: elaboración propia (2023).

Luego de realizar la comparación el lenguaje más apropiado para lo que se quería hacer era Python, ya que este tiene un conjunto de librerías tales como:

- BeautifulSoup: esta realiza extracción de información de sitios web de manera muy sencilla.
- Scrapy: este es un framework específicamente para realizar web scraping.
- Selenium: se utiliza para la extracción de información cuando la información se genera de forma dinámica.

Además, es un lenguaje con una sintaxis muy fácil y clara de comprender, y no se necesita de un gran equipo para poder usarla de formas eficiente como lo afirma “Python es un lenguaje que se puede aprender en unas pocas semanas y que te permitirá hacer cosas útiles de inmediato” (Mathes, 2022).

6. Pruebas de funcionamiento

Por un lado, para probar el correcto funcionamiento escogió varias páginas webs aleatoriamente y la información a extraer, luego de aplicar el scraping los resultados fueron satisfactorios.

Por otro lado, estas pruebas ayudaron para corregir o simplificar aún más la forma de extraer la información y corregir ciertos errores de funcionamiento a la hora de mostrar en pantalla.

7. **Mantenimiento**

Esta fase no es una de las menos importante porque en esta se sitúa estar en un ciclo de mejora continua porque a la medida que se van configurando los sitios web, van surgiendo ideas de como optimizar el programa y agregar nuevas funcionalidades que permitan mejorar el programa final.

RECOMENDACIONES

Durante todo este tiempo se logró terminar el desarrollo de la plataforma para la extracción de información, documentos, etc., de sitios web, como apreciación personal hubiera sido interesante que todo el proceso de web scraping se realizara en Python, ya que esta es una de las mejores opciones como lo dice Sha (2021)

> Python es el idioma de facto para la ciencia de datos, y es ampliamente utilizado en la extracción de datos. Las bibliotecas de Python como BeautifulSoup, Scrapy y Selenium tienen la capacidad de realizar web scraping de manera eficiente y eficaz.

Continuando, el trabajo colaborativo y la constante comunicación en el equipo de desarrollo permitió que la plataforma se construyera mucho antes de lo establecido y también el haber subdivido el trabajo logró que cada uno se centrara en algo especifico.

> En el aprendizaje activo, el trabajo colaborativo es la vía para que el alumnado aprenda a interactuar con los demás seres, mediante un esfuerzo conjunto que tiene como fin la transformación de una realidad o la solución de un problema. (Revans, 1983)

CONCLUSIÓN

A modo de cierre, se puede decir que a partir de la combinación de los prototipos de los integrantes del equipo de programación se logra realizar el objetivo planteado que era la construcción de un software para la extracción de información de sitios web, la técnica de web scraping es la mejor forma de extraer información de páginas web y hay muchos lenguajes que permiten la realización de web scraping a manera de apreciación me quedo con el lenguaje Python, ya que cuantas con un conjuntos de librerías que permiten realizar la extracción de información, documentos, imágenes, etc., de manera mucho más sencilla. (Brooks, Apress, 2019) "Python es el idioma más popular para el web scraping, y por buenas razones. Las bibliotecas de Python como Beautiful Soup y Scrapy hacen que el web scraping sea fácil y accesible para todos".

REFERENCIAS BIBLIOGRAFICAS

Brooks, D. (2019). Apress. Obtenido de Web Scraping práctico para ciencia de datos: mejores prácticas y ejemplos con Python: https://www.apress.com/gp/book/9781484235816

Mathes, E. (Diciembre de 2022). Curso acelerado de Python, 3.ª edición. Obtenido de Una introducción práctica a la programación basada en proyectos: https://nostarch.com/python-crash-course-3rd-edition

Revans, R. W. (1983). Aprendizaje activo para el desarrollo de la psicomotricidad. Obtenido de Active Learning to Develop Motor Skills and Teamwork: https://www.revistas.una.ac.cr/index.php/EDUCARE/article/download/8476/12370?inline=1

Sha, R. (2021). O'Reilly Media, Inc. Obtenido de Manual de Python para ciencia de datos: herramientas esenciales para trabajar con datos.: https://www.oreilly.com/library/view/python-for-data/9781491912126/