Design Document: Mini Project 1

Database Experts

System overview & User Guide

This program is a mockup of a Twitter application. Users can perform some of the actions found on twitter, such as the making and searching of accounts and tweets, following users, and other such functions. Load up the database and follow the on-screen instructions to traverse the application.

To start the application, install main.py into the repository with the database used, and run the command `python3 main.py <database file name>` in the repository.

Once started, the program can be interacted with through the terminal. The user is able to navigate through the different screens/actions in the application by inputting the numerical value corresponding to the next screen/action. A menu will be displayed each time an action is required showing the corresponding number value for the possible options.

The application consists of the following screens:

```
|–Login
    |--Home Screen
        |--Tweet Search
                |--Tweet Selected
                        |--Send Reply
        |--User Search
                |--User Selected
        |--Send Tweet
        |--List Followers
                |--User Selected
```

Login: The user is first greeted with the login screen when the application is started. From here you can: 1:sign up, 2:sign in, or 3:exit the program
Home Screen: After signing up/ signing in, the user arrives at the home screen. The home screen presents the 5 most recent tweets and retweets made by users followed. Then the following actions can be made: 1:search for tweets, 2:search for users, 3:send a tweet, 4:list followers, 5:logout, 6:list more tweets (lists 5 more), 7:back to top (show most recent tweets).
Tweet Search: Allows users to search for tweets containing keywords. The tweets found are displayed and the user can: 1-5:select a tweet out of the 5 tweets shown, 6: go back to home screen, 7: show more tweets.
Tweet Selected: Displays information about the tweet selected. The user can: 1: reply to the tweet (goes to send tweet screen), 2: retweet the tweet, 3: go back to tweet search.

User Search: Allows users to search for users with username containing keywords. The users found are displayed and the user can: 1-5:select a user out of the 5 users shown, 6: go back to home screen, 7: show more users.
User Selected: Displays information and 3 most recent tweets of the user selected. You can: 1: follow the user, 2: see more tweets from the user, 3: go back to user search.
Send Tweet: Allows you to type in a tweet, sends the tweet using the account logged in. Goes back to the home screen after completion.
List Followers: Displays a list of your followers (5 shown). You're given the option to: 1-5: select the follower out of the 5 shown, 6: go back to home screen, 7: show more followers

Software Design & Components

Primary Functions: main(), signup(), login(), actions(usr), tweet_search(usr), select_tweet(uid, tid), user_search(usr), select_user(u1, u2), send_tweet(usr, replyto_tid=None), follower_list(usr)
Secondary Functions: connect(path), get_flwer_tweets(usr), get_searched_tweets(keyword_list), retweet(uid, tid), follow_usr(u1,u2), print_5_rows(table, start_index, numbered=False), print_3_rows(table, start_index, numbered=False)

Description of primary functions:
main(): Calls connect(path) to connect to the database. Runs loop for the login screen until user exits the program. User inputs: 1: calls signup() 2: calls login() 3: exits program

signup(): Requests user input for: name, email, phone, password. Inserts a new user into the users table, "logs in" by calling actions(user_id).

login(): Requests user input for: user_id, password. Checks if there is a match in the database, and if so, "logs in" by calling actions(user_id)

actions(usr): Is called after successful login from login() or signup(). Lists the tweets and retweets of followed users obtained from get_flwer_tweets(usr). Runs a loop that requests user input for the next action until user "logs out"
user input:

1: calls ACTION 1: tweet_search()
2: calls ACTION 2: user_search()
3: calls ACTION 2: user_search()
4: calls ACTION 2: user_search()
5: ACTION 5: logs out by exiting form actions() and returning to login screen in main()
6: see more tweets of followed users, only appears if there are more tweets
7: go back to the top of the homepage, showing the most recent tweets

tweet_search(): Handler for ACTION 1. Requests user input for: keywords, a string of comma separated keywords. Calls get_searched_tweets() to get a list of tweets containing keywords. Exits the function if no tweets are found. If tweets are found: runs a loop requesting user input for the next action, until user enters a valid choice.
user input:

1 - (number of tweets printed): select a tweet from the list of tweets printed (5 or less), calls select_tweet(uid,tid)
6: back to actions()
7: see more tweets, only appears if there are more tweets

select_tweet(uid, tid): Displays the retweet count and reply count of the tweet selected. Runs a loop requesting user input for the next action until a valid choice is entered.
user input:
1: calls send_tweet(uid,tid) to reply to the tweet (insert a new reply tweet to tweets table)
2: calls retweet(uid,tid) to retweet the tweet (insert a new row to retweet table)
3: exits the function

user_search(usr): Handler for ACTION 2. Requests user input for a keyword. Retrieves a list of users with username containing the keyword. Exits the function if no users are found. If users are found: runs a loop requesting user input for the next action, until user enters a valid choice.
user input:
1 - (number of users printed): select a user from the list of users printed (5 or less), calls select_user(u1,u2), with u1 being the uid of the current session and u2 the uid of the selected user
6: back to actions()
7: see more users, only appears if there are more users

select_user(u1, u2): Displays the tweet count, following count, and follower count of the user selected. Displays the top 3 most recent tweets made by the user. Runs a loop requesting user input for the next action until a valid choice is entered.
user input:
1: calls follow_user(u1,u2), which inserts a row into follows table
2: show 3 more tweets if there are more
3: exits the function

send_tweet(usr, replyto_tid=None): Handler for ACTION 3. Requests user input for the tweet text. Checks if the tweet is valid (no repeat hashtags allowed). Inserts the tweet into the tweets table and any hashtags in the tweet into the hashtags_mentions table.

follower_list(usr): Handler for ACTION 4. Gets a list of followers of usr. Exits the function if no followers are found. If followers are found: display 5 of the followers. Runs a loop requesting user input for the next action until a valid choice is entered.
user input:
1 - (number of followers printed): select a follower from the list of followers printed (5 or less), calls select_user(u1,u2), with u1 being the uid of the current session and u2 the uid of the selected user
6: back to actions()
7: see more users, only appears if there are more users

Testing Strategy

Created a new test database with more entries to test if various aspects of the application worked correctly, such as the many parts of the application where a list is printed in several parts.

Test scenarios involved performing various actions within the application and ensuring the program behaves as expected. These include:

- Following a user and then returning to the main menu
- Performing various actions (making tweets, following users) before logging out & logging back in
- Viewing tweets and followers when there are more than 5 entries to be shown
- Testing various search queries in the different search functions

The test cases performed should theoretically cover any series of actions a user might perform in the application.

Bugs found:

- around 3 instances of incorrect SQL queries causing issues within the application
- 3+ instances of needing to account for incorrect user input
- some modifications to match project specifications
- occasional typos and print spacing issues

Group Work Break-Down Strategy

The project was first split into separate functions that could each be worked on independently. Each group member could then develop the separate functions as needed. Communication over clarifications and other coordination between group members was primarily over Discord.

Member	Tasks	Time spent (hrs)
Vero Bullis	print_5_rows helper function tweet_search and get_searched_tweets send_tweet debugging documentation	~9
Victor Kwok	code comments user_search, select_user, follow_user, follower_list improved print format select_tweet, retweet documentation	~12
Taran Purewal	Created boilerplate functions initial login code initial signup initial handling of login follower tweets/retweets	~7
Krish Rajani	follower_list, user_search signup various fixes/changes	