**Project Name: Influencer Byte**

**Team (C3-Group 7):**

Jenny Niemann (jmnieman@andrew.cmu.edu)

Darshil Pandya (darshilp@andrew.cmu.edu)

Abhirup Panja (apanja@andrew.cmu.edu)

Kavyaka Pellakuru (kpellaku@andrew.cmu.edu)

**About Project:** The goal of influencerByte is to recommend marketing options to an individual customer for a chosen product on some of the famous marketing platforms.

**Sources Used:**

**Youtube:** We have scraped the details of videos from YouTube using Selenium. The python scripts search for the product reviews on YouTube and scrolls down the page 15 times to ensure that we have enough data. We did not scroll beyond 15 as it increased the time required for scraping the data. From every row of result, we are extracting the video title, view count, age of the video, description, and name of the creator. This data is then stored in a clean format in the csv file which is used as a data source.

**Twitter:** We have scrapped tweets from twitter using tweepy, a python library to access Twitter API. The Python script extracts tweets from twitter based on a particular hashtag input for the required product, and collects user profile specific data like username, tweetText, number of followers etc. to give an insight into the potential of the user as an influencer. To begin with, we have restricted the tweet data to about 300 latest tweets out of which the top influencers are picked based on the highest number of followers for a user.

**Patreon/Stitcher:** We have used Selenium and Beautiful Soup to scrape the top podcasts from Patreon and Stitcher. The python scripts search for the podcasts related to the product that we are attempting to advertise for. It loads the preliminary search page, then clicks the top ten podcasts listed and extracts the name of the podcast, description, number of subscribers, and monthly revenue generated by the show. We use this data to recommend the top influencer and use a basic equation to estimate the potential revenue that could be generated from advertising with a podcast show.

**Solution Design:**

**Tools Used/Software Requirements:**

* Python 3.6 and above
* Spyder
* Jupyter Notebook
* Anaconda Prompt IDLE
* Tkinter
* Selenium
* BeautifulSoup

**Files/Modules submitted:** The influencerByte GUI folder contains all the files necessary to run the project. The files are listed below:

influencerByte\_GUI>

influencerByte\_GUI.py

patreon\_cleaned.xls

patreon\_laptop.py

patreon\_mobile.py

patreon\_tv.py

patreon\_watch.py

podcast\_laptop.py

podcast\_mobile.py

podcast\_tv.py

podcast\_watch.py

twitter\_cleaned.xlsx

twitter\_laptop.py

twitter\_mobile.py

twitter\_tv.py

twitter\_watch.py

youtube\_cleaned.xlsx

youtube\_laptop.py

youtube\_mobile.py

youtube\_tv.py

youtube\_watch.py

chromedriver.exe

DataGatheringFiles>

Podcast>

patreon\_cleaned.xls

patreon\_links.py

raw\_data.txt

stitcherSelenium.py

Twitter>

data.csv

json1-csv.xlsx

json1.txt

json2-csv.xlsx

json2.txt

json3-csv.xlsx

json3.txt

twitter\_cleaned.xlsx

YouTube>

twitterRaw.com.har

youtube\_cleaned.xlsx

youtube\_scrape.py

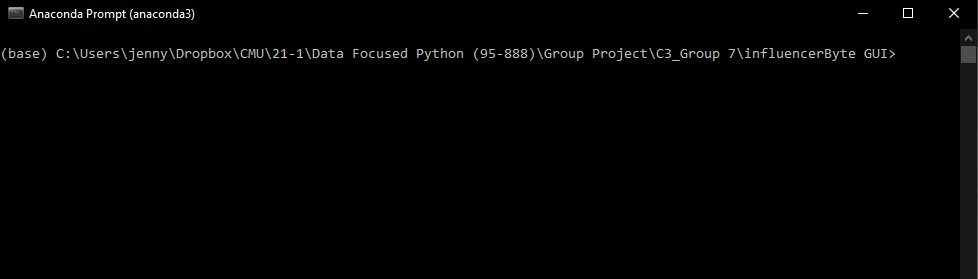
**How to install and run ?**

Ensure before running that Python 3.6 or above is installed. Using the anaconda prompt, confirm Tkinter is installed. (If not, enter: conda install -c anaconda tk) Additionally, confirm selenium is installed. (If not, enter:conda install -c conda-forge selenium) Lastly, ensure beautiful soup is installed. (If not, enter:conda install -c anaconda beautifulsoup4)

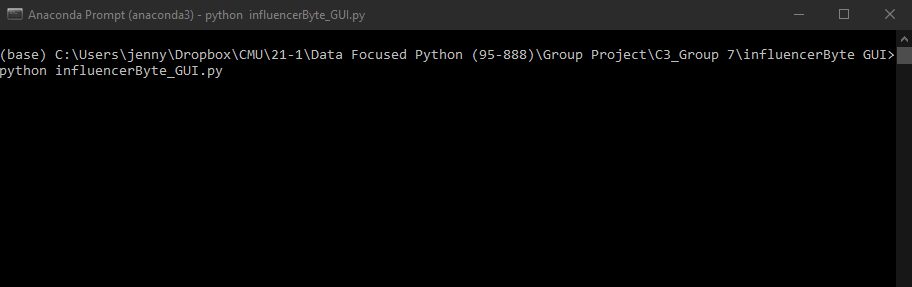
Download the C3\_Group 7 zip file.

If there is an error with the static podcast data scraping, the user should re-download the appropriate ChromeDriver for their version of Google Chrome. (At https://chromedriver.chromium.org/downloads) The chromedriver.exe needs to be saved into the influencerByte\_GUI folder.

Navigate to the C3\_Group 7 >influencerByte\_GUI folder from the anaconda prompt.



Run the program with the command pythoninfluencerByte\_GUI.py



The influencerByte GUI will pop up on the screen and all further user interactions will be completed with the GUI. The outputs from the program will display on the anaconda terminal.

*Alternatively, the C3*\_*Group 7 >influencerByte\_GUI >influencerByte\_GUI.py file can be opened in idle and run to achieve the same as above.*

**What new to Install?**

* Tkinter
* Selenium
* BeauitfulSoup
* Google Chrome

**Link to Video:**