

## **High-Level Design Document Outline**

Project Title : Simplified Twitter (Tentative)

Document Version Number : 1.0

Printing Date : 25/1/2023

Group ID : E3

Group Members :

TSE Hui Tung, 1155158864

CHEUNG Kwong Tai, 1155142517

NG Man Tik, 1155158302

TAI Long Kwan, 1155152117

LAI Chuen Fung, 1155144433

Department : Department of Computer Science and Engineering

University : The Chinese University of Hong Kong

## Table of Content

1. Introduction	
1.1. Project Overview	p.3
1.2. System Features	p.3
2. System Architecture	
2.1. Technologies	
2.1.1. Database	p.4
2.1.2. User Interface (UI)	p.4
2.1.3. Programming Language	p.4
2.2. Architecture Diagram	p.5
2.3. System Components	
2.3.1. Profile System	p.6
2.3.2. Post System	p.6
2.3.3. Search Engine	p.6
2.3.4. Follower System	p.6
2.3.5. Chat System	p. 6
2.3.6. User management System (Admin)	p. 6

# **1 Introduction**

## **1.1 Project Overview**

In this project, we aim to create a streamlined version of Twitter, a state-of-the-art online communication and media-sharing platform. The platform allows users to publish text-based posts, along with multimedia content such as photos, GIFs, and videos, referred to as "tweets." Regardless of their connection to the poster, other users can engage with these tweets through actions such as "liking," "retweeting," and "commenting." As a result, Twitter has become a vital tool for disseminating information, fostering communication, and promoting social activism by enabling users to connect and interact with one another. Furthermore, users can follow other users who align with their interests, allowing them to receive the most relevant and engaging tweets.

## **1.2 System Features**

The Simplified Tweeter platform consists of three crucial components: tweet interaction, user interaction, and tweet display. In terms of tweet interaction, users are offered the ability to express their opinions and thoughts on a post using "like" or "dislike" buttons or by leaving comments, promoting communication and the exchange of ideas between users. The platform also enables users to share tweets from other users by "retweeting" them to their followers, thereby amplifying the dissemination of information.

The user interaction component is equally vital in Simplified Tweeter. Users can follow their friends and stay updated on their tweets by searching for their Twitter handle or discovering them through their tweets. If users want to share more personal thoughts and ideas, they can make their tweets private by setting them as "protected", which limits viewing to only their followers.

Finally, the Simplified Tweeter features a main page that displays all tweets from the users' followers and offers an aesthetically pleasing user interface (UI) that enhances the user experience. To further enhance the user experience, the platform utilizes a recommendation algorithm that prioritizes tweets that are most relevant to the user's interests and from favored users, allowing users to quickly find and engage with the most captivating tweets on the platform.

## **2. System Architecture**

### **2.1 Technologies**

#### **2.1.1 Database**

The NoSQL database MongoDB will be employed in our application. To fulfill Twitter application functionality characteristics, which included read-heavy and write-heavy functionality, support for multimedia for viewing and posting tweets, and permission control for the public or specified followers. The preliminary database schema will include (a) account collections, which define user identity and authority; (b) follower collections, which include followers and followees of a user; and (c) tweet post collections, which store all content elements of a tweet, including the poster, the poster's avatar, date, text, view, like, and comment.

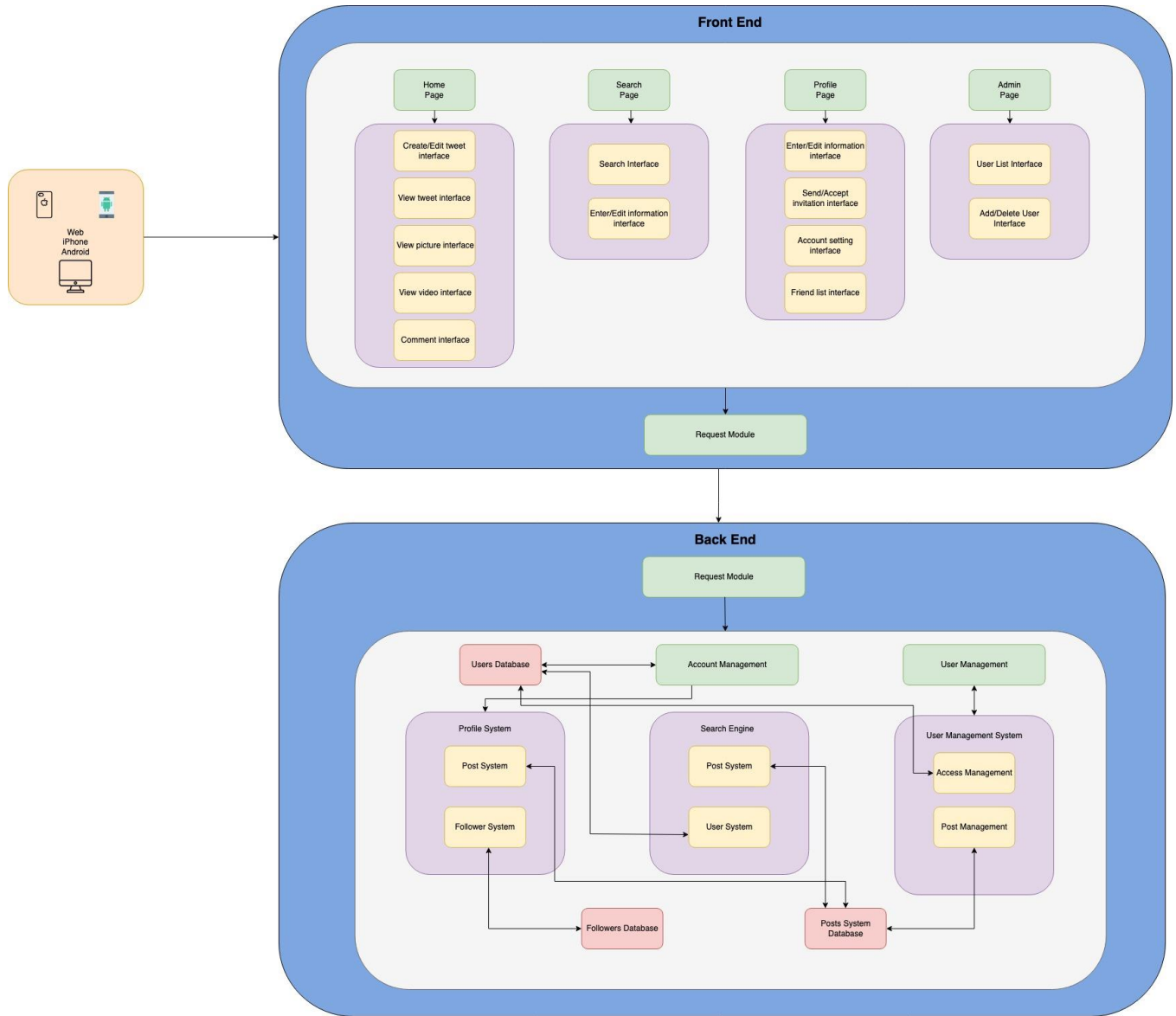
#### **2.1.2 User Interface (UI)**

The basic user interface (UI) design consists of three separate columns for the sidebar, feeds, and widgets and is responsive. The Material UI, a React component library, will be utilized to load persistent graphical elements such as buttons and icons for Twitter, home, search, notification, messaging, bookmark, listing, etc. CSS designs will be employed to enhance responsiveness and guide the user through button usage, using cursor control and hovering effects. CSS attributes will also be specified using CSS variables to ensure consistency in design and maintainability. In addition, by combining the Bootstrap 5 UI framework, the UI design tool Figma will be used for interface design and formatting.

#### **2.1.3 Programming Language**

JavaScript will be used as our primary programming language and the MERN stack (MongoDB, Express.js, React.js, and Node.js) as our web development framework. The React framework will oversee client-side JavaScript for front-end development, allowing for reusability and easy maintenance of replicated features by reusing React components. For middleware, Express will be used to handle HTTP requests. For the backend, Node will be responsible for retrieving the data on the server-side.

## 2.2 Architecture Diagram



## **2.3 System Components**

### **2.3.1 Profile System**

Users can upload their personal details, including their last name, first name, nickname, avatar, and a concise description, which are visible to all users, including other guest users.

### **2.3.2 Post System**

Users can post tweets with images or videos that include privacy controls; the post can be either public or private, with the latter only visible to their friends. Additionally, users can like, dislike, comment, and retweet all public and friend tweets.

### **2.3.3 Search Engine**

Users can use different keywords to search for different users and tweet tags; the search engine will retrieve the results from the database and display them in a list format.

### **2.3.4 Follower System**

Users can send follower requests by searching for the user in the search engine; they can also accept or decline other users' follower requests in the account interface. Additionally, they can choose to post private tweets, which can be viewed only by their followers.

### **2.3.5 Chat System**

Users can chat with their followers individually, with the conversation being visible only to the user and the follower. Access to the chatroom is granted solely by the user and his or her followers, making it inaccessible to others.

### **2.3.6 User management System (Admin)**

The admin account has a different user interface, where they can view a list of all existing user accounts and grant access to their information. They can also add new user accounts or remove existing user accounts from the list.