

Capstone 2: Twitter Clone - Project Overview

Front End - Twitter Clone:

1. Authentication:

- Implemented a user-friendly login page and registration page.
- Collected user data such as full name, email, username, and password for registration.
- Validated login credentials (email or password) against the backend.

2. User Interface:

- Designed a sidebar with links to the home page, profile page, and a logout option.
- Organized the sidebar with the home page featured first.
- Included a tweet list in the second section, showcasing tweets from the user and others.

3. Tweet Interaction:

- Integrated tweet cards allowing users to like, reply, and retweet.
- Implemented the ability to delete the user's own tweets while restricting deletion for others.
- Provided a detailed view of tweets, including replies and the option to reply on the same page.

4. Profile Page:

- Displayed user details like profile picture, date of birth, join date, and location.
- Enabled users to upload a profile picture.
- Implemented features to edit personal information such as name, date of birth, and location.
- Showcased the user's tweets and replies on the profile page.

5. Logout Functionality:

- Implemented a logout link redirecting users to the login page.

Back End - Twitter Clone:

1. Data Models:

- Created two models: one for users and another for tweets.
- User model schema includes fields for name, username, email, password, profile picture, location, date of birth, following and followers' arrays, tweets, and timestamps.
- Tweets schema includes content, likes, retweets, images, replies, and timestamps.

2. User API Endpoints:

- Developed APIs for user registration, login, user details retrieval, following/unfollowing users, user profile editing, fetching user tweets, and updating profile pictures.

3. Tweet API Endpoints:

- Implemented APIs for creating tweets, liking/disliking tweets, replying to tweets, retrieving tweet details, fetching all tweets, deleting tweets, and retweeting.

4. Token Authentication:

- Utilized JSON Web Tokens (JWT) to generate tokens upon user login.
- Validated tokens on the backend to ensure the user's authenticity.
- Stored the user ID in **req.user** after successful validation.

5. Image Handling:

- Integrated Multer for handling image uploads, enabling users to upload tweet images and profile pictures.

By combining a robust front-end user interface with a well-structured backend, the Twitter Clone project allows users to seamlessly navigate, interact, and share their thoughts in a familiar social media environment.