**Proposer Details**

| Group Number | *65* |
| --- | --- |
| Registration Number of Group Members | **2020-CS-83** |

**Proposal Details**

|  |  |
| --- | --- |
| ***Project*** |  |
| Proposed Project Title | **Twitter Squeaker** |
| Executive Summary | Every second, on average, around 9,685 tweets are tweeted which corresponds to over 580,000 tweets sent per minute and 800 million tweets per day. This makes Twitter an excellent place to get data for projects, as Tweets are an accurate representation of today’s natural language on social media.  Twitter data can be used for research, machine-learning, sales leads, etc. Now a days most news is extracted from twitter. News channels or representatives can use Squeaker to get daily news related to different categories in a single file and assort them with respect to re-tweets.  While working on a project a developer can check the recent trends related to his project or programme and estimate the expected response to his program and check whether he should continue his work on the project or not.  Sports enthusiast can retrieve information of their favorite teams, tournaments or players at one place. Similarly, many games or books enthusiast and a common man would be able to access information and save it in their device with the help of Squeaker.  Total entities of the file to be used would be:   1. Username 2. Handle 3. Time Stamp 4. Comments 5. Likes 6. Re-tweets 7. Text 8. Category   The user can sort information according to two or more options given above, i.e Multi-level searching and multi-column searching would be allowed.  For web scraping **selenium web driver** would be used and the **browser** to be used is **MS-Edge**. To access elements form the web page , **x-path** and its resources would be used.  The files would be saved as .csv files. A separate folder would be made for all of the csv files. The name of file would be saved by the time stamp, scraping was done at, however it could be changed by the user, according to their preference.  For the UI, pyqt would be used. This program would have its own login interface, to access the app. The user would be required to give their twitter account information to login to twitter, and start with the web scraping. |
| ***Business Case*** |  |
| Outline the business need for the project | * *Use this tool to scrape tweets and replies and use this as away to connect with society* * *Gaining background Knowledge on a target audience* * *Following latest Trends* |
| End user of the product | 1. *Marketing and advertising* 2. *Social Media Influencers* |
| Motivation for Project | *Using social media for constructive work* |
| State the level of impact expected should the project proceed and implications of not proceeding | 1. *For marketing and sales would save them time for research* 2. *Influencers would gt to know their followers better and they can design their content for their majority audience* |
| ***Technical Details*** |  |
| Name of Entity | 1. Time Stamp (Merge Sort) 2. Number of Likes (Insertion Sort) 3. Re-tweets (Bubble Sort) 4. Category (Selection Sort) |
| Attributes of Entity  (Minimum seven attributes/rows can be increased) | |  |  |  | | --- | --- | --- | | *Name* | *Data Type* | *Description* | | Time Stamp(year) | datetime | It can be sorted for a specif year/ form latest to old/old to latest | | Time Stamp(date) | datetime | According to specific date/form a specific date/before a specific date | | Time Stamp(month) | datetime | In a month every year | | Time Stamp(time) | datetime | At specific time of day throughout different years | | Time Stamp(date/month/year) | datetime | Latest/ oldest | | Number of Likes | integer | Maximum/Minimum | | Number of Re-Tweets | integer | Maximum/Minimum | | Category | string | According to a specific name of category | |
| Sample of Scrapping Source | *twitter* |
| Github Repository Link | *https://github.com/LaibahAzhar/CS261F21PID65.git* |
| **Sorting Algorithms** |  |
| |  |  | | --- | --- | | **Algorithm Name** | **Description(Each algorithm in 2-3 lines)** | | Merge Sort | All the time stamp entities would be sorted by using quick sort | | Insertion Sort | Number of Likes would be sorted by Insertion sort by comparing the number of likes of each tweet | | Bubble Sort | Re-tweets of each tweet would be compared and sorted with the help of bubble sort | | Selection Sort | Categories of different tweets would be sorted with selection sort | | Quick Sort | Multi-level sorting of different entities together would be done using quick sort | | |
| Searching Algorithms | 1. *Sequential Search* 2. *Binary Search* |
| Searching Filters for each data type | 1. *Time :* 2. *By year* 3. *By Month* 4. *By Date* 5. *By Time* 6. *Number of likes:* 7. *Re-tweets* 8. *Category* |
| Multi-Level Sorting | On clicking filter button they would have the option to fill the check boxes all with filled check boxes would be used to sort at a multi-level |
| Any other features | 1. The information is given as a preview first and it would only be saved if the user enters the save button 2. After 30 minutes of inactivity the user would be asked to login again 3. A secure folder for all the saved files of the user, so that they are only accessible by the user. |
| ***Interfaces for your project*** |  |
| *[Draw layouts in the pencil tool. For each picture of the UI, provide the following table.]*   |  |  | | --- | --- | | Use Case ID | *U01* | | Name | ***Login/Registration*** | | Actor | *User* | | Description | *The user can register their account. Once registered they can use their respective username and password to login into their account. While registering the user would be asked to login into their twitter account.* | | Layout in pencil tool | LoginRegistration | |  |  |  |  |  | | --- | --- | | Use Case ID | *U02* | | Name | ***Menu*** | | Actor | *User* | | Description | *User would be given different options to chose from* | | Layout in pencil tool | filter buttonsave filtertwitter1Search Filter | |  |  |  |  |  | | --- | --- | | Use Case ID | *U03* | | Name | ***Squeaker*** | | Actor | *User* | | Description | *User would be shown their filtered and sorted search of twitter information* | | Layout in pencil tool | Squeaker | |  |  | | Use Case ID | *U04* | | Name | ***Filter/ Sorting*** | | Actor | *User* | | Description | *User will set his filters, save them and apply them on his research* | | Layout in pencil tool | Filter | |  |  | | |