



Scraper to API

**DESARROLLO DE APLICACIONES
PARA CIENCIA DE DATOS**

2º

**G. CIENCIA E INGENIERÍA DE
DATOS**



UNIVERSIDAD DE LAS PALMAS DE GRAN
CANARIA

ÍNDICE

1.Resumen

2.Recursos utilizados

3.Diseño

4.Conclusiones

5.Líneas futuras

6.Bibliografía

RESUMEN

El programa realizado en esta práctica tiene como principal función la extracción de los datos procedentes a una página web, en este caso “Booking.com”. Estos datos en específico son de los hoteles situados en Las Palmas de Gran Canaria. Los datos a extraer son: el nombre del hotel, la ubicación exacta de este, los servicios que ofrece, los comentarios de la gente y el rating del propio hotel.

Todo esto realizado en el lenguaje de programación java, mediante diferentes clases con funciones en específico. Un módulo en el cual se encuentran las clases POJO en las que instanciamos los objetos para ser utilizados en la clase llamada Scrapper, en la cual realizaremos la conexión a la página web de “Booking.com” a través de la url de la misma. Una vez conectados, estableceremos una “id” mediante la que podremos recorrer los diferentes hoteles que hay en la página web. A su vez, tenemos la oportunidad de coger la url en específica de cada hotel para poder recoger la información que necesitamos.

Mediante otra conexión a esta nueva url, ya podemos ir cogiendo los datos del hotel que queramos realizando diferentes select. Una vez seleccionada toda la información que necesitemos, tan solo queda crear una lista e ir añadiendo los hoteles con toda la información que hayamos recogido para así poder devolver esta lista cuando hagamos la request de nuestra API. Cuya API podemos acceder a ella mediante los siguientes get:

`/hotels/:name`

`/hotels/:name/services`

`/hotels/:name/comments`

`/hotels/:name/ratings`

R E C U R S O S U S A D O S

En cuanto a los entorno de desarrollo utilizados, se ha usado IntelliJ IDEA para la programación en java, y para la creación del diagrama de clases se ha usado StarUML.

A parte, también se ha hecho uso del navegador web de Google, para poder comprobar los resultados de la API a través de:

<http://localhost:4567/v1/hotels/>

También se ha usado la herramienta de GitHub para poder subir el proyecto a un repositorio al que se puede acceder mediante un link y poder visualizar el contenido del proyecto.

D I S E Ñ O

En cuanto a los patrones de diseño realizado, se ha intentado seguir el MVC (Model View Controller), para poder separar los datos de la aplicación, la interfaz de usuario y la lógica de control en tres componentes diferentes.

CONCLUSIONES

Gracias a como se ha realizado el programa, y lo complejo que me ha resultado entender como funcionaba, he logrado entender el concepto de hacer “scrapping” a una página, y a lo que esto conlleva, ya que buscando información en internet he llegado a la conclusión de que esta realización puede llegar a ser hasta ilegal, ya que si se hace de una manera profunda y en ciertas páginas, puedes llegar a obtener datos que ni yo mismo pensaba que se pudiera.

Si pudiera repetir el trabajo y tuviera más tiempo, a lo que más le dedicaría sería a la API, ya que a día de hoy entiendo mucho mejor como funciona gracias a que ya he terminado el Proyecto Final de esta asignatura.

LÍNEAS FUTURAS

Para hacer evolucionar este producto, lo que se me ocurre es implementar una interfaz gráfica, en la que el propio usuario sea el que escriba que dato desea recoger y de qué página web quiere extraer ese dato, y que con tan solo escribir esas dos cosas, el programa le devuelva por pantalla lo que ha pedido.

B I B L I O G R A F Í A

<https://www.booking.com/>

<https://www.bbvaapimarket.com/es/mundo-api/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos/>

<https://github.com/>



Joel
Clemente

López
Cabrerera

DESDE EL DÍA 1/01/2023 HASTA
EL DÍA 10/01/2023

2º

G. CIENCIA E INGENIERÍA DE
DATOS

UNIVERSIDAD DE LAS PALMAS DE GRAN
CANARIA