|  |
| --- |
| Google Reviews Scraper  Deliverable 2 |

|  |  |  |
| --- | --- | --- |
| Group 1 |  |  |
| Monique Dai/ Chester Rae de Vera/ Heping Song  Snehalata / Hua Zhang / Yateng Geng / Chester Rae de Vera/ Heping Song |  |  |



# 1. scraper's architecture

**1.1 Basic Architecture**

**a. Front-End:**

1. Allow users to input the place name for which they want to scrape reviews;
2. Allow users to check reviews directly as Json list or download reviews as csv file;
3. Giver user guidance, i.e. when user input the place name that has no reviews found, tell them “We found no reviews for this place, please check your input and try again.”

**b. Back-End:**

1. When a user submits a place name, the back-end retrieves the place ID using the Google Maps Places API.
2. If a place ID is found, the back-end uses the place ID to scrape the reviews from the Google Maps website using web scraping techniques.
3. The scraped reviews are processed and formatted into a JSON list or CSV file.
4. If no reviews are found for the specified place, a message is returned to the front-end indicating that no reviews were found.

**c. Data Output:**

1. The output data are stored in a specific folder.
2. The output csv file is named after the user input, trimmed and adding “Reviews” at the end of the file name.
3. If the scraper doesn’t find anything with the user input, there won’t be any output data generated.

**1.2 Data Flow and Components Required for Scraping Google Reviews**

**A picture containing text, screenshot, font, number

Description automatically generated**

# 2. Core Functionality

**2.1 Command-Line Interface with csv file output (limited reviews)**

**A screenshot of a computer program

Description automatically generated with medium confidence**

**A screenshot of a computer

Description automatically generated with medium confidence**

Please try our test\_CLI.py file for this function. It will show which place id it’s associated with and the URL you can double check if it’s the right place you are searching for. After exiting you could check the reviews scraped. But due to google API limitation the reviews are limited to 5.

**2.2 HTML Interface with Json output(limited reviews)**

**A screenshot of a computer

Description automatically generated with medium confidence**

**A screenshot of a computer

Description automatically generated with medium confidenceA screenshot of a computer screen

Description automatically generated with low confidence**

Please try our test\_jason\_frontend.py file for this function. You could input a place name and the reviews would be showing as Json list as in the picture. But due to google API limitation the reviews are limited to 5.

**2.3 HTML Interface with Both output format(All reviews, but not done yet)**

**A screenshot of a computer program

Description automatically generated with medium confidence**

Please try our main.py file for this function. In order to scrape all reviews we are using selenium and BeautifulSoup. Now it seems we could use the place name to associate with google ID and google URL for the place use input, but the scraper can’t scrape the reviews yet.

# 3. set up

3.1 Replace 'Your\_API\_Key' with your real google API key for testing in all python files.

3.2 Install Firefox brower, and the matched geckodriver from <https://github.com/mozilla/geckodriver/releases>. You could use test\_selenium.py to check if you installed a matched version. If so when you run it, you will see it open a Firefox browser window on google (it will be closed quickly).

Please check our repo for more details: **https://github.com/raeDV/Google-Reviews-Scraper**