**Web Scraping - Senior Project**

**Website:** [**https://github.com/comp195/senior-project-spring-2023-web-scraping**](https://github.com/comp195/senior-project-spring-2023-web-scraping)

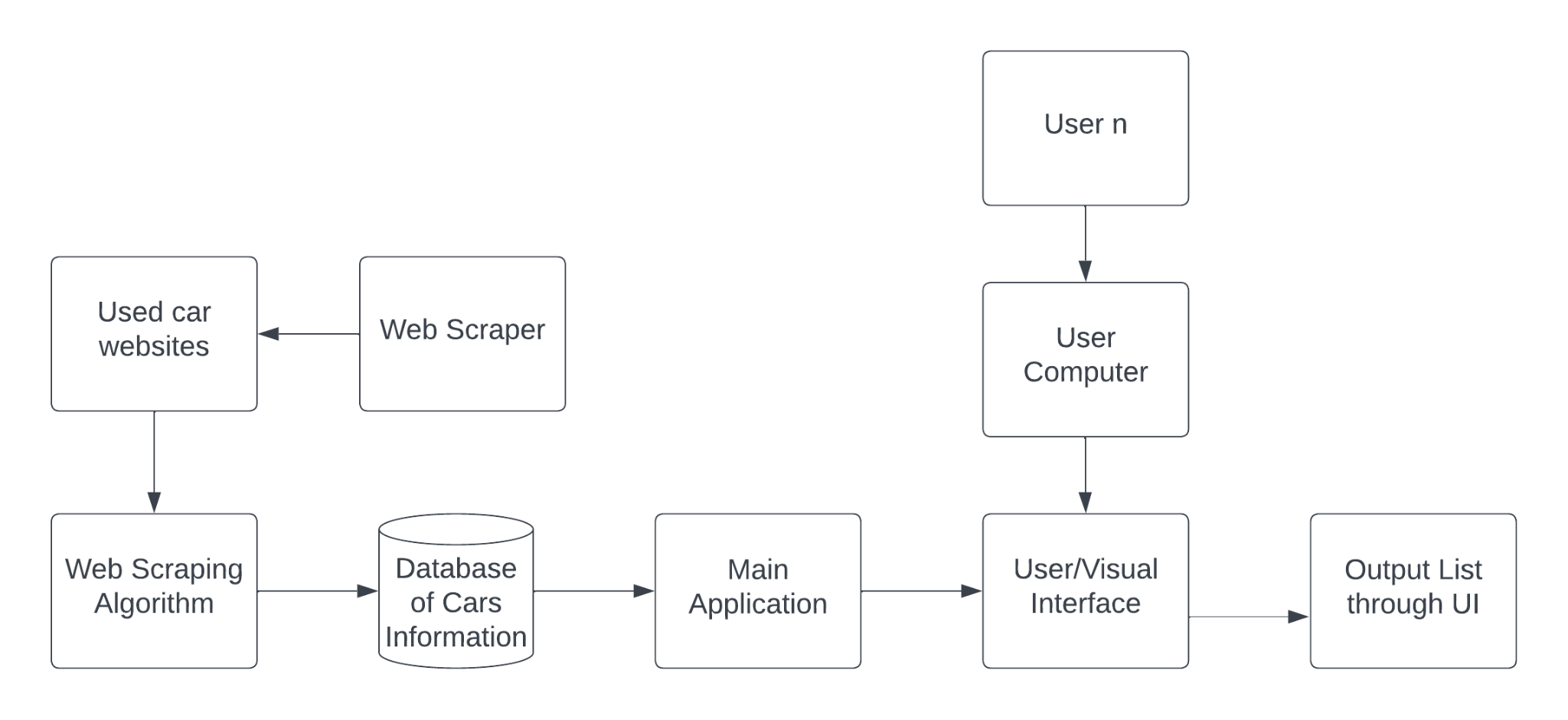
**Bryan Herrera:** [**b\_herrera6@u.pacific.edu**](mailto:b_herrera6@u.pacific.edu)

**Tran Hoang:** [**g\_hoang2@u.pacific.edu**](mailto:g_hoang2@u.pacific.edu)

**Sloan Kim:** [**g\_kim22@u.pacific.edu**](mailto:g_kim22@u.pacific.edu)

**Last revision: 2/5/2023**

**System Architecture**

****

**Hardware, Software and System Requirements**

* Hardware Requirements: internet connection
* Software Requirements: Python 3.10.4, selenium 4.8.0, MySQL (version), Cloud Server to run database
* System Requirements: Windows 10/11

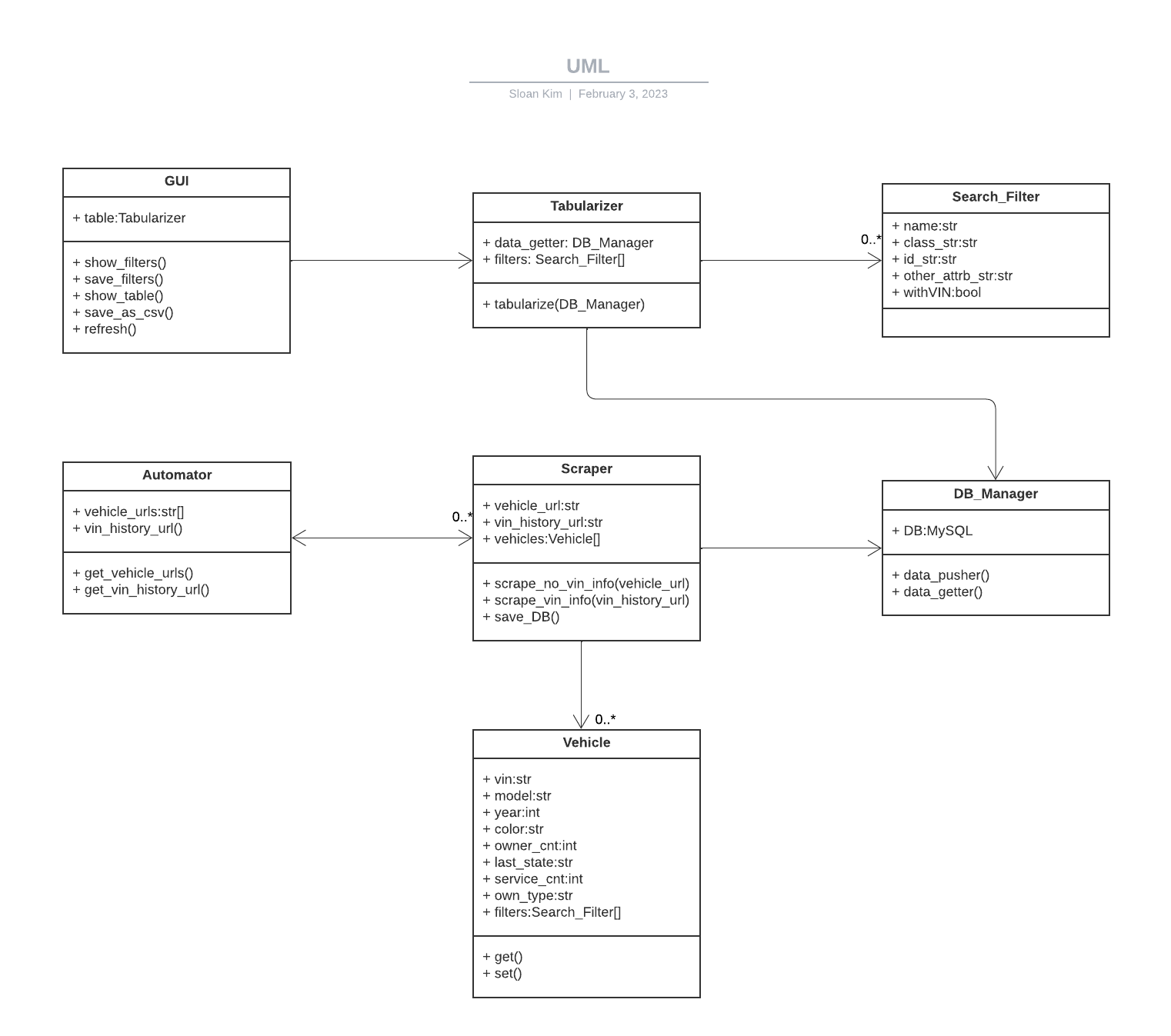
**External Interfaces**

There will be communication with 2 external interfaces, the first being the database that will hold all the car data and information, and the second which will be the used car websites which we will be collecting data from.

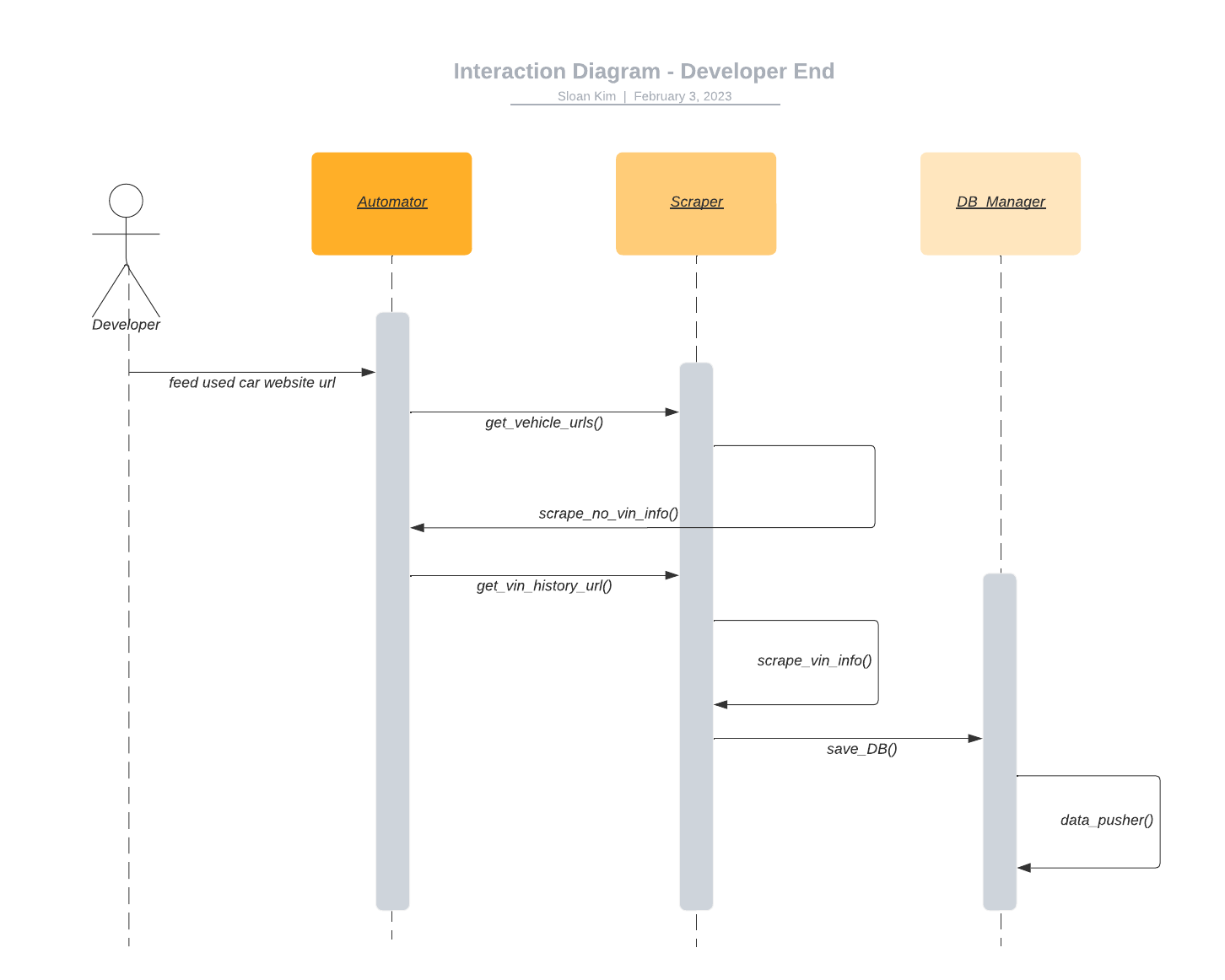
* MySQL Database
  + Since a MySQL database will be used we will have to install the MySQL driver
    - This will allow us to import the MySQL connector to python and integrate it into our code
    - Will allow us to add data into the database using proper authentication
* Used Car Websites
  + Selenium will allow us to get information from URLs that are provided based on VIN
    - The automator will connect us to those sites and get filtered information
  + Filtered information will then be directed back into the python program to parse

**Software Design**

* **Class Diagram**

****

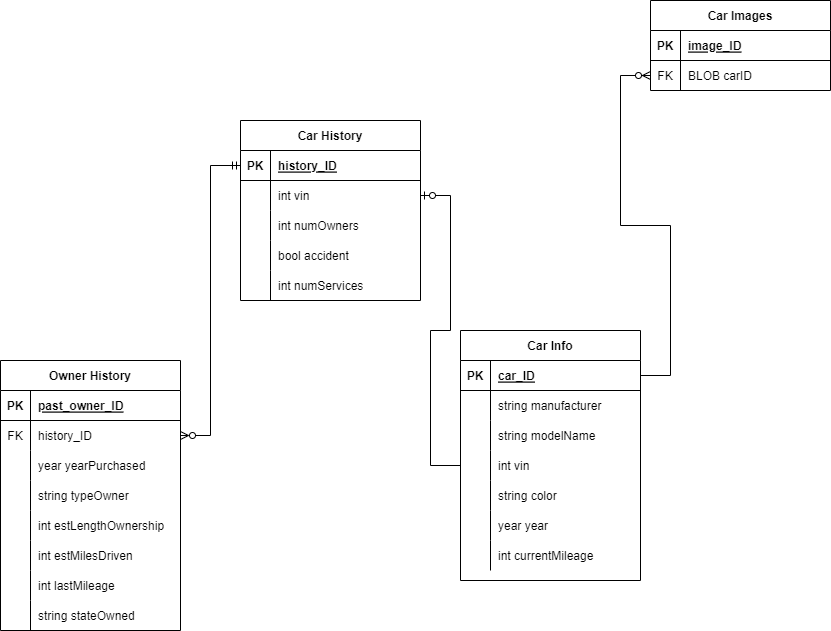
* **Interaction Diagrams**

****

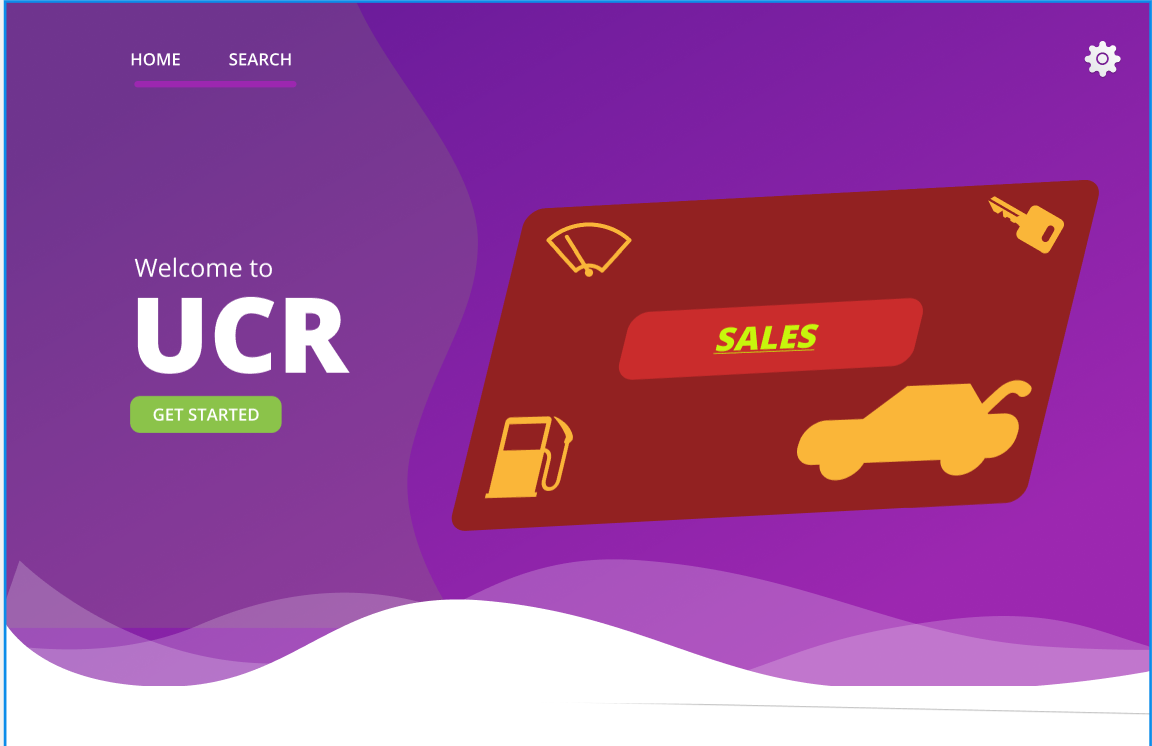
* **Design Considerations**

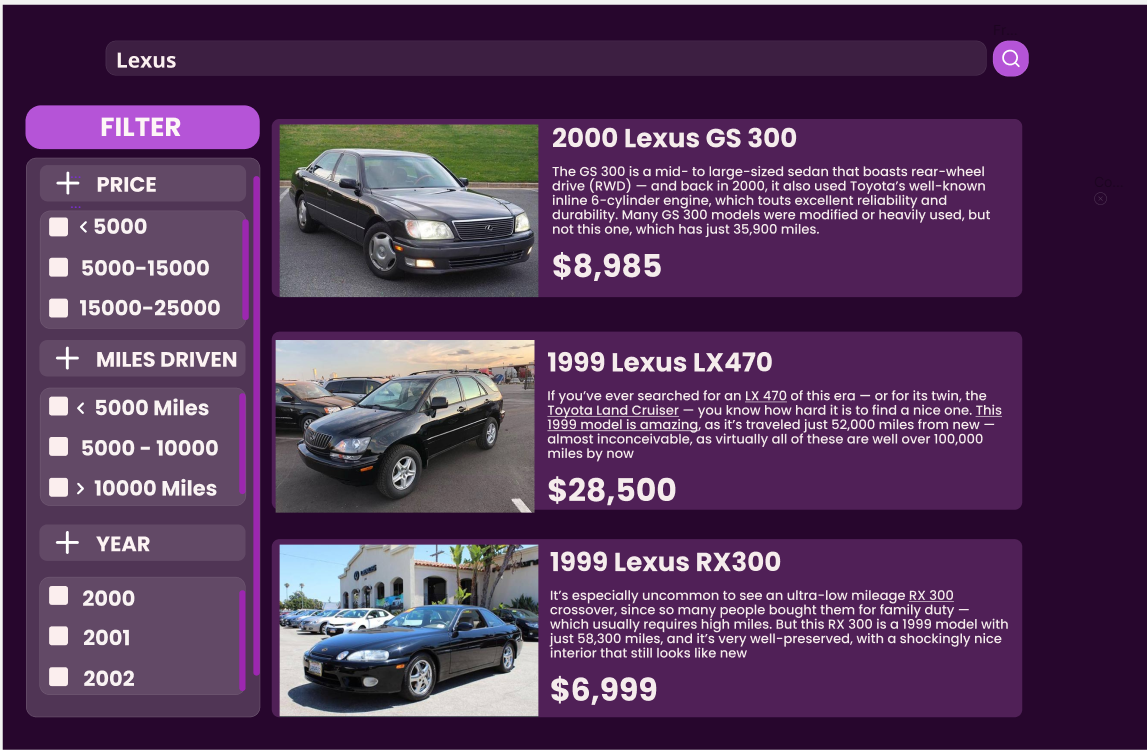
The Interface Segregation Principle was considered in the design of classes. Users should not be forced to depend on interfaces they won’t use. The GUI class that is exposed to the user is designed to contain only user related methods.

* **Grant Model**

****

**User Interface Design**

****

****

**Glossary of Terms**

* Web-scraping: fetching a website to extract necessary information
* Selenium: an open source automation testing tool; this project uses its package in Python to automate the search and to scrape the search result
* Database: an organized collection of data stored and accessed by software
* Authentication: credentials that allow us to connect to the database
* Xpath: XML path language; a flexible way of addressing different parts of an XML document
* HTML class: an attribute specifying class names for HTML element
* HTML id: an attribute specifying a unique id for an HTML element
* WebElement: a Selenium object representing HTML element
* Selenium WebDriver: a web framework to control browsers by scripts
* Selenium Wire: a package giving access to the underlying requests from browser

**References**

Anon. Selenium-wire. Retrieved February 3, 2023 from https://pypi.org/project/selenium-wire/

Anon. 2. getting started¶. Retrieved February 3, 2023 from https://selenium-python.readthedocs.io/getting-started.html

Anon. WebDriver. Retrieved February 3, 2023 from https://www.selenium.dev/documentation/webdriver/

*MySQL Connector/Python Developer Guide :: 5.1 connecting to MySQL Using Connector/Python*. MySQL. (n.d.). Retrieved February 5, 2023, from <https://dev.mysql.com/doc/connector-python/en/connector-python-example-connecting.html>