Design Document for "Twiddle"

1. Introduction

Project Name: Twiddle
Purpose: To develop a microblogging social media platform where users can post short messages (tweets), interact with others' tweets, and participate in groups.
Goals: Create a secure, user-friendly, scalable, and responsive platform supporting real-time interactions and a rich feature set.

2. Background

Twiddle is inspired by the popular microblogging service, Twitter. It aims to provide a platform for users to share short posts (tweets), interact with others' content, and participate in groups. The app will include features to enhance user engagement and discoverability, such as suggested users and groups.

3. Requirements

Functional Requirements

- User registration via email or other accounts like google and github and a secure authentication process using a strong and robust third-party service.
- Tweet creation, and deletion.
- Viewing tweets from other users.
- Like, comment, retweet, and share tweets.
- Sidebar navigation for different pages.
- Pages: Home, Search, Notifications, Tweet, Groups, Profile.
- Group creation and invitation functionality.
- Right sidebar for suggested users and groups.
- Responsiveness across all screen sizes.

Non-Functional Requirements

- High availability and scalability.
- Secure data handling and privacy protection.
- Fast response times.
- Intuitive, beautiful, and responsive user interface.

4. Architecture

Overall Architecture

The system will follow a microservices architecture to ensure scalability and maintainability. Key components include:

Server Actions: Responsible for handling client requests, performing business logic, and interacting with databases or other services.

Authentication Service: Manages user login and registration.

User Service: Handles user profiles.

Tweet Service: Manages creation, retrieval, and deletion of tweets.

Notification Service: Show notifications to users when an activity happens on their profile.

Search Service: Enables searching for users, and groups.

Group Service: Manages group creation, membership, and content.

Frontend: Web and mobile interfaces.

5. Detailed Design

Authentication Service

Third-Party Service: use a third-party service for secure authentication.

Server Actions:

- `POST /auth/register`: Register a new user.
- `POST /auth/login`: Authenticate user.

User Service

Database Schema:

- `User`: id, username, name, onboarded, image, bio, tweets. retweets, likedTweets, replies, groups.

Server Actions(user actions):

- `GET /users/{id}`: Retrieve user profile.
- `PUT /users/{id}`: Update user profile.

Tweet Service

Database Schema:

`Tweet`: id, text, author, retweetOf, group, createdAt, parentId, children, likes

Server Actions (tweet actions):

`POST /tweets`: Create new tweet.

`GET /tweets/{id}`: Retrieve tweet.

`POST /tweets/{id}/like`: Like tweet.

`POST /tweets/{id}/comment`: Comment on tweet.

`DELETE /tweets/{id}`: Delete tweet.

`POST /tweets/{id}/retweet`: Retweet.

Group Service

Database Schema:

`Group`: id, username, name, image, createdBy, tweets, members

Server Actions (group actions):

- `GET /groups/{id}`: Retrieve group.
- `POST /groups`: Create new group.

Search Service

Server Actions:

- `GET /search`: Search for users, and groups.

Notification Service

Server Actions:

- `GET /notifications`: Retrieve notifications for user.

6. User Interface (UI)

Wireframes

- **Home Feed**: Displays a feed of tweets from users and groups.
- **Profile Page**: Shows user's profile information and tweets.
- **Tweet Creation**: Interface for creating a new tweet.
- **Notifications**: List of notifications for user activity.
- **Groups**: List and manage groups.
- **Search**: Search interface for users, and groups.

Interaction Flows

1. **User Registration**:
   - User enters details → System validates input → User account is created → Redirect to home feed.
2. **User Login**:
   - User enters email and password → System authenticates user → Redirect to home feed.
3. **Posting a Tweet**:
   - User clicks 'New Tweet' → Enters content → Clicks 'Tweet' → Post is saved and displayed on home feed.
4. **Liking a Tweet**:
   - User views a tweet → Clicks 'Like' → System registers like → Like count is updated.
5. **Commenting on a Tweet**:
   - User views a tweet → Clicks 'Comment' → Enters comment → Clicks 'Post' → Comment is saved and displayed.
6. **Retweeting**:
   - User views a tweet → Clicks 'Retweet' → Confirms action → Retweet is saved and displayed.
7. **Deleting a Tweet**:
   - User views their own tweet → Clicks 'Delete' → Confirms action → Tweet is removed.
8. **Navigating Pages**:
   - User uses sidebar → Clicks on Home, Search, Notifications, Tweet, Groups, or Profile → Page loads corresponding content.
9. **Creating a Group**:
   - User navigates to Groups page → Clicks 'Create Group' → Enters group details → Clicks 'Create' → Group is created.
10. **Inviting to Group**:
    - User views group → Clicks 'Invite' → Enters email of invitee → Clicks 'Send Invitation' → Invitation is sent.
11. **Viewing Suggested Users and Groups**:
    - User logs in → Right sidebar displays suggested users and groups based on interactions and interests.

11. Deployment

Deployment Process

- **Development:** Local environment setup with NodeJS and Next JS.
- **Production:** Deploy to production utilizing Vercel for deployment.

12. Security Considerations

- **Authentication**: Use Clerk for secure user authentication.

13. Appendices

Glossary

- **Tweet**: A short message posted by a user.
- **Like**: Express approval of a tweet.
- **Retweet**: Share another user's tweet.
- **Comment**: Respond to a tweet.
- **Group**: A collection of users with shared interests.

14. Technology Stack

Based on the requirements, we will use Next.js, Clerk for authentication, TypeScript, MongoDB, and shadcn/ui, among other technologies, to bring this project to life. The technology stack for Twiddle is determined based on the requirements and constraints outlined in this design document. Considerations include scalability, security, ease of development, and community support.