Y

General Overview

User Guide

- Make sure all the dependencies are installed.

```
pip3 install prettytable
```

- Enter the name of the database in the terminal.
- At each screen, a toolbar is shown at the bottom of available actions.
- Choose whether you want to log in or sign up
- Once you login/sign up your feed will be displayed, showing 5 tweets at once.
- You can choose to scroll through your tweets, showing up to 5 tweets at once.
- You can also search for users based on their name and city
- You can also search for tweets based on the hashtags they mention and the words they contain, which will be drawn from all the tweets in the database.
- You can also compose tweets, reply or retweet, which will be shown to your followers
- You can also follow people, whose tweets will be shown to your feed.

Detailed Design

In our design, we have one main App class, that contains an instance of the connection to the database. Within this App class, we have methods that handle all of the specified behaviour that was required.

Users are able to search for tweets with either hashtags, or the tweet text ordered by the latest tweets. Each tweet can be viewed with its number of retweets and replies. Users can also be searched either by their name or city, sorted by length. Searches are outputted in set of five. User profiles can be seen with their stats.

Our app has a user-friendly interface that is made by curses.

Testing Strategy

We began testing our code with the general functionalities like signing up and logging in. We tried to log in with incorrect passwords to make sure the login was rejected.
We used the general functionality of the program with the expected cases. Then we moved on to the edge cases. We tried to test as many edge cases as we could and improvised the code to deal with edge cases.

Software Design

Our system is divided into 3 classes Tweet, User and App. The Tweet class is used for storing and manipulating information related to tweets. The objects of the User class are used to represent users. The App class implements most of the functionality of the program. The main.py file takes care of the user interface and user arguments.

The main.py file is responsible for the user-facing aspect of our application. We have chosen to employ the curses library to design an interactive and text-based user interface that enhances the user experience. This ensures that users can seamlessly navigate the application, view tweets, explore user profiles, and interact with the system's various features.

To present the tweets and user information in a structured and visually appealing manner, we have employed the PrettyTable library. This choice enables us to organize and display data in a clear and logical format.

Group Work break-down strategy

Below is a list of all the method and file each group member worked on.

- Abdul: signup, tweet, retweet, reply, mention and hashtag
- Ahnaful: get_user_search, show_tweet, show_user
- Bassam: select_user, select_tweet
- Uzair: login, logout, User class, Tweet class, get_feed_tweets, get_search_tweets, get_user_tweets, get_followers, main.py

We brainstormed the design of the app together and Uzair led down the blueprint for all the classes and methods. Then we divided the implementation of different methods according to our strengths and weaknesses and what the individual members wanted to work on.