International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 06 | June - 2021 ISSN: 2582-3930

Web Scraping with Browser Automation Techniques

Vijay Choudhary[1], Anuj Kushwah[2], Ujjwal Patidar[3]

[1]Department of Computer Science, Acropolis Institute of technology & Research, Indore - 452001, Madhya Pradesh, India.
[2]Department of Computer Science, Acropolis Institute of technology & Research, Indore - 452001, Madhya Pradesh, India.
[3]Department of Computer Science, Acropolis Institute of technology & Research, Indore - 452001, Madhya Pradesh, India.

Abstract – In this work we are going to study about the Combination of Web scraping with Browser Automation techniques. If we need to fetch some information from website. To do so, copy and paste the data displayed by the website is very tedious job that may take many hours. So here comes the need of web scraping. Web scraping is process of getting data from websites using programming and replaces the manual work of copy pasting data from different websites. Browser Automation is process of automating Browsers like Google chrome, Firefox using Programming and tools like Puppeteer, Selenium.

Key Words: Web Scraping, browser automation, Puppeteer

1. INTRODUCTION

First of all, lets discuss about Web scraping and Browser Automation and their combination to make things easier. Combining Web scraping with Browser automation make different complicated processes easier.

1.1 Web Scraping

It is Process in which we can gather specific data from websites and can store it in database, spreadsheet for later fast retrieval. It includes acquiring data using HTML request libraries and then parsing it to get the exact information we want.

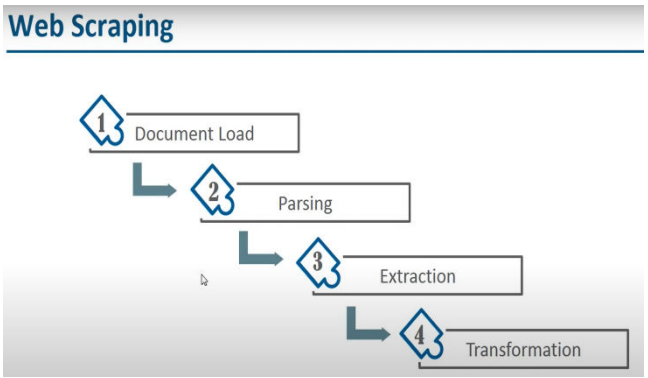

Fig -1: Web scraping

1.2 Browser Automation

It is a process in which we automate the browser and make browser perform tasks we want like opening a website, clicking on links, waiting for some time. The reasons for which browser automation is required is to test end-to end against application and to gather information from website having no API for gathering data.

2. Methodology

2.1 Web Scraping Using NodeJS

Suppose I want to scrape product details from Flipkart.com then we can use page.goto() method to reach website

```
await page.goto('https://www.flipkart.com')
```

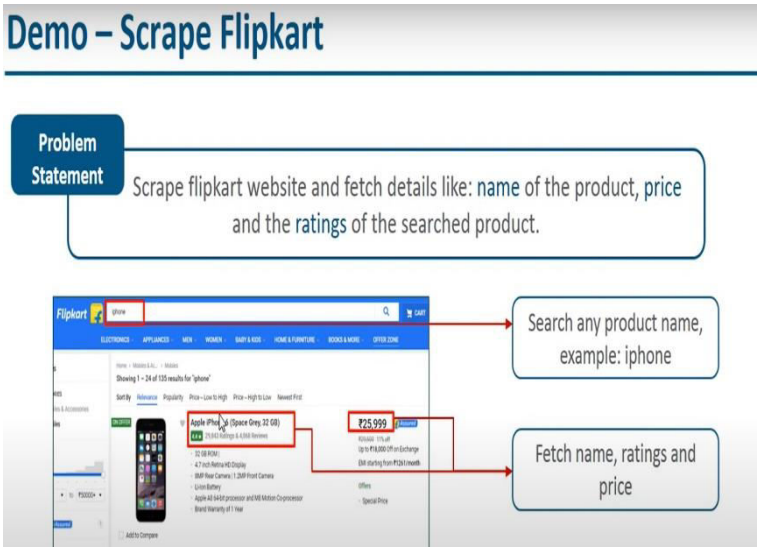

Fig -2: Demo-Scrape Flipkart

Then we need to inspect and find html tags. We can see html tags by right clicking and using inspect option on browser.

Fig -3: Inspecting Elements

We can then use the tags in our code like this

```
const attr1 = $('h2>a').map((i,element) =>
                $(element).attr("href")).get(0);
```

We can use Console.log() to print the extracted data.

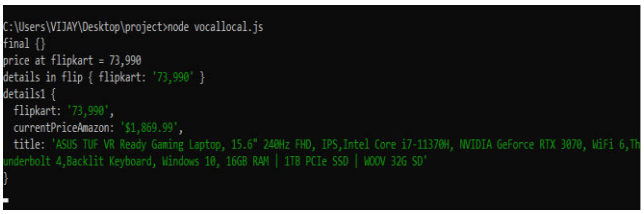

Fig -4: Scraped data from Websites

2.2 Browser Automation Using Puppeteer

Puppeteer is a Nodejs library which is used to control Chrome or chromium browser. It can be used headless or non-headless. First of all we need to include it in our code using

```
const puppeteer =require('puppeteer');
```

We can use this code to launch it in NodeJS program

```
const browser =await puppeteer.launch({
        headless: false
});
```

The headless browser means the Puppeteer is interacting with a chrome browser as a background application, which means that the chrome UI is not visible on the screen.

We can use puppeteer to go to specific websites of our choice

```
const page =await browser.newPage();

await page.goto('https://www.amazon.com');

await page.type('#twotabsearchtextbox','asus tuff');
```

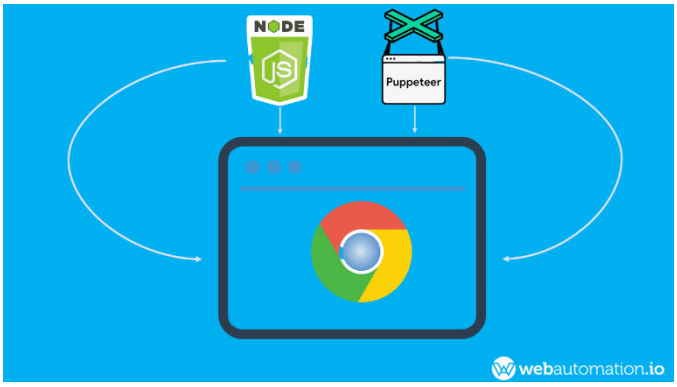

Fig -5: Browser Automation

2.3 Saving the Extracted data

Csv is good format to save the extracted data and present it to users because files can be opened with spreadsheets like excel, google etc. There are various JS modules to use in writing Json files.

3. CONCLUSIONS

In this study we discussed web scraping using NodeJS and Browser Automation using Puppeteer NodeJS library. Combining web scraping and browser automation can be really helpful in solving real world problems of manual work. Complex scraping operations can be managed by using automation tools to greatly increase success rate.
Here we have used flipkart website to perform web scraping and shown usage of puppeteer to make things easier and good. This combination of Nodejs and puppeteer is very useful as it provides headless mode for web scraping which is not provided by any other Automation tools.

REFERENCES

1. Mayank Dhiman Breaking Fraud & Bot Detection Solutions *OWASP AppSec Cali' 2018* Retrieved February 10, 2018.

2. National Office for the Information Economy (February 2004). "Spam Act 2003: A practical guide for business"(PDF). Australian Communications Authority.p. 20. Retrieved 2017-12-07.

3. National Office for the Information Economy (February 2004). "Spam Act 2003: An overview for business". Australian Communications Authority. p. 6. Retrieved 2017-12-07.