DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FOURTH SEMESTER

MINIPROJECT SYNPOSIS SEPTEMBER -2023

Project title : Automatic Quiz Solver Using Web scraping

Student Names:

- **SIDDHARTH R (1DT21CS149)**
- **SUHAS PYDE(1DT21CS160)**
- **THARUN V (1DT21CS170)**
- **THEJ VENKAT P (1DT21CS171)**

Abstract:

Web scraping is the process of extracting data from web pages. Selenium is a python library that can automate web browsers and interact with dynamic web pages. To use Selenium for web scraping, you need to follow these steps. Firstly importing the selenium library and installing a driver for a browser (such as ChromeDriver).then create a webdriver object that can controls the browser. Next specify the URL of the desired web page to scrape and use the webdriver to navigate to it. Locate the elements on the web page that contain the data which is needed to scrape (such as images, links, text, etc.) using methods like find_element_by_id, find_element_by_class_name, find_element_by_xpath, etc. Next extract the data from the elements using attributes like text, href, src, etc. and store them in a variable or a file. Finally close the webdriver when you are done.

Implementation using Selenium:

The code begins by importing modules for webdriver, service, time and sound. Then, it defines a sound function that is empty for now, but could be used later. Next, it defines two dictionaries: tgvt and egvt. The tgvt dictionary contains the URL and answers for a tenth grade vocabulary game, while the egvt dictionary contains the URL and some answers for an eleventh grade vocabulary game. After that, it creates a service object with the path to chromedriver.exe, which is needed to control Chrome. It also creates an options object to customize the browser settings. Using these objects, it creates a driver object that launches Chrome. Furthermore, it imports the By module to locate elements on a web page. The code then loops over the tgvt dictionary and navigates to the URL for each question and answer pair. It waits for 5 seconds for the web page to load. It finds the element with the question text using XPath, which is a way of specifying the location of an element in an XML document. It also finds the elements with the answer options using class name, which is an attribute that groups elements with similar characteristics. It loops over the answer options and clicks on the one that matches the answer in the dictionary. This way, it completes the trivia game for tenth grade vocabulary.

Conclusion:

In this project, usage of selenium was excellently explored, which can automate web browsers and interact with dynamic web pages, for web scraping purposes and use for trivia games, news articles, images, etc. Locating the elements on a web page that contain the data needed to be scraped were also demonstrated using methods like XPath, class name, id, etc. Lastly the advantages and disadvantages of using selenium for web scraping, such as its ability to handle complex and dynamic web pages, but also its dependency on a browser driver and its slowness compared to other methods. Hence it can be concluded that selenium is a powerful and versatile tool for web scraping, but it should be used with caution and respect for the website owners and their terms of service.