Twitter-clone App Project

High-Level Design Document

Group C4:
ZHENG, Xinhao (1155143819)
YI, Jian (1155157207)
LI, Yuk Ting (1155159914)
SHI, Juluan (1155160208)
ZHANG, Xue (1155160250)

Department of Computer Science and Engineering
The Chinese University of Hong Kong
CSCI3100: Software Engineering
February 4, 2023
Version 1

Table of Contents

1 INTRODUCTION

1.1 Project Overview

In this project, we will build a cross-platform social networking tool called ”Twitter”, which aims at providing users with a platform to interact with others. Firstly, users can sign-up/ login to their accounts and share short posts, photos, or videos with their followers. Users can also like/ dislike, leave comments, or even retweet tweets. To make our software more user-friendly, we keep the user interface clean and tidy. Apart from the basic functions, we also add some other awesome functions to our ”Twitter”, including private chats, anonymous tweets, tweet tags, and a bug report entry.

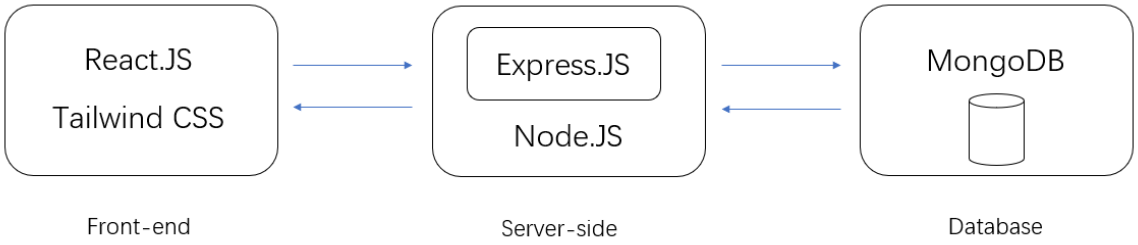
1.2 System Features

In regard of the system features, we provide general features to normal users, such as posting, liking, and sharing of Tweets, as well as private chatting and profile customization. Additionally, we provide advanced features to admin users, for the purpose of easy management of all the users. Detailed features for different kinds of users are listed below:

- Features for General Users
  - User Registration and Login
  - Posting and Deleting user’s Tweets
  - Adding a searchable Tags to a Tweet
  - Setting the Visibility of Tweets among Followers
  - Anonymous Tweets
  - Liking/ Disliking, Commenting, and Sharing of Tweets
  - Searching for Users
  - Following other Users
  - Seeing other users’ Tweets
  - Private Chats
  - Profile Customization
  - Reporting Bugs
- Features for Admin Users
  - Listing all Users
  - Deleting Users when necessary

2 SYSTEM ARCHITECTURE

2.1 Technologies



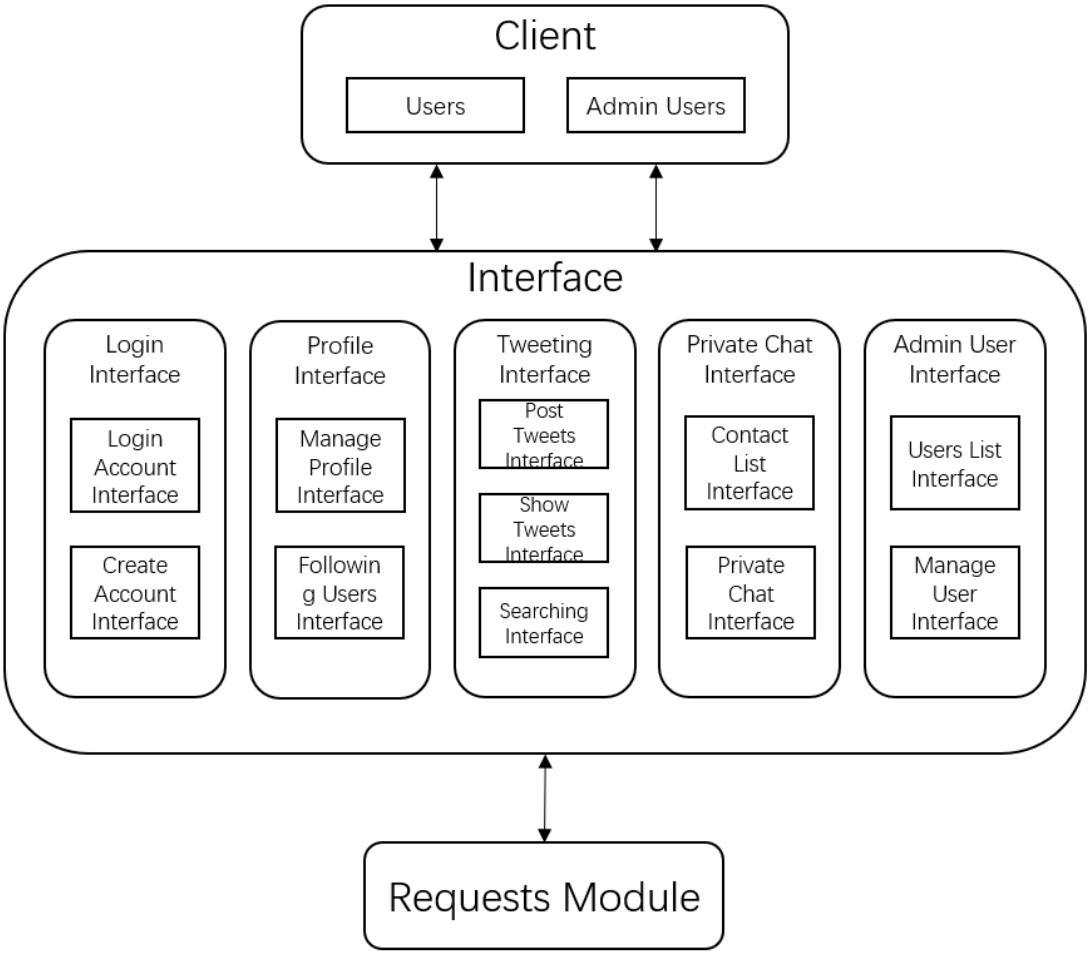
We plan to use the MERN stack (MongoDB, Express, React, Node) as the Tech Stack to build our software.

As for the front end, we choose React.js as the JavaScript framework to build the user interface. React.js can help us make dynamic website pages more conveniently. Besides, we will use the Tailwind CSS library to beautify our UI components. As for the back end, Node.js works as the running environment for Express.js. Express.js is the server-side framework that can help handle the HTTP requests sent by the client side. To handle the client-side request, the server end may need to fetch data from MongoDB and send it back to the client side. Besides, the hidden logic also happens on the server end. Generally, Express.js is responsible for logically manipulating data and works as the bridge between the front end and the database. MongoDB is a NoSQL database that stores data in JSON format. It stores users' accounts, tweets, and other related information.

- Summary
  - Database: MongoDB
  - UI: React.js, Tailwind CSS
  - Server-end Environment: Node.js
  - Server-end Framework: Express.js
  - Programming Language: JavaScript

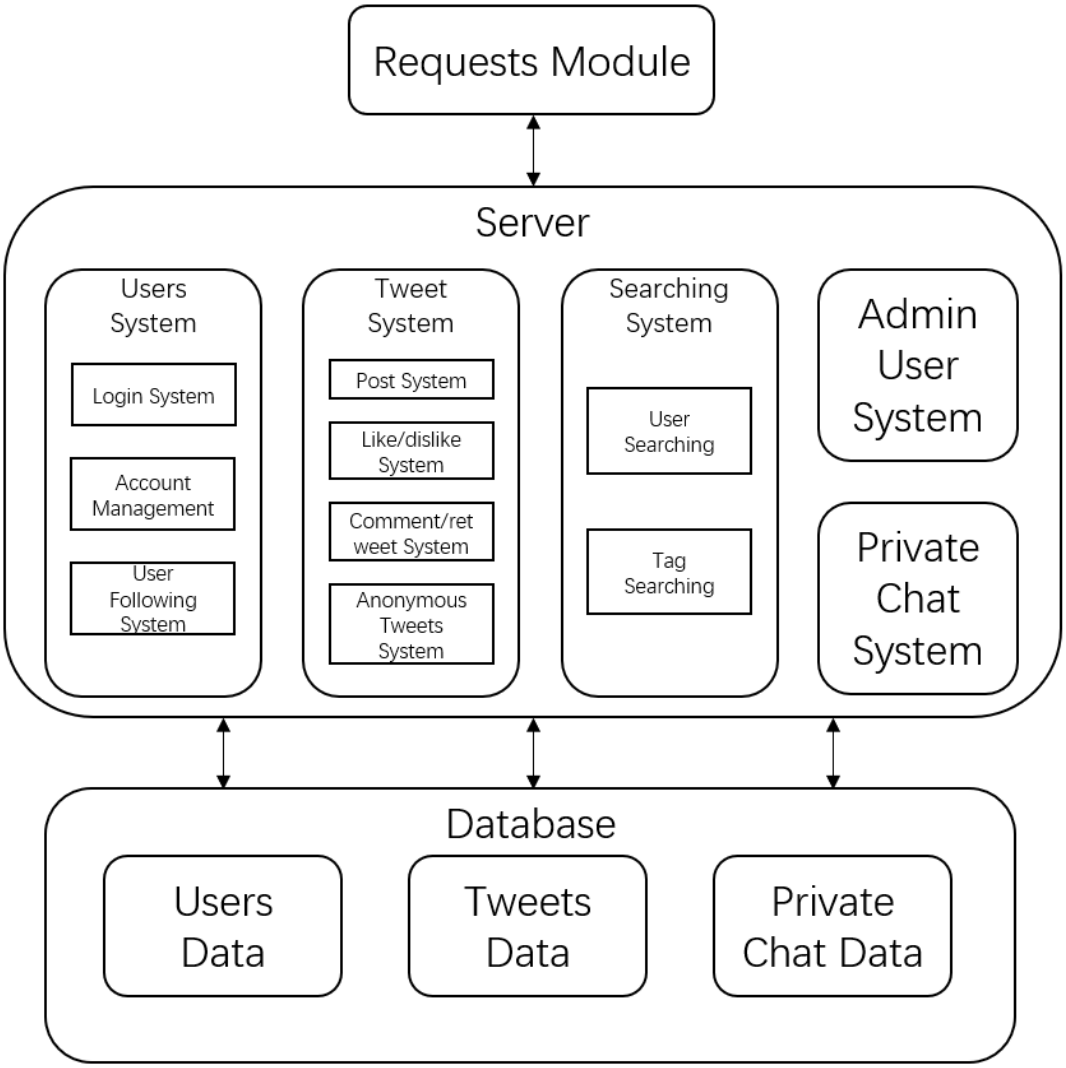
2.2 Architecture Diagram

2.2.1 Front End



This diagram shows the client-side of our Twitter application. The front-end provides users with an interface to interact with the system. Requests for content and operations from clients will be sent to the server.

2.2.2 Back End



This diagram shows the server side of our Twitter application. Requests from users are received by the server, which supports various functions for the users in collaboration with the database. After registering, users can edit and post their Tweets, view others' posts, set up their profiles, manage their accounts and followings, and chat privately with other users.

2.3 System Components

2.3.1 User System

- Login System

  Registered users can log into their accounts, while unregistered users can register in the login interface. Additionally, registered users can recover their accounts in this system.

- Account Management System

  Users can edit their profiles, change account passwords or log out their accounts.

- Following System

  Users can manage their followings, such as adding, deleting, or starring their following accounts. Additionally, a user can view the list of people who follow them in this system and check followers' information by clicking their head portraits. Users will be informed of the updates of their following accounts here.

2.3.2 Tweet System

- Post System

  Users can type a Tweet including text, photos, a GIF, or a video into the compose box and post it.

- Like/ Dislike System

  When viewing others' Tweet, users can give thumbs-up or thumbs-down to the Tweet.

- Comment System

  Users can post comments under the Tweet they are viewing.

- Anonymous Tweet System

  Analogous to tweet system, but users' personal information will be hidden.

2.3.3 Searching System

- User Searching System

  Users can search for a certain user by ID number or nickname.

- Tag Searching System

  Users can search and check contents related to one particular tag.

2.3.4 Private Chat System

- Users can send private messages to other users.

2.3.5 Administrator System

- Administrators can review the content that common users post and decide whether to publish it to the public. Additionally, admin users can lock or suspend accounts that release harmful content.