

## **Title:** Twitter Clone Backend using Node.js

### Project Description:

#### **Overview:**

The Twitter Clone Backend project using Node.js is a server-side implementation of a simplified version of the popular social media platform Twitter. This project focuses on developing the core functionality of Twitter, including user authentication, posting tweets, following users, and retrieving user timelines. While this project only covers the backend, it serves as a foundational component for a full-fledged Twitter clone application.

### Key Features:

#### 1. User Authentication:

- Implement user registration and login functionalities.
- Securely store user credentials and session management.

#### 2. Tweet Management:

- Allow users to create, edit, and delete tweets.
- Store tweets in a database for retrieval.

#### 3. User Following/Followers:

- Enable users to follow and unfollow other users.
- Maintain a list of followers and following for each user.

#### 4. User Timelines:

- Generate user timelines with tweets from users they follow.
- Implement a feed that shows tweets in chronological order.

#### 6. Search Functionality:

- Allow users to search for other users and tweets based on keywords and hashtags.

## Technologies Used:

**Node.js:** The primary backend framework for building the server.

**Express.js:** To create RESTful API endpoints and handle HTTP requests.

**SQLite:** SQL is used as the primary data storage and retrieval mechanism database for storing user data, tweets, and user relationships in this project. It plays a pivotal role in storing and organizing various types of data, including user profiles, tweets, follower/following relationships, and more.

## Project Workflow:

1. Database Design: Design the database schema to store user information, tweets, followers, and other relevant data.
2. User Authentication: Implement user registration and login using JWT tokens for authentication.
3. Tweet Management: Develop endpoints to create, read, update, and delete tweets.
4. User Following/Followers: Create endpoints to allow users to follow and unfollow other users.
5. User Timelines: Implement a timeline generation algorithm to fetch tweets from followed users and display them on the user's timeline.
7. Search Functionality: Develop search endpoints for users and tweets.
8. Testing: Write unit tests and perform thorough testing to ensure the functionality is working as expected.
9. Documentation: Create clear and concise documentation for API endpoints and project setup.

## Project Goals:

- Develop a scalable and efficient backend for a Twitter clone.
- Practice building RESTful APIs and handling real-time functionality.
- Gain experience with user authentication and data storage.

- Create a foundation for a potential full-stack Twitter clone in the future.

Note: This project represents the backend of a Twitter clone and can be extended in the future to include a frontend using technologies like React.js, Angular, or Vue.js to provide a complete user experience.