Temple Course Helper (Application for helping students find courses, using C# and SQL)

Arthur Kozhevnik

Table of Contents

Project Abstract 4
High Level Requirement 4
Conceptual Design 4
Background 4
Required Resources 4

Project Abstract

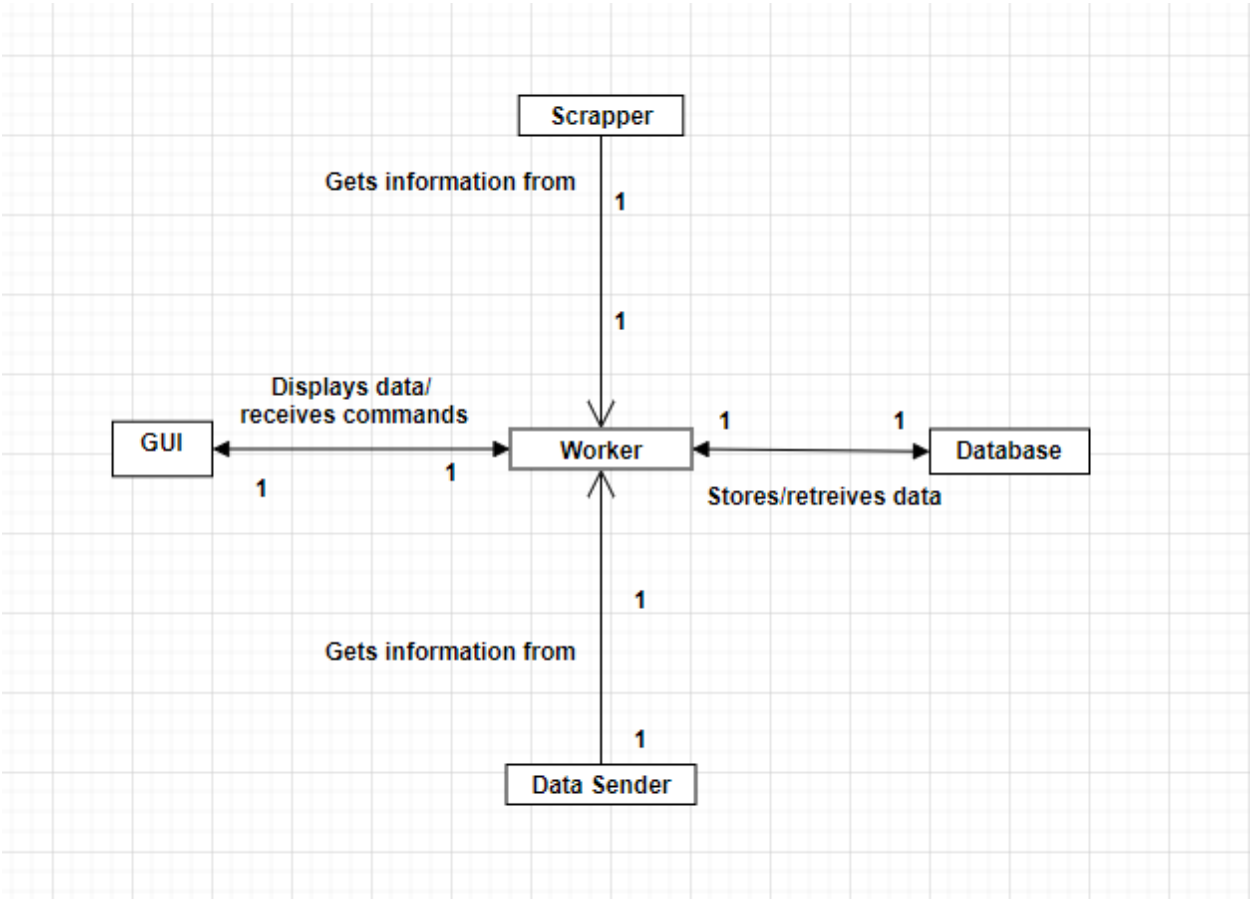
Temple Course Helper is an application that will take the users' course numbers and find the course information from the catalog on Temple's website as well as teacher ratings from Rate My Professor. The application will take all of this information and store it on a database along with the student ID entered. The database will allow returning users to open up their previous search for ease of use. On the screen, all of the information (course time, cost, professor, rating, description) will be displayed, and the user will be asked if they would like to be sent this information as well. This will help Temple students better plan their semesters and hopefully will save them time.

High Level Requirement

The application will first prompt the user if they have an already entered student ID to display their previous data. If it is a first-time user the user will enter their student ID as well as the courses they want to find. This script will use webdriver from the selenium package. This is a package that allows the script to interact with a web browser, in the case of this project the browser will be Chrome. Once the script gets online it will find the necessary xpaths that are found by inspecting the page. The script will pull the information from these xpaths, and store the info on a hashmap. The information will be stored in a different class where all general information about the course will be stored (besides the course ID). This class will be used as the value in the hashmap while the course ID will be used as the key. Following this, the script will fill in a record in a database so the user can have their information saved in case they want to revisit it. The information will be displayed on the screen, and if the user wants, they can be sent the information through a bot.

Conceptual Design

Around 4 classes should be present (main/GUI interaction class, database communication class, communication with school website class, communication with SMS/email class). Done either in C#(easier to work with GUI) or Java(more familiarity) depending on the level of comfort with the team. Knowledge of SQL commands and Access will also be useful and can be improved on with this project. How to properly use the inspect tool on browsers as well as being able to break down what is information is being given in HTML will also be a skill that will be necessary and can be improved on. We will be scraping info from the site by using xpaths. Xpaths are the codes that designate the different parts, or controls, of the UI that users see on browsers such as buttons, text boxes to type etc.



Proof of Concept

https://github.com/cis3296s22/ArthurKozhevnikProject

Background

This project has will be built from scratch with no resources taken from any open sources. However, the functionality will be similar to a search tool. If for example, you enter a course number on Temples website, you will be brought to the description of the course entered. Temples website does not have any functionality that displays multiple courses at once or sends you the information via phone, which is something that is planned for this project.

CUNYFirst is also quite similar to what I am trying to build. From my understanding, it will search globally to find a course that suites the users. The search is much more in-depth requiring a lot more information to conduct the search, but its range is quite large. Temple Course Helper on the other hand will be focused on Temple Universities Catalog and will also be displaying ratings of professors teaching those classes based on reviews from multiple websites.

https://home.cunyfirst.cuny.edu/psp/cnyepprd/GUEST/CAMP/c/COMMUNITY_ACCESS.CLASS_SEARCH.GBL?FolderPath=PORTAL_ROOT_OBJECT.HC_CLASS_SEARCH_GBL&IsFolder=false&IgnoreParamTempl=FolderPath%2cIsFolder

Required Resources

Operating System: Windows OS is required because the .NET framework runs primarily on it. Windows Forms, the way we will be building the GUI only works on Visual Studio and is not included in Visual Studio Code. Therefore without a Windows OS the only other option would be a Virtual Machine

Software:
1. Visual Studio is the best IDE for this project because of its built-in Windows Forms.
2. Chrome will be needed for the script to interact with. It will be the easiest browser to work with. Along with the browser, an add-on must be installed for the interaction to be facilitated: https://www.tutorialspoint.com/how-to-launch-chrome-browser-via-selenium
3. Selenium Package. This can be installed onto Visual Studio and will provide the library needed for the script to interact with Chrome.
4. Microsoft Access will be used to store user data for future use.
5. Twilio Package. This can be installed onto Visual Studio and will provide the library needed for the script to send a text out to the user