

# Lead Generation Software

This project is a Lead Generation Software that allows you to search for leads based on specific keywords and filters. The software consists of three main files: **crawler.py**, **output.py**, and **gui.py**.

To run the project from Command prompt, follow these steps:

Extract the rar file into the folder. Open that folder and click on the url bar and write cmd and hit enter.

1. Install the required libraries by running the following command in Command prompt:

```
pip install -r requirements.txt
```

2. Run the **gui.py** file to start the GUI:

```
python gui.py
```

3. In the GUI, enter the keywords and filters you want to search for in the respective entry fields.
4. Click the "Search" button to start the search. The search results will be displayed in the text area below the button.
5. To export the search results to a CSV file, click the "Export to CSV" button. After this it might take time to go to all pages and extract the information, after that a popup will appear, indicating that the CSV file has been exported successfully. The CSV file will be saved in the same directory as the **output.py** file.

## **crawler.py**

This file contains the **crawl** function, which searches Google using the provided keywords and filters, and returns a list of URLs found in the search results. The **export\_to\_csv** function is also defined here, which takes the list of URLs and the GUI keyword as input, extracts contact details from the URLs, and writes the details to a CSV file.

## **output.py**

This file contains the **write\_to\_csv** function, which writes the contact details to a CSV file. It takes a list of dictionaries as input, where each dictionary contains the extracted contact details for a single lead.

## **gui.py**

This file contains the graphical user interface (GUI) for the Lead Generation Software. It uses the **tkinter** library for creating the GUI and the **crawl** function from **crawler.py** for searching leads.

The **crawl** function in **crawler.py** performs the following steps:

1. Constructs the Google Custom Search API URL using the provided keywords and filters.
2. Sends a GET request to the API URL and retrieves the search results.
3. Extracts the URLs from the search results and returns them as a list.

The **export\_to\_csv** function in **crawler.py** performs the following steps:

1. Iterates through the list of URLs and sends a GET request to each URL.
2. Parses the HTML content of each URL and extracts the contact details (names, addresses, phone numbers, and email addresses).

3. Stores the extracted contact details in a list of dictionaries.
4. Writes the list of dictionaries to a CSV file using the **write\_to\_csv** function from **output.py**.

The **write\_to\_csv** function in **output.py** performs the following steps:

1. Opens the CSV file in append mode.
2. Writes the header row to the CSV file.
3. Iterates through the list of dictionaries and writes each dictionary as a row in the CSV file.
4. Closes the CSV file.

The **gui.py** file creates the following GUI elements:

1. A label and entry field for the keywords.
2. A label and entry field for the filters.
3. A button for starting the search.
4. A label and text area for displaying the search results.
5. A button for exporting the search results to a CSV file.

When the "Search" button is clicked, the **search\_keywords** function is called, which retrieves the keywords and filters from the entry fields and calls the **crawl** function from **crawler.py**. The search results are then displayed in the text area.

When the "Export to CSV" button is clicked, the **export\_to\_csv\_button** function is called, which retrieves the GUI keyword from the keywords entry field and calls the **export\_to\_csv** function from **crawler.py**. A popup is then displayed, indicating that the CSV file has been exported successfully.