Website Scraping

Divyesh Vasudeo Aswar

roll no.: 209

Ramanand Arya D.A.V College, Bhandup(east), Mumbai – 400042.

email: aswardivyesh@gmail.com

Introduction:

The Internet today is a huge repository of information, the larger portion of it being accessible via the World Wide Web. The Web Browser being the standard tool to access this information. Although the browser is a great tool, it is limited to human users. With such a large set of information available online, it would be helpful if machines could automatically grab the content in some way. This could be for re-purposing data, analysis or creating mashups.

Many of the large websites today provide access to their content with the help of an API, either using REST or SOAP protocols. Users can retrieve the content from the site using the API and repurpose it however they wish (of course while adhering to the terms and conditions of the site). Unfortunately for the most part websites do not provide an API and the only way to get to the data is via web scraping, also know as spidering or screen scraping.

What is Web Scraping:

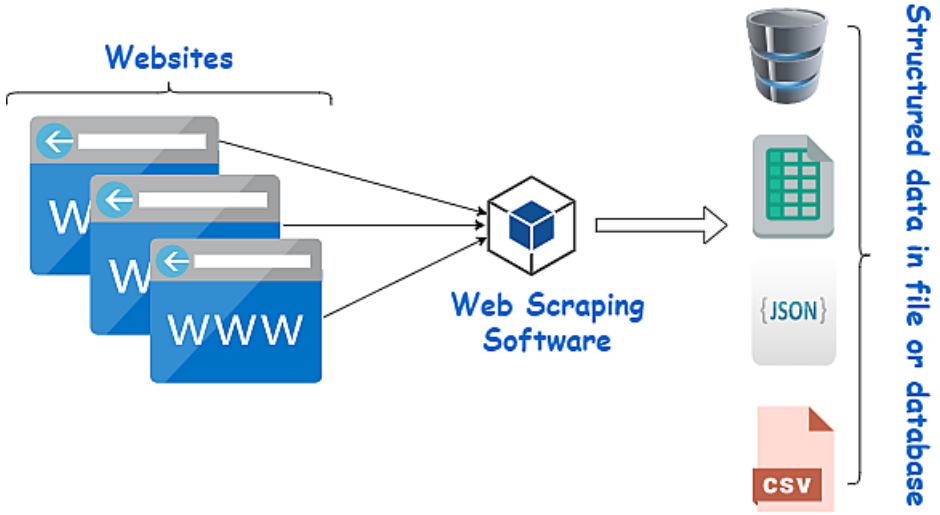
Web scraping is a method involving the automatic retrieval by programs, of semi-structured data from web pages. Commonly today a web page is built in a markup language such as HTML or XHTML as shown below.

```
<html>
<head>
<title>Hello HTML</title>
</head>
<body>
<p>Hello World!</p>
</body>
</html>
```

Web Scraping Framework:

As we can see in the code above, the information here is the string 'Hello World!', while most of the page content is HTML markup, which the browser renders for the user. If we need to scrape the above 'Hello World!' string using a computer program, than we will need to download the page and parse the content suitably, eliminating all the superfluous HTML markup to get to the pure text content. Of course the markup example given above is a very simple one. In reality web pages are very complex, harboring various HTML elements in a variety of combinations. Some of the HTML may be ill formed, missing tags or being nested incorrectly. Modern browsers usually ignore these problems and try to auto correct

the inconsistencies before displaying a page. However, when we are writing a web scraper we have to take all of these factors into consideration. This can make parsing web pages a difficult task. Fortunately there are various libraries and tools available in Python to make that undertaking easier for us.



Reasons to scrape:

Now that we have defined scraping and had a cursory look at the idea, we need to answer the question: why? Why bother to scrape?

Still providing some answers to the above question will help us enlarge our scraping skills to a wide variety of domains.

a. Aggregate and search specific kind of data:-

Although different websites provide different types of data, most of them are semantically connected in some way. For example if we are interested in blogs related to science, and we have around 100 blog feeds in our reader, it would be difficult to go through all of them on a regular basis and find items of particular interest. However, we can write a scraper to collect all the blog feeds and search for a particular keyword of interest, thus transferring the drudgery of data filtering to a machine.

b. Gaining automated access to web resources:-

If we need to regularly check a price for some product on an ecommerce store to see if any discount is available, we could regular visit the site to check on it. However, that would be time consuming and tedious. A better way would be to write a small scraper program that would regular visit the site and get the price, and email we if some price change is found. We could also regularly check for fresh images and download them to our computer.

c. Combine information and present it in an alternate format:-

This method, one of the most common uses of scraping, also known as 'mashups', allows we to gather different kind's of information from various sites and combine them in some interesting way that would be valuable to the end user.

Future of Web Scraping:

Web scraping is becoming increasingly important as the amount of data available on the internet grows. Many businesses are now giving customized web scraping solutions to their customers, which collect data from all across the

internet and organise it into valuable and understandable information. It saves time and money by eliminating the need to manually visit each page and collect data. Crawlers undertake wide scraping, while web scrapers are created and coded for each and every page. Scraping data from a difficult website requires more coding than scraping data from a simple one. Web scraping has a bright future ahead of it, and it will become increasingly important for all businesses as time goes on.

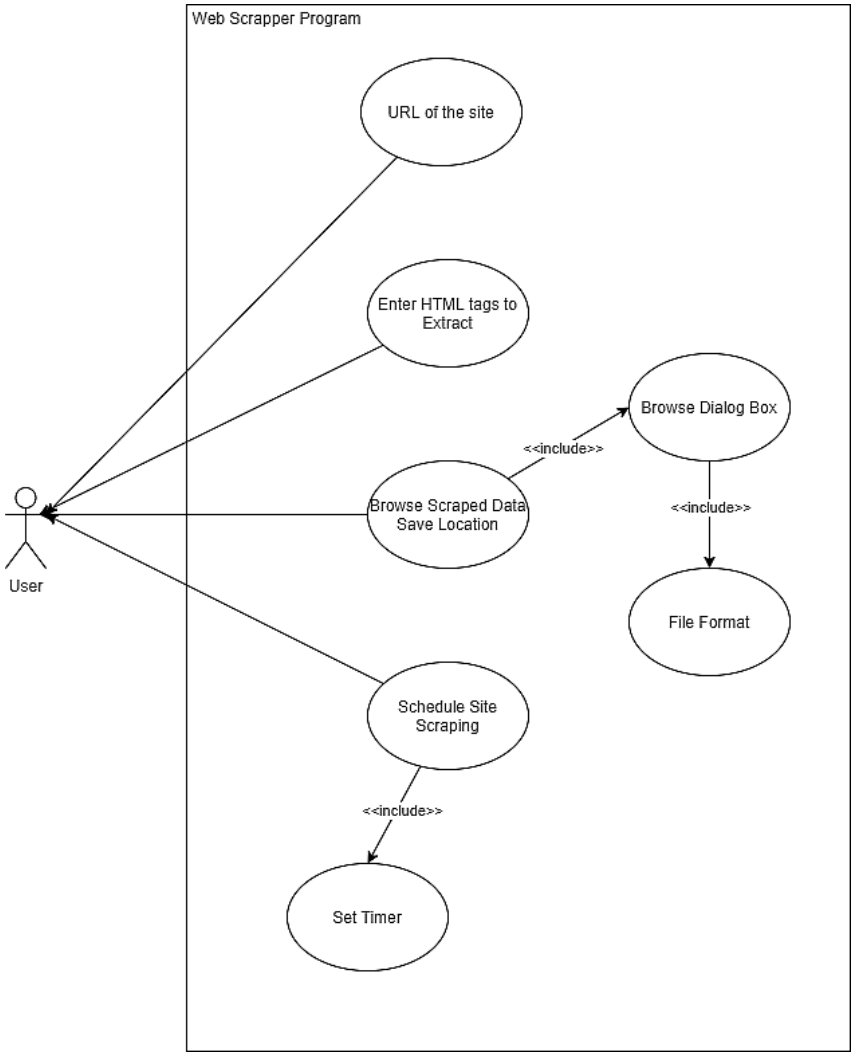
Aim of the Project:

Web Scraping is fundamental to gather relevant data form the internet weather it may be for research purposes or for general data gathering of for tracking of certain information like product prices form online stores.

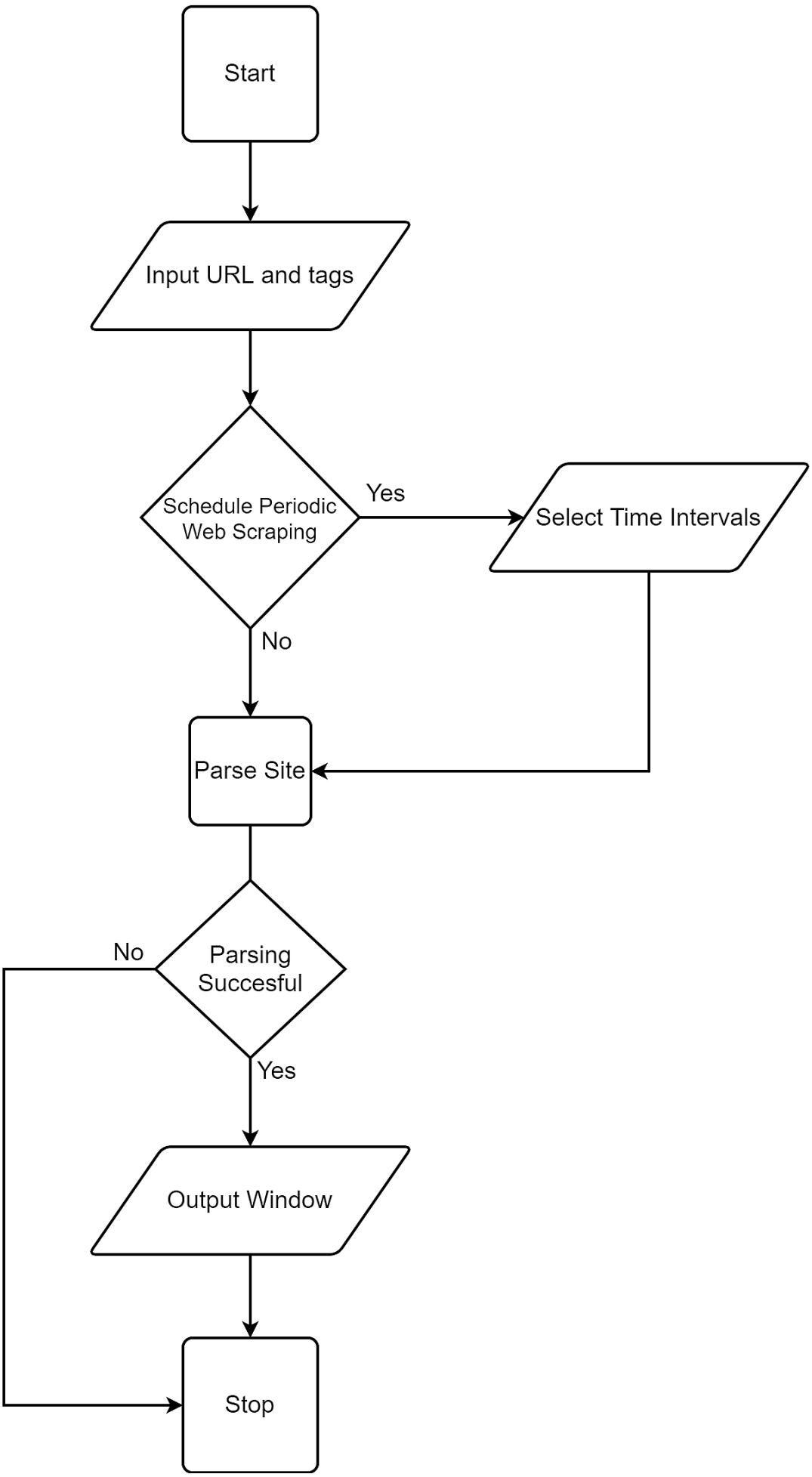
Scope of the Project:

The scope of the project is better understood as a system for data gathering form the internet and then storing it on a local device like a personal computer. The software can identify and track data on most website and scrape it from the website using dedicated tags provided by the user. The user can also provide the software with a time interval in order to ensure that the software scrapes desired data in a periodic manner, for example it the user set the time interval for 1 hour the software will scrape the data from the desired website periodically that is every 1 hour.

Use Case Diagram:



Flow Chart:


Start
Input URL and tags
Schedule Periodic Web Scraping
Yes
Select Time Intervals
No
Parse Site
Parsing Succesful
No
Yes
Output Window
Stop

Functional Requirement:

There are number of things we need to keep in mind in order to make this software functional:

URL entry page:
This is the page where we have to enter the address of the website which one has to scrape.

Enter Tag:
This is where you enter a specific HTML tag which you desire to scrape data from, this allows us to limit data usage and save space on disk otherwise the entire site could be scraped.

Parse Site data:
Once the URL and tags are added the software starts scraping the data from the site by first reading the page and the decoding to extract relevant information.

Data Storage:
Once the data is being scraped then it should be save on a device like a personal computer, in order to achieve that user can select a folder to store data of the desired website.

System Requirements:

User is required to have Google Chrome Web Browser and a decent personal computer in order to run Web Scraping and storing the data on the Hard disk.

Minimum requirements:

1. Hardware:

- RAM – 1 GB
- CPU – Dual Core Processor
- HDD – 150 GB
- Ethernet or WiFi – 1 Mbps

2. Software:

- Web Scraping program(.py or .exe) file
- Google Chrome Browser or any equivalent software

Problem Statement:

This project requires user to submit a URL of the website which he/she wants to scrape, the program also requires user to provide specific html tags in order to determine which specific data to scrape from the given website. But user may encounter some basic problem while scraping the website:

- The URL provided by the user is not working that is the website is down and isn't functional.

- The website blocks access of web scraping programs in general and do not allow to copy HTML data in order to protect proprietary information.
- User did not specify correct tag from the website, this could lead to download of wrong data from the site.
- The program may fail if the internet is not working.
- User's computer may lag of work slower because of the fact that user is trying to download large sums of data from the site which may overwhelm the CPU, RAM and HDD.

Problem Solution:

In order to overcome the above mentioned problems we need to take the following steps:

- User may need to check weather the site is working or not before entering the URL of the site in the Software.
- Website which block access to the Scraping software may be doing so in order to protect its data therefore user can search for a similar site on the internet and use that for scraping purposes.
- Software may not work scrape wrong data if the tags provided are wrong to solve this issue user can review the tags and provide the correct ones.
- User may have to check weather the Hardware to his or hers computer is up to the mark as specified in the System Requirements, if not then user may have to deal with some lag.

Feasibility Study:

A feasibility study is a preliminary investigation conducted to identify and document the viability of a project. The paper that results from the feasibility study is also referred to as a feasibility study.

The findings of this investigation are utilised to determine whether or not the project should be pursued. If it does really lead to a project's approval, it will do so before the project's actual work is completed. It is an examination of various alternative solutions to a problem and a suggestion for the best option. It can, for example, determine whether an order should be processed by a new, more efficient system than the prior one.

Because just one out of every fifty ideas is commercially feasible, a feasibility study is a crucial aspect of developing a business strategy for a new venture. If the results of the investigation indicate that a project is feasible, the next natural step is to carry it out. The feasibility study's research and information will help with detailed planning and cut down on research time.

Technical:

The capacity of a process to take use of the current state of technology in order to pursue additional improvements is referred to as technical feasibility.

Our project has been created in Python programming language and requires an internet connection and a web browser to operate properly.

Operational:

There are two parts to checking the system's operational functionality. One is concerned with technical performance, while the other is concerned with acceptance.

Technical performance refers to whether or not the system generates the proper and timely output that the end user requires. Variable inputs and outputs can be used to test the application.

This system will allow the user to submit new suspect information, examine all suspect information, add suspects to the most wanted list, locate suspects, and amend suspect information.

Economics:

The economic feasibility study is conducted to determine the project's financial viability in terms of the amount of money invested in the system and the expected output. It also includes the costs incurred during the system's development, as well as future costs for maintenance and other extraneous expenses.

The cost of development is very low because the hardware and software requirements are readily available at a low cost.

Technology Used:

Our project has been created in Python programming language and requires an internet connection and a web browser to operate properly.

One of the reasons to use python is that it is high level english like language with object oriented programming capabilities. Also, the support from the python community is quiet amazing as there are various ‘libraries’ available than can be imported within the code to be used as required.

Python allows for rapid development of software and we are going to use libraries like ‘BeautifulSoup’ and ‘Pandas’ for our Web Scraping software.

BeautifulSoup: This is a very well know python library which very good at parsing HTML content and saving it in a usable format on the user’s disk.

Pandas: This another useful python library which make sure that the data collected on the disk can presented in an understandable manner to the user making sure that the software is usable for even the most average user.

Class Diagram: