Ciencia Latina Revista Científica Multidisciplinar
Julio-Agosto, 2023, Volumen 7, Número 4.
https://doi.org/10.37811/cl_rcm.v7i4.7943

Indexación de Sitios Web para Optimizar la Búsqueda de Paquetes Turísticos Basado en Web Scraping

Gerardino Juvenal Cauna-Huanca[1]
gcaunah406@gmail.com
https://orcid.org/0000-0003-3733-9238
Universidad Nacional del Altiplano Puno
Perú

Jorge Condori Chávez
jococha644@gmail.com
https://orcid.org/0000-0003-1284-7850
Universidad Andina Néstor Cáceres Velásquez: Juliaca, Puno, PERÚ

Roger Quispe Caira
qcrogger@gmail.com
https://orcid.org/0009-0005-4739-4347
Universidad Nacional del Altiplano Puno- Perú

Edson Denis Zanabria Ticona
edsnzanabria@gmail.com
https://orcid.org/0000-0003-1708-8515
Universidad Nacional del Altiplano Puno
Perú

Ludwin Arocutipa Marca
ludwin.am@gmail.com
https://orcid.org/0000-0003-3463-3933
Universidad Nacional del Altiplano Puno- Perú

RESUMEN

La técnica del Web Scraping permite la extracción de contenido de varios sitios web, recabando información de interés para el usuario. El presente proyecto tiene como finalidad desarrollar un sitio web en la cual pueda almacenar información de los diferentes paquetes turísticos que son ofertados por las agencias de viaje que operan en la región de Puno utilizando la técnica del web Scraping. La población está conformada por 38 páginas web según inscritas en IPERÚ Puno. Para la elaboración del algoritmo de extracción se utilizó la metodología de desarrollo de software XP y para el contaste de la hipótesis se utilizó prueba de rangos con signo de Wilcoxon. Como resultado, el análisis de la estructura DOM permitió el desarrollo del algoritmo de extracción, haciendo uso de Python como lenguaje de programación, también se puso a prueba la eficiencia del algoritmo, el cual demostró ser eficiente en comparación con la el programa webscraper. Se determinó que la complejidad algorítmica es lineal. Del desempeño de nuestro sitio web según la puntuación global de PageSpeed Insights está en la categoría rápida (97 puntos). La evaluación del sitio web basado en la norma ISO 25000 proporcionó una valoración de 6.67/10 puntos como calidad total, considerado como nivel aceptable y grado satisfactorio. Se concluye que la implementación del sitio web facilita la búsqueda de diferentes paquetes turísticos.

Palabras clave: *paquetes; recuperación de información; sistema web; scraping; turismo*

[1] Autor principal.
Correspondencia: gcaunah406@gmail.com

Indexing Of Websites To Optimize The Search For Tourist Packages Based On Web Scraping

ABSTRACT

The Web Scraping technique allows the extraction of content from various websites, collecting information of interest to the user. The purpose of this project is to develop a website in which you can store information on the different tourist packages that are offered by travel agencies that operate in the Puno region using the web Scraping technique. The population is made up of 38 web pages as registered in IPERÚ Puno. For the elaboration of the extraction algorithm, the XP software development methodology was used and the Wilcoxon signed rank test was used to test the hypothesis. As a result, the analysis of the DOM structure allowed the development of the extraction algorithm, making use of Python as the programming language, the efficiency of the algorithm was also tested, which proved to be efficient compared to the webscraper program. The algorithmic complexity was determined to be linear. The performance of our website according to the global PageSpeed Insights score is in the fast category (97 points). The evaluation of the website based on the ISO 25000 standard gave a rating of 6.67 / 10 points as total quality, considered as acceptable level and satisfactory grade. It is concluded that the implementation of the website facilitates the search for different tourist packages.

Keywords: *information retrieval; packages; scraping; tourism; web system*

Artículo recibido 15 julio 2023

Aceptado para publicación: 25 agosto 2023

INTRODUCCIÓN

La sobreabundancia de información en Internet, es uno de los principales componentes de su éxito (Rizaldi & Putranto, 2017); sin embargo, el tratamiento de esta, exige una enorme cantidad de tiempo y energía a fin de seleccionar la calidad de los datos dentro de un enorme repositorio (Gheorghe et al., 2018). Según Villarroel Colque, (2015) refiere que en la actualidad la sobrecarga de información que recibe un usuario, en especial de Internet en todas sus formas, puede causarle la sensación de no poder abarcarla ni gestionarla y, por tanto, llegar a generarle una gran angustia, para (Toffler, 1974) menciona que hay demasiados conocimientos para tomar una decisión o para mantenerse constantemente informado sobre algún asunto.

La información que se muestra en las páginas web tiene como fin ser entendida y procesada por personas, por ello, resulta muy difícil que una máquina sea capaz de entender un texto que no esté estructurado. Actualmente las páginas web cuentan con metadatos, semántica que describen el contenido y la relación entre los datos, de forma que es posible evaluarlas automáticamente, además, la web 3.0 intenta conseguir que los datos sean identificables dentro de la estructura de internet, es decir, que las búsquedas sean mucho más concretas y fiables (Vàllez, 2017). Este motivo es una de las principales razones por las que emerge el concepto de web scraping que se va a aplicar durante el trabajo investigación.

Web Scraping (raspado web) es el proceso de extracción de datos específicos de sitios web (Julian & Natalia, 2015), usando un programa que simula la exploración humana mediante el envío de peticiones HTTP (Muñoz Mandujano *et al.*, 2018) o emulando un navegador web completo, además Web Scraping está muy relacionado con la indexación de la web, la cual indexa información utilizando un agente web automatizado y es una técnica global adoptada por la mayoría de los motores de búsqueda. En este sentido, cada vez que hemos realizado la acción de copiar y pegar por diferentes páginas de la web para obtener datos, y que posteriormente hemos utilizado para otra actividad diferente, lo que se ha realizado ha sido un raspado de datos en la web. Pero esta práctica resulta muy costosa al tener que emplear mucho tiempo en el caso de querer obtener muchos datos de diferentes páginas, luego organizar, estructurar, almacenar y analizar en una base de datos central, en una hoja de cálculo o en alguna otra fuente de almacenamiento (Hanretty, 2013; Zhao, 2017).

La tecnología y los procesos que lo permiten se han explorado ampliamente en el campo de la ciencia de datos, donde los investigadores aplican esta información a múltiples dominios de contenido (Landers *et al*., 2016); según lo descrito por Marres & Weltevrede (2013), en el periodismo, el raspado web se ha utilizado para evaluar la importancia de las noticias internacionales contando la cantidad de veces que los usuarios de las redes sociales mencionaron esas historias, Kiran & Mownika (2021) indica que el web Scraping es el procedimiento que se utiliza comúnmente en la minería de datos, Januzaj *et al*., (2019) utilizaron para comparar la demanda del mercado y los planes de estudio universitarios, Ullah *et al*., (2018) presentaron un estudio para extraer tendencias y sugerir el mejor precio de un producto objetivo, Muehlethaler & Albert, (2021) utilizo en la industria textil y afirma que en menos de 24 horas lograron extraer 68 campos basados en texto que describen un total de 24,701 prendas que le ayuda a realizar estimaciones, Almeida de Oliveira & Arantes Baracho Porto (2016) utilizo para analizar información relevante sobre las atracciones turísticas de Minas Gerais, *Landers et al.* (2016) lo utilizo en el área de la psicología para encontrar patrones de comportamiento humano, Muñoz Mandujano *et al.* (2018), lo empleo para almacenar datos climatológicos y Hernández *et al.* (2015), lo utilizo para realizar análisis político en las redes sociales. Por ello web Scraping es una técnica que facilita la extracción de información de forma masiva y automatizada, para que pueda ser usada de acuerdo a las necesidades del usuario (Huaman Hilari & Quispe Ramos, 2019; Dewi et al., 2019) por ello es una herramienta de empoderamiento (Uriarte et al., 2020).

Existen multitud de herramientas destinadas a la obtención de datos (Khalil & Fakir, 2017), indican que el lenguaje R proporciona un conjunto de funciones útiles para el rastreo web, mientras que (Rizaldi & Putranto, 2017), manifiesta que el uso del método del selector XPATH para los sitios de noticias de raspado web produce artículos más completos que el uso del método del selector CSS. La empresa BBVA, (2016) recomienda el lengua de programación Python para la extracción de datos no estructurados, para los usuarios con conocimientos de programación, de modo que, programando se pueda obtener y organizar la información residente en las diferentes páginas, en concreto, para nuestro caso de estudio, páginas web de agencias de viajes que operan en la región de Puno, dado que el turismo es uno de los sectores con enorme potencial de desarrollo, y cuenta con importantes recursos turísticos reconocidos y otros que recién están tomando auge. El sector del turismo es un ambiente muy dinámico,

los productos (paquetes turísticos) cambian continuamente, como así también los intereses de los usuarios y precisamente estos, deben pasar bastante tiempo en el ordenador o su dispositivo móvil a fin de poder encontrar un paquete acorde a sus necesidades.

Los objetivos de esta investigación fueron, desarrollar un software para la indexación de sitios web y optimizar la búsqueda de paquetes turísticos de la región de Puno basado en web Scraping y los objetivos específicos fueron, y los objetivos específicos fueron, desarrollar un algoritmo para extraer la información relevante de las páginas web de las agencias de viaje que operan en la región de Puno, implementar y determinar el rendimiento del sitio web basado en la experiencia de usuario que facilite la búsqueda de los paquetes turísticos. establecer el grado en que el producto satisface a un usuario final basado en la norma ISO/IEC 25000

MÉTODOS

Lugar de estudio

La región de Puno se encuentra en la parte sur del Perú. Está ubicada a orillas del lago Titicaca y sobre los 3,827 metros s.n.m. según el estudio realizado por Laurente & Machaca (2020) Puno es la cuarta región más visitada por los turistas internacionales.

Descripción de Métodos

La población estuvo conformada por 38 páginas web de las agencias de viaje que operan en la ciudad de Puno. La investigación es de tipo aplicada.

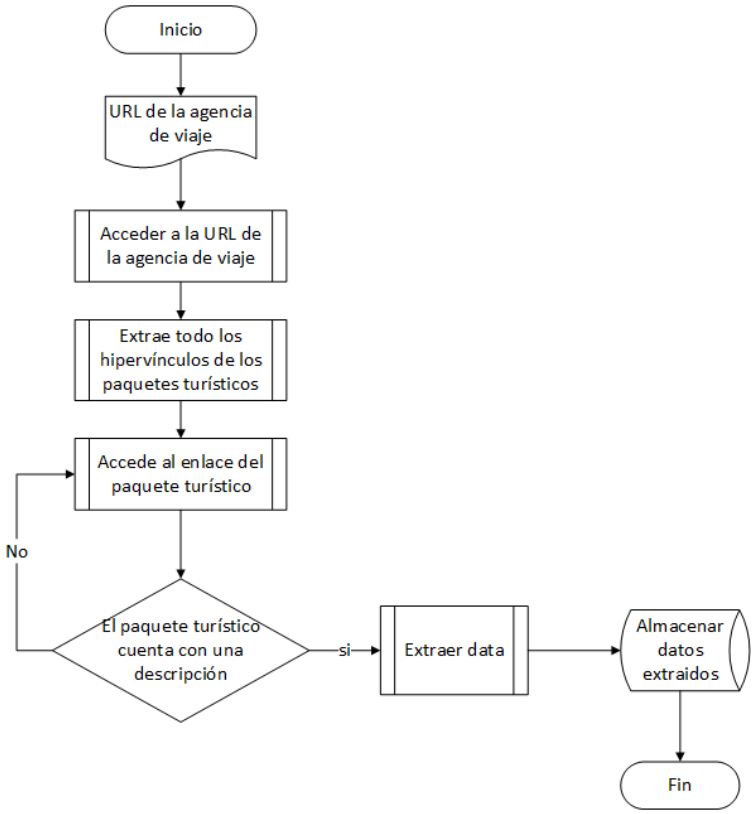
Para el desarrollo del algoritmo de extracción de datos

Se desarrollo con base en la metodología de desarrollo XP por ser muy rápida y ligera. Lo primero fue establecer la URL padre de cada página web, luego se analizó la estructura DOM para identificar los contenedores HTML repetitivos con ayuda del inspector de elementos del navegador; para ubicar el enlace que redirige a la información detallada del paquete turístico ofertado, finalmente establecer el selector de CSS para extraer la información relevante.

Se opto por hacer uso del lenguaje de programación de Python, el cual posee librerías para manipular contenido de la web. La librería Requests permitió realizar solicitudes HTTP sin necesidad de tanta labor manual, haciendo que la integración con los servicios web sea mucho más fácil. La librería BeautifulSoup permitió analizar y manipular gramáticamente documentos HTML, en un formato más

simple para su procesamiento y luego mediante código poder realizar las consultas y almacenarlo en una base de datos(Muñoz Mandujano et al., 2018).

Figura 1. *Diagrama de flujo para la implementación del algoritmo*



Para determinar el rendimiento del sitio web

PageSpeed Insights es una herramienta gratuita creada por Google que informa sobre el rendimiento de las páginas tanto en dispositivos móviles como en ordenadores, además, es capaz de ofrecer una serie de sugerencias y herramientas asociadas para mejorar los resultados. PSI de Google ofrece seis métricas que miden diversos aspectos del rendimiento relevantes para los usuarios, con ello se busca, reducir al máximo el número de llamadas HTTP realizadas, reducir a su mínima expresión el tamaño de las respuestas tras una petición HTTP y optimizar el renderizado de la página en el navegador del usuario

Para establecer el grado en que el producto satisface a un usuario final

Se utilizo familia de normas ISO/IEC 25000 que se desarrolla en el proyecto SQuaRE, y es el esfuerzo que hace la ISO para cubrir más temas relacionados a la calidad de producto software. El modelo de calidad de uso según la ISO/IEC 25010 define como el grado en el que un producto o sistema puede ser

utilizado por usuarios específicos para satisfacer sus necesidades y alcanzar sus objetivos específicos con eficacia, eficiencia, libertad de riesgo y satisfacción en un contexto específico de uso, y el modelo ISO/IEC 25022 define específicamente las métricas para realizar la medición de la calidad en uso del producto.

RESULTADOS Y DISCUSIÓN

Para lograra el objetivo específico 1, iniciamos el algoritmo definiendo la estructura de cada página web a ser extraída, indicando las variables asociadas a las etiquetas HTML y/o selectores CSS en un archivo denominado Config.yaml, el cual actúa como un mapa para realizar la extracción de los datos. YAML es un formato para guardar objetos de datos con estructura de árbol, este lenguaje es muy legible para las personas, más legible que un JSON y XML (Contreras, 2016).

Dado que la web es un lugar dinámico del cual no se tiene el control y puede ser modificado en cualquier instante, se debe de separar los selectores de la lógica de nuestro algoritmo, de tener alguna modificación en la web, basta con actualizar los selectores en el archivo YAML. El programa utilizara la variable news_sites para acceder a todas las url de los sitios web a ser scrapeado; cuando acceda a una página web, realizara las siguientes consultas:

Homepage_article_links: Obtiene el hipervínculo de los paquetes turísticos ofertados por la agencia de viaje

Article_title: Obtiene el nombre del paquete turístico

Article_body: Obtiene la descripción del paquete turístico

Itinerario: Obtiene el itinerario del paquete turístico

Incluye: Obtiene la descripción de los servicios que incluye el paquete turístico

Email: Obtiene los datos de contacto de la agencia de viajes

Img: Obtiene el link del logo de la agencia de viaje

Imagen: Obtiene el link de la imagen de portada del paquete turístico

En el archivo news_page_objects(ver figura 2) inicia importando la librerías BeautifulSoup, Requests e importamos la configuración de nuestra archivo YAML. La función _visit() recibe como parámetro la URL en consulta, su función es comunicarnos con la web, determinar el estado de la respuesta del servidor y enviarle como parámetro a la librería BeautifulSoup y extraer los datos del archivo HTML.

La función _select recibe como parámetro una cadena de caracteres que contiene un selector CSS y ayuda a obtener información del árbol de nodos que BeautifulSoup almaceno en la variable _html

Figura 2. *Código para realzar solicitudes y manipular las etiquetas HTML*

news_page_objects.py 1

agencia > extract > news_page_objects.py > HomePage > __init__

```python
import bs4
import requests
from common import config


class NewsPage:
    def __init__(self, news_site_uid,url):
        self._config = config()['news_sites'][news_site_uid]
        self._queries = self._config['queries']
        self._html = None
        self._visit(url)
        self._url=url

    def _select(self, query_string): # funcion auxilia
        return self._html.select(query_string)


    def _visit(self, url):
        response = requests.get(url)
        response.raise_for_status()
        self._html = bs4.BeautifulSoup(response.text, 'html.parser')
```

Se puso a prueba la eficiencia del algoritmo para la extracción de los datos, comparándolo con un software gratuito denominado Webscraper.io. Una vez instalado y configurado la extensión se procedió a registrar y analizar el tiempo promedio (en minutos) que le toma al software extraer los datos y la cantidad de paquetes, para efectos de prueba se configuro tres agencias de viajes dentro de Webscraper.io y se hizo el raspado de los datos, en la Tabla 1, se puede visualizar que Webscraper.io le tomas más tiempo, además la cantidad de paquetes turísticos extraídos es menor en comparación en el algoritmo de extracción propuesto en la presente investigación.

Tabla 1.

Comparación de tiempo y los datos extraídos

Agencia	Extensión Web Scraper		Extracción por el algoritmo	
	Tiempo promedio en min.	Cantidad de paquetes extraídos	Tiempo promedio en min.	Cantidad de paquetes extraídos
Arcobaleno	01:43:02	27	00:29:45	31
Andespuno	00:36:29	10	00:17:45	12
Amarutours	00:16:19	06	00:11:74	09

Como indica Ujwal et al.(2017) la técnica del web scraper permite la extracción de la información con una alta precisión y recuperación en bloques repetitivos, además la implementación del algoritmo en el lenguaje de programación Python facilito el desarrollo, al poseer una gran cantidad de librerías integradas de forma estándar a las que se le pueden adjuntar otras librerías (Hernández Herrero, 2014)

Se realizo el análisis de la complejidad algorítmica con la notación Big O del código para la extracción de todos los enlaces de los paquetes turísticos encontrados en una página web.

Figura 3. *Complejidad algorítmica para la obtención de hipervínculos*

news_page_objects.py 1 ×

agencia > extract > news_page_objects.py > ...

```
23  class HomePage(NewsPage):
24      def __init__(self, news_site_uid,url):
25          super().__init__(news_site_uid, url)
26
27
28      @property
29      def article_links(self):
30          link_list = []      #O(1)
31          for link in self._select(self._queries['homepage_article_links']):  #O(n)
32              if link and link.has_attr('href'):  #O(2)
33                  link_list.append(link)  # O(1)
34          return set(link['href'] for link in link_list)  # O(1)
35
```

La línea de código número 30 se crea una lista vacía, el cual tiene una valoración de O(1) por que actúa como una constante y siempre va inicializar en una lista vacía cada que se invoque a la función article_links

El ciclo for de la línea de código numero 31 itera dependiendo de cuantos enlaces va recorrer y eso equivale a un O(n)

La línea de código numero 31 realiza dos comparaciones las cuales va devolver un verdadero o un falso, en el peor de los casos de que ambos fueran verdaderos realiza dos acciones eso equivale a un O(2)

La línea de código numero 33 agrega elementos a la lista vacía, eso equivale a O(1)

La línea de código numero 34 va a retornar un valor, lo cual equivale O(1)

Realizamos la sumatoria

$O(G) = O(1) + O(n) + O(2) + O(1) + O(1)$

$O(G) = O(n+ 5)$

$O(G) = O(n)$

Por lo tanto, podemos determinar que la complejidad algorítmica para la extracción de los hipervínculos de los paquetes turísticos es lineal O(n) y el tiempo de ejecución es cada vez mayor de modo proporcional a cómo se incrementa el tamaño de los enlaces encontrados en cada página web.

Para lograr el objetivo específico 2, se implementó la página web gopuno.com, se utilizó CodeIgniter que es un entorno de desarrollo web escrito en PHP, cuya base es el patrón Modelo-Vista-Controlador (MVC) y permite diseñar software de forma flexible. Para determinar el rendimiento del sitio web en dispositivos móviles como en ordenadores utilizaremos la API de PageSpeed Insights (PSI) de Google el cual nos ofrece seis métricas que miden los diversos aspectos del rendimiento relevantes para los usuarios.

First Contentful Paint (FCP): mide cuánto tiempo le toma al navegador procesar la primera parte del contenido DOM después de que un usuario navega a la página web.

Tabla 2.

Puntuación FCP

Tiempo FCP (en segundos)	Código de colores	Puntuación FCP (percentil de archivo HTTP)
0-2	Verde (rápido)	75-100
2-4	Naranja (moderado)	50-74
Más de 4	Rojo (lento)	0-49

Luego de realizar el análisis se obtuvo un resultado de 1.8 segundos para dispositivos móviles y 0.5 para ordenadores, en ambos casos está dentro de la categoría rápido (color verde), el cual indica que no le toma mucho tiempo al navegador procesar el contenido DOM desde que comenzó la navegación hasta que el contenido principal de la página se muestre en la pantalla.

Speed index: mide la rapidez con la que se muestra visualmente el contenido durante la carga de la página.

Tabla 3.

Puntuación del Speed index

Tiempo FCP (en segundos)	Código de colores	Puntuación del índice de velocidad
0-4.3	Verde (rápido)	75-100
4.4-5.8	Naranja (moderado)	50-74
Más de 5.8	Rojo (lento)	0-49

Luego de realizar el análisis se obtuvo un resultado de 2.9 segundos para dispositivos móviles y 1.1 para ordenadores, en ambos casos está dentro de la categoría rápido (color verde), lo cual indica el tiempo en mostrar el contenido del sitio web

Largest Contentful Paint(LCP): informa el tiempo de renderizado de la imagen o el bloque de texto más grande visible dentro de la ventana gráfica.

Tabla 4.

Puntuación Largest Contentful Paint

Tiempo FCP (en segundos)	Código de colores	Puntuación del índice de velocidad
0-4.3	Verde (rápido)	75-100
4.4-5.8	Naranja (moderado)	50-74
Más de 5.8	Rojo (lento)	0-49

Fuente: (PageSpeed Insights, 2021)

Como resultado se obtuvo 2.3 segundos en dispositivos móviles y 0.8 segundos para ordenadores. tiempo desde que la página comienza a cargarse hasta que el bloque de texto o elemento de imagen más grande se representa en la pantalla, en ambos casos están en la categoría rápido (color verde)

Time to Interactive (TTI): mide el tiempo que tarda una página en volverse completamente interactiva.

Tabla 5.

Puntuación Time to Interactive

Métrica TTI (en segundos)	Código de colores
0 – 3.8	Verde (rápido)
3.9 – 7.3	Naranja (moderado)
Más de 7.3	Rojo (lento)

Como resultado se obtuvo 2.3 segundos en dispositivos móviles y 0.6 segundos para ordenadores, ambos casos están en la categoría rápido (color verde), lo que indica que la página muestra contenido útil, los controladores de eventos están registrados para la mayoría de los elementos visibles y la página responde a las interacciones del usuario en 50 milisegundo

Total Blocking Time: mide la cantidad total de tiempo que una página está bloqueada para que no responda a la entrada del usuario, como los clics del mouse, los toques de la pantalla o las pulsaciones del teclado.

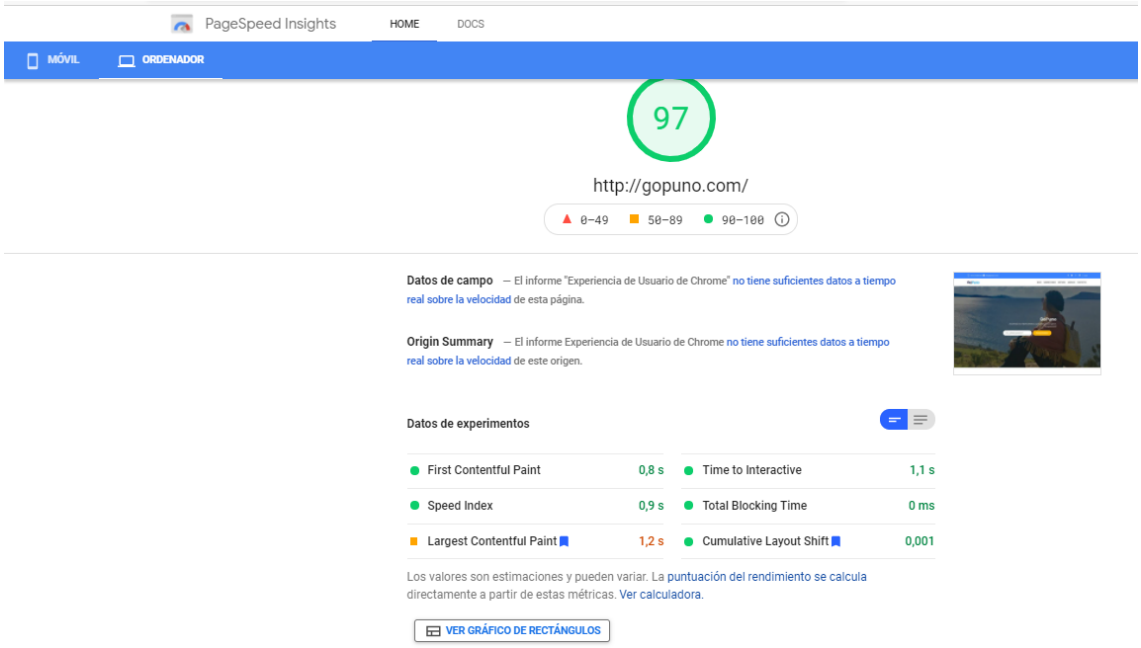
Tabla 6.

Puntuación Total Blocking Time

Tiempo TBT (en milisegundos)	Código de colores
0 – 300	Verde (rápido)
300 – 600	Naranja (moderado)
Más de 600	Rojo (lento)

El informe que generó PSI muestra una puntuación global que resume el rendimiento del sitio web GoPuno, el desempeño para dispositivos móviles y ordenadores según la puntuación global está en la categoría rápida (color verde), el cual garantiza una buena experiencia de usuario.

Figura 4. Puntuación global del rendimiento para ordenadores



Para lograra el objetivo específico 3, la métrica de la calidad en uso de un producto se software según la ISO/IEC 25022, se aplicó a 10 participantes y de acuerdo a los valores obtenidos de las características de calidad en uso evaluadas, se tuvo un resultado satisfactorio, lo que indica que el nivel de uso del sitio web el usuario se encuentra satisfecho con la utilización, además Marcos et al. (2008) indica que el control de la calidad se debe realizar desde un punto de vista cuantitativo.

Tabla 7.

Valor total obtenido de la calidad en uso.

Característica	Valor parcial total (/10)	Nivel de importancia	Porcentaje de importancia	Valor final	Calidad del Sistema
Efectividad	10	A	30%	3	
Eficiencia	5.57	A	30%	1.67	6.67
Satisfacción	5	A	40%	2	

Según Vaca & Jácome, (2018) afirma la importancia de aplicar un modelo de calidad de software con el fin de garantizar la calidad de datos y satisfacer requerimientos, además la familia de normas ISO/IEC 25000 proporciona flexibilidad, vigencia y estructura técnica para su aplicación

CONCLUSIONES

El análisis de la estructura DOM de cada uno de los sitios web de las agencias de viaje que operan en la región de Puno, permitió el desarrollo del algoritmo de extracción, limpieza y almacenamiento de los datos haciendo uso de Python como lenguaje de programación, también se puso a prueba la eficiencia del algoritmo, el cual demostró ser mucho más rápido y la cantidad de datos extraídos fue mayor en comparación con la extensión de Google denominado webscraper.io, además se determinó que la complejidad algorítmica es lineal O(n) para la extracción de los datos, que el tiempo de ejecución es proporcional a la cantidad de páginas a extraer. La evaluación del sitio web basado en la norma ISO 25000 proporcionó una valoración de 6.67 sobre 10 puntos como calidad total. considerado como nivel "Aceptable" y grado "Satisfactorio"; resultado que evidenció la importancia de tomar en cuenta métricas de calidad en uso según el modelo de calidad de software aplicado.

AGRADECIMIENTO

Mis más sinceros agradecimientos a la Unidad de Post Grado de la Facultad de Ingeniería estadística e Informática

REFERENCIAS BIBLIOGRÁFICAS

Almeida de Oliveira, R., & Arantes Baracho Porto, R. M. (2016). Extração de dados do site tripadvisor como suporte na elaboração de indicadores do turismo de minas gerais: Uma iniciativa em big data. *Pesq. Bras. em Ci. da Inf. e Bib.*, *11*(2), 026-037. https://repositorio.ufmg.br/handle/1843/ECIP-AN2PRB

BBVA. (2016, enero 11). *Herramientas de extracción de datos: Para principiantes y profesionales*. BBVAOpen4U. Recuperado de https://bbvaopen4u.com/es/actualidad/herramientas-de-extraccion-de-datos-para-principiantes-y-profesionales

Contreras, F. (2016, septiembre 27). *Conoce que es un YAML - fercontreras*. Recuperado de https://fercontreras.com/conoce-que-es-un-yaml-e18e9d21ade4

Dewi, L. C., Meiliana, & Chandra, A. (2019). Social Media Web Scraping using Social Media Developers API and Regex. *Procedia Computer Science*, *157*, 444-449. https://doi.org/10.1016/j.procs.2019.08.237

Gheorghe, M., Mihai, F.-C., & Dârdal, M. (2018). Modern techniques of web scraping for data scientists. *Revista Romana de Interactiune Om-Calculator*, *11*(1), 63-75. http://rochi.utcluj.ro/rrioc/articole/RRIOC-11-1-Gheorghe.pdf

Hanretty, C. (2013). Scraping the Web for Arts and Humanities. *UNIVERSITY OF EAST ANGLIA*, 50. https://silo.tips/download/s-c-r-a-p-i-n-g-t-h-e-w-e-b-f-o-r-a-r-t-s-a-n-d-h-u-m-a-n-i-t-i-e-s

Hernández, A. T., Vázquez, E. G., Rincón, C. A. B., & García, J. M. (2015). Metodologías para análisis político utilizando Web Scraping. *Research in Computing Science*, *95*, 113-121. https://www.researchgate.net/publication/339207165_Metodologias_para_analisis_politico_utilizando_Web_Scraping

Hernández Herrero, C. (2014). *Aplicación de Técnicas de Web Scraping al Boletín Oficial de Castilla y León (BOCyL)* [Universidad de Valladolid]. https://uvadoc.uva.es/handle/10324/5794

Huaman Hilari, J. Z., & Quispe Ramos, M. A. (2019). *Modelo de búsqueda de productos alimenticios en supermercados online categoría abarrotes utilizando asistente virtual de tipo Chatbot y extracción de datos con Web Scraping*. Universidad Tecnológica del Perú. https://hdl.handle.net/20.500.12867/2381

Januzaj, Y., Luma, A., Aliu, A., Selimi, B., & Raufi, B. (2019). *WEB DATA SCRAPING TECHNIQUE AND PREPARATION FOR COMPARISON TECHNIQUES BETWEEN DIFFERENT DOCUMENTS*. *11*, 17. https://publons.com/publon/28225522/

Julian, L. R., & Natalia, F. (2015). The use of web scraping in computer parts and assembly price comparison. *2015 3rd International Conference on New Media (CONMEDIA)*, 1-6. https://doi.org/10.1109/CONMEDIA.2015.7449152

Khalil, S., & Fakir, M. (2017). RCrawler: An R package for parallel web crawling and scraping. *SoftwareX*, *6*, 98-106. https://doi.org/10.1016/j.softx.2017.04.004

Kiran, M., & Mownika, N. (2021). Machine learning integrated emotions detection on lockdowns in India using advanced web scraping. *Materials Today: Proceedings*. https://doi.org/10.1016/j.matpr.2021.01.460

Landers, R. N., Brusso, R. C., Cavanaugh, K. J., & Collmus, A. B. (2016). A primer on theory-driven web scraping: Automatic extraction of big data from the Internet for use in psychological research. *Psychological Methods*, *21*(4), 475-492. https://doi.org/10.1037/met0000081

Laurente Blanco, L. F., & Machaca Hancco, R. W. (2020). Modelamiento y proyección de la demanda de turismo internacional en Puno-Perú. *Revista Brasileira de Pesquisa em Turismo*, *14*(1), 34-55. https://doi.org/10.7784/rbtur.v14i1.1606

Marcos, J., Arroyo, A., Garzás, J., Piattini, M., Marcos, J., Garzas, J., & Arroyo, A. (2008). La norma ISO/IEC 25000 y el proyecto KEMIS para su automatización con software libre. *Revista Española de Innovación, Calidad e Ingeniería del Software, 4*(2), 133-144. https://www.redalyc.org/articulo.oa?id=92218339013

Marres, N., & Weltevrede, E. (2013). SCRAPING THE SOCIAL?: Issues in live social research. *Journal of Cultural Economy*, *6*(3), 313-335. https://doi.org/10.1080/17530350.2013.772070

Muehlethaler, C., & Albert, R. (2021). Collecting data on textiles from the internet using web crawling and web scraping tools. *Forensic Science International*, 110753. https://doi.org/10.1016/j.forsciint.2021.110753

Muñoz Mandujano, M., Hernández Valerio, J. S., González Serrano, S. R., & Pérez Liévana, A. (2018). Web scraping para la recopilación de datos meteorológicos. *Revista NTHE*, *24*, 91-95. https://www.researchgate.net/publication/336140626_Web_scraping_para_la_recopilacion_de_datos_meteorologicos

Murillo, D., & Saavedra, D. (2017). Web Scraping de los Perfiles y Publicaciones de una Afiliación en Google Scholar utilizando Aplicaciones Web e implementando un Algoritmo en R *4to Congreso Internacional AmITIC 2017*, 8. https://revistas.utp.ac.pa/index.php/memoutp/article/view/1465/2111

PageSpeed Insights. (2021). https://developers.google.com/speed/pagespeed/insights/

Rizaldi, T., & Putranto, H. A. (2017a). Perbandingan Metode Web Scraping Menggunakan CSS Selector dan Xpath Selector. *Teknika*, *6*(1), 43-46. https://doi.org/10.34148/teknika.v6i1.56

Rizaldi, T., & Putranto, H. A. (2017b). Perbandingan Metode Web Scraping Menggunakan CSS Selector dan Xpath Selector. *Teknika*, *6*(1), 43-46. https://doi.org/10.34148/teknika.v6i1.56

Toffler, A. (1974). *El «Shock» del futuro*. Plaza & Janés.

Ujwal, B. V. S., Gaind, B., Kundu, A., Holla, A., & Rungta, M. (2017). Classification-Based Adaptive Web Scraper. *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 125-132. https://doi.org/10.1109/ICMLA.2017.0-168

Ullah, H., Ullah, Z., Maqsood, S., & Hafeez, A. (2018). Web Scraper Revealing Trends of Target Products and New Insights in Online Shopping Websites. *International Journal of Advanced Computer Science and Applications*, *9*(6), 6. https://doi.org/10.14569/IJACSA.2018.090658

Uriarte, J. I., Toro, G. R. R. M. de, & Larrosa, J. M. C. (2020). Web scraping based online consumer price index: The “IPC Online” case. *Journal of Economic and Social Measurement*, *44*(2-3), 141-159. https://doi.org/10.3233/JEM-190464

Vaca, T., & Jácome, A. (2018). *Calidad de software del módulo de talento humano del sistema informático de la Universidad Técnica del Norte bajo la norma ISO/IEC 25000*. https://www.researchgate.net/publication/325022337_Calidad_de_software_del_modulo_de_talento_humano_del_sistema_informatico_de_la_Universidad_Tecnica_del_Norte_bajo_la_norma_ISOIEC_25000

Vàllez, M. (2017). Tesis doctoral – Síntesis. Exploración de procedimientos semiautomáticos para el proceso de indexación en el entorno web. *HIPERTEXT.NET. Anuario Académico sobre Documentación Digital y Comunicación Interactiva*, *15*, 91-99. https://doi.org/10.2436/20.8050.01.50

Villarroel Colque, K. (2015). Infoxicación. *Revista de Investigación Scientia*, *4*(1), versión On-line. http://www.revistasbolivianas.org.bo/scielo.php?pid=S2313-02292015000100006&script=sci_arttext

Zhao, B. (2017). Web Scraping. En L. A. Schintler & C. L. McNeely (Eds.), *Encyclopedia of Big Data* (pp. 1-3). Springer International Publishing. https://doi.org/10.1007/978-3-319-32001-4_483-1