



Web Scraping Tool

(Project Proposal)

Project Code:

Group No:14

Project Advisor:

Mr.Fahad Maqbool

Project Manager:

Sir Saad Razzaq

Project Team:

- Zohaib Hassan(BSSE51F20R001)
- Moizam BilaL(BSSE51F20R016)
- Muhammad Awais Khan(BSSE51F20R032)

Table of Contents

1.	Abstract.....	3
2.	Background and Justification.....	3
3.	Project Methodology.....	3
4.	Project Scope.....	4
5.	High level Project Plan	4
6.	References.....	5

1. Abstract

The web contains a vast amount of data that is valuable for businesses and individuals alike. However, this data is often difficult to access and process manually. Web scraping tools can help to automate the process of data collection, but many existing tools are complex and difficult to use. Our project is to build a web scraping tool that is easy to use and efficient. Our tool would be able to scrape data from any website, even those with complex JavaScript code. This can be useful for a variety of purposes, such as market research, price tracking, and competitive analysis. Our web scraping tool can provide a number of academic and industrial benefits. Academically, our tool can be used to collect data for research purposes. Industrially, our tool can be used by businesses to collect data for market research, price tracking, and competitive analysis.

2. Background and Justification

Web scraping is widely used, but it can also be misused for malicious purposes. Websites have added measures to deter scraping. Still, the benefits of web scraping are significant. Existing tools like Scrapy, BeautifulSoup, Selenium, and Octoparse offer features but are often complex. We will enhance and focus on user-friendliness, anti-scraping resilience, and specialization. We will create an easier-to-use tool with a graphical interface, robustness against anti-scraping measures, and specialization for tasks like social media or JavaScript-heavy sites.

We will make sure that our work will focus on solving a real-world problem. This will help to attract users and make a significant contribution to the field of web scraping.

3. Project Methodology

To build a web scraping tool, we will follow these steps:

Define objectives: What data need to scrape? What format do we need the data to be in? Once we have defined objectives, we will start to plan your scraping process.

Identify the websites need to scrape: We will make a list of all the websites that contain the data we need. We may also need to identify the specific pages on each website that contain the data we need.

Choose a web scraping tool: There are a number of different web scraping tools available, both free and paid. We will choose a tool that is appropriate for our needs and skill level.

Learn how to use your web scraping tool: Once we have chosen a web scraping tool, we will take some time to learn how to use it.

Write scraping code: Once we have learned how to use web scraping tool, we will write the code that will scrape the data from the websites we have identified. The specific code we need to write will vary depending on the web scraping tool we are using and the websites we are scraping.

Test your scraping code: Once we have written scraping code, we will test it to make sure it is working properly. We can do this by running code and scraping some data from the websites we have identified.

Deploy web scraping tool:After,we tested scraping code and made sure it is working properly, we will deploy our web scraping tool. This may involve uploading our code to a server or making it available as a web service.

To accomplish our objectives, we will make sure that our web scraping tool is robust and reliable. Our tool should be able to handle a variety of challenges:

Anti-scraping measures:Many websites have implemented anti-scraping measures to make it more difficult for bots to access their data.We will make sure that our web scraping tool is able to bypass these measures.

Complex websites: Some websites are more complex than others. Our web scraping tool should be able to scrape data from even the most complex websites.

New technologies: Websites are constantly evolving and new technologies are being used all the time. Our web scraping tool should be able to scrape data from websites that use new technologies, such as JavaScript and AJAX.

4. Project Scope

Defining the limits of a web scraping tool project is crucial. It means deciding what the tool will and won't do. It should include things like gathering data from different websites, handling anti-scraping measures, letting users save data in various formats, scheduling tasks, and having an easy setup. On the flip side, it should avoid tasks like scraping sites needing logins, following robots.txt rules, dealing with sensitive data like banking, and in-depth data analysis. To set these boundaries effectively, understand what our users need, prioritize important features, set achievable goals, and get user feedback during development. This clarity keeps the project on track and ensures it serves users well.

5. High level Project Plan

Week 1: Scope and research

In this week we will define the scope of our project and gaining information about existing tool. What kind of data do we want to scrape? From what websites? What will we do with the scraped data?

Week 2:Installation of tools

In this week we will choose a programming languages and then install necessary libraries and tools for web scrapping

Week 3: 50% Designing user interface

In this week we will complete 50% of our user interface.

Week 4: 50% Designing user interface

In this week we will complete our remaining 50% of user interface so that it will be easy to engage by user

Week 5: Selectors finding and integration

In this week we will connect our web scraping tool to a backend system. Once we have integrated our web scraping tool with a backend system, we can start scraping data and storing it in the backend system.

Week 6: Integration

In this week we will involve identifying the HTML elements that contain the data we want to scrape. Inspecting the HTML, CSS, and JavaScript code of a web page to understand how it works and to identify the elements we need to scrape

Week 7: Iframe loading and element selection

In this week we will create web pages that can be loaded into other web pages.

Week 8: Element selection

In this week we will identify and inspect the working of element selectors

Week 9: Sending data from frontend to backend

Once you have scraped the data from the web, you need to send it to your backend. This can be done using a variety of methods, such as HTTP requests, WebSockets, or messaging queues.

Week 10: Learning python

In this week we will learn all necessary concept of python which will be necessary for web scrapping tool

Week 11: BS4

In this week we will use BS4 to parse the HTML of the web page we want to scrape. We will use CSS selectors or XPath expressions to select the elements that contain the data we want to scrape.

Week 12: Scrapping website

In this week we will work on sending a request to the website we want to scrap. Parse the HTML of the response and extract the data we want to scrape from the parsed HTML.

Week 13: Backend to frontend & Showing data on frontend

In this week we will prepare the data in a format that can be understood by the frontend. Send the data to the frontend using a method such as HTTP requests, WebSockets, or messaging queues.

Week 14: Showing data on frontend

Handle any errors that may occur while sending the data and then showing data on frontend

Week 15: Deployment

Prepare the project for deployment, ensuring all components are properly configured and optimized.

Week 16: Review

Deploy the system to a live environment, monitor its performance, conduct a project review to assess its success and identify areas for future enhancement.

6. References

1. Glez-Peña, D., Lourenço, A., López-Fernández, H., Reboiro-Jato, M., & Fdez-Riverola, F. (2014). Web scraping technologies in an API world. *Briefings in bioinformatics*, 15(5), 788-797.
2. Sirisuriya, D. S. (2015). A comparative study on web scraping.
3. Diouf, R., Sarr, E. N., Sall, O., Birregah, B., Bousso, M., & Mbaye, S. N. (2019, December). Web scraping: state-of-the-art and areas of application. In *2019 IEEE International Conference on Big Data (Big Data)* (pp. 6040-6042). IEEE.
4. Krotov, V., & Silva, L. (2018). Legality and ethics of web scraping.