# General Overview

This purpose of this document is to outline the functionalities of this project and give users a guide as to how to use it.

Our program imitates a mini application of a social media platform where users can follow other users creating a platform of followers for each user. There will be two menus shown for the user, a login/registration page, and a main menu where the user can interact with tweets. After successful login or registration, a list of tweets will be provided. The user can then 1. select one of the tweets to interact with, 2. show more tweets or 3. enter 0 to see the main menu. The menu is where users can select certain functionalities including 1. viewing and interacting with tweets from users they follow, 2. search for tweets, 3. search for users, 4. compose tweets, 5. list followers and 6. log out.

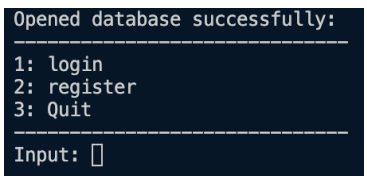
## User Guide

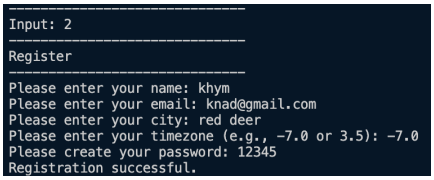
- 1. View and interact with tweets from users they follow: Brings user back to the initial screen where latest tweets from followed users are shown
- 2. Search for tweets: Searching for tweets allows the user to input one or more key words and all tweets containing the word or words will be provided, the user can then choose to reply to the selected tweet or retweet it themselves.
- 3. Search for users: A keyword is inputted and all users whose name contains the keyword will be listed in ascending order of name length, this is followed by users who's city contains the keyword and these users are listed in order of ascending city name length. The user can choose to see five more users, or the remaining amount from their search, and subsequently see the five previous if they had previously chosen to see five more. The user can also select a user to view data about that user during which they also have the option of following the user.
- 4. Compose a tweet: A user is able to compose a tweet which can contain hashtags that will add the tweet to the mentions table of the specific hashtag it contains
- 5. List followers: The user can view all of the users who follow them through this functionality. Through list followers they are also able to select a follower and see data on the selected user and have the option of following the user or seeing more tweets from the user.
- 6. Logout: allows the user to log out of the system

# Design

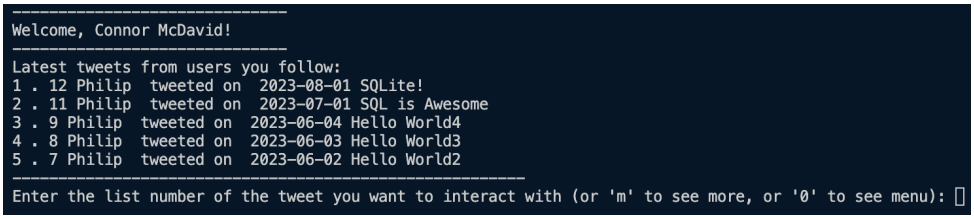
As per the project, the user will first be prompted with a database to use once the program starts to run. When the database is verified, the program will prompt the user with a login or registration menu.

- Registration: If the user decides to register, they would be prompted to input their name, email, city, timezone, and password. With each input, the program would ensure that all of the user inputs were applicable for registration. Name would only include characters from the alphabet, email would require a domain with an @ sign, city would only include characters from the alphabet, timezone would only include characters that would be turned into a float, and password would accept anything. If registration was successful, the user information would be inputed into the database and the new user would be automatically login into the platform. The new user would then be prompted with the main menu.
- Login: User is prompted to to 1: login, 2: Register, 3: Quit. If a user selects login they are prompted to enter a user ID followed by a password, if they are valid inputs, the user will be logged in and sent to the menu page.
    - def login_page(): is the main function that is used to evaluate the login request. It evaluates the input, either being 1, 2, or 3 as valid inputs and anything else as invalid and respectively handles each valid input

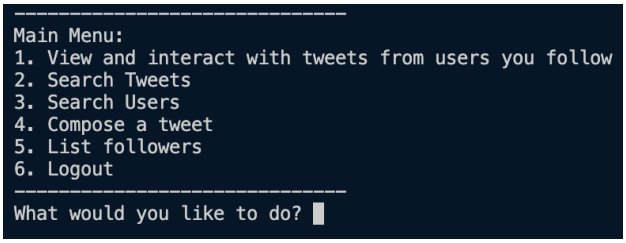





- Recent Tweets Menu: The Recent Tweets Menu displays five of the most recent tweets from other users they follow. The user is prompted to view and interact with a specific tweet by inputting the number listed beside the tweet as well as options to view more tweets by inputting ‘m’ or going to the functionalities menu by inputting ‘0’
    - def show_tweets(user_id) is the main function for displaying the recent tweets after successful login or registration




- Functionalities Menu: The Functionalities Menu allows users to access six functionalities. View and interact with tweets from users they follow, Search Tweets, Search Users, Compose a tweet, List followers, Log out.
    - def menu(user_id) is the main function for displaying the main menu




- Search Tweets: Prompts the user to input in a key word and provides a list of tweets, five at a time. The user can exit the beginning search for tweets by not inputting anything and pressing enter. The user can see the next five tweets by inputting ‘n’ the previous five tweets, if they exist, by inputting ‘p’, and ‘q’ to quit.
    - def search_tweets(user_ID) is the main function for searching tweets
- Search Users: Prompts the current user to input in a key word and provides a list of users, five at a time. The user can exit the beginning search for users by not inputting anything and pressing enter. The user can see the next five users by inputting ‘n’ the previous five users, if they exist by inputting ‘p’, and ‘q’ to quit.
    - def search_users(user_id) is the main function for searching for users
- Compose a tweet: Prompts the user for tweet text. Once the tweet has been made, a new tid will be made for that tweet, inserted into the database and will determine if there are any hashtags made within the tweet. If the hashtag does not exist within the preexisting hashtags table, update the table and update the mentions table
    - def compose_tweet(user_id, replyto) is the main function for composing a tweet
- List followers: Prompts the current user for the id of a existing user. Once an existing user is inputted, the number of tweets, number of followed accounts, and number of followers will be displayed. The current user will then be prompted to follow the user. Regardless if the current user says ‘yes’ or ‘no’ they will then be prompted to see the user’s tweets.
    - def list_followers(user) is the main function for listing followers

- Logout: The current user will be sent back to the login/register page where they can decide action from there.
    - def menu(user_id) is the main function. Logout is in the menu function which allows the user to go back to the login/register page.

# Testing Strategy

Our general strategy for testing for any bugs would be to constantly go through the code and test different cases with the inputs to make sure all the functionalities were working properly. We set breakpoints between functions that were called within one another to determine how values that were passed were altered in order for proper output to be provided. Here are some of the common bugs we ran into:

- **Input bugs**: A few testing strategies to test for bugs in any of the inputs of the program, would be to try many different characters. Characters that were capitalized, number, special characters, etc. In many cases when inserting data into the database, certain inputs would not work.
- **Pre Existing data bugs**: To test for bugs for inserting data into the database that might already have the same preexisting data, the INSERT function would have to include an OR IGNORE function beside it. If a input was already within the database this would allow to program to keep running and keep what was initially within the database.
- **Proper function calls**: Some functions would return to different sub functions instead of returning to the proper one.
- **Out of index bugs:** For the see more and see previous tweets, there were bugs when it came to using the right index.
- **Foreign or unique key constraints:** Incorrectly inputting a value for SQL queries so that the keys were not consistent.
- **SQL commit bugs:** If any commits were not properly implemented for the database, a bug would occur.
- **Putting code together (variable bugs):** When putting together each of our parts, some of the variables were different. There were some implementation issues that caused bugs when consolidating code.

# Group work strategy

The group work for the project was split up between, the login screen and functionalities. Since the login menu deals with view and interact with tweets, whomever was doing the login menu would be handling that function as well. Then the other 4 functionalities would be split up between the other two members. One member would handle the functions for search for users and compose tweet, while the other would handle the functions for search for tweets and list followers.

**Khym Nad:**

Responsible for: Search for users and compose tweets
Time allocated: 20 hours

**Samuel Chan:**

Responsible for: Login menu and search for tweet function
Time allocated: 20 hours

**Andrew Zhang:**

Responsibile for: Search for tweets and list followers
Time allocated: 20 hours

Our main method of coordination was to meet up on campus and share the work we had done for each of our responsibilities. As well as that, we would fix each others code to make sure the functions we were responsible for would work with one another.