Document
Version 1.0

# UML Specification &
# UI Design Document

Group ID: A1

Group Members:

LAM Man Ho (1155159171)

MA Yu Ying (1155157879)

CHEUNG Ka Ho (1155158622)

FUNG Ngai Man (1155158312)

LI, Eric John (1155159116)

Department of Computer Science and Engineering (The Chinese University of Hong Kong)

Printing Date: 25/03/2023

## Content Page


**1. UML Design** ..... 5

**1.1. Account Management System** ..... 5

**1.1.1. Structural Diagram** ..... 5

**1.1.2. UMLs** ..... 5

**1.1.3. Functionalities** ..... 6

**1.2. User Operation** ..... 7

**1.2.1. Structural Diagram** ..... 7

**1.2.3. Functionalities** ..... 8

**1.2.4. Procedures and Functions** ..... 8

**1.3. Buzz Features** ..... 9

**1.3.1. Structural Diagram** ..... 9

**1.3.2. UMLs** ..... 9

**1.3.3. Functionalities** ..... 10

**1.3.4. Procedures and Functions** ..... 10

**1.4. Searching System** ..... 11

**1.4.1. Structural Diagram** ..... 11

**1.4.2. UMLs** ..... 11

**1.4.3. Functionalities** ..... 11

**1.4.4. Procedures and Functions** ..... 11

**1.5. Recommendation System** ..... 12

**1.5.1. Structural Diagram** ..... 12

**1.5.2. UMLs** ..... 12

**1.5.3. Functionalities** ..... 12

**1.5.4. Procedures and Functions** ..... 12

**1.6. Private Chat System** ..... 13

**1.6.1. Structural Diagram** ..... 13

**1.6.2. UMLs** ..... 13

**1.6.3. Functionalities** ..... 14

**1.6.4. Procedures and Functions** ..... 14

**1.7. Hive System** ..... 14

**1.7.1. Structural Diagram** ..... 14

**1.7.2. UMLs** ..... 14

**1.7.3. Functionalities** ..... 15

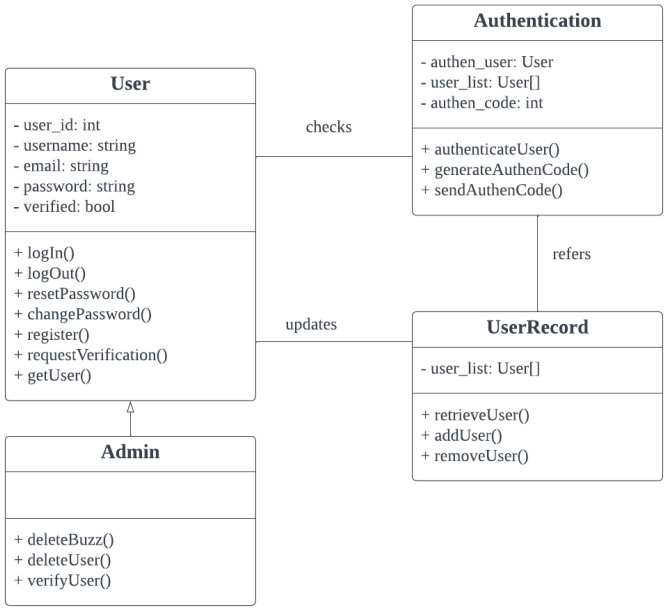
**1.7.4. Procedures and Functions** ..... 15
**1.8. Admin Operation System** ..... 15
**1.8.1. Structural Diagram** ..... 15
**1.8.2. UMLs** ..... 15
**1.8.3. Functionalities** ..... 16
**1.8.4. Procedures and Functions** ..... 16
**2. UI Design** ..... 16
**2.1. Sign in & Registration** ..... 16
**2.1.1. Description of the view** ..... 16
**2.1.2. Screen Images** ..... 16
**2.1.3. Objects and Actions** ..... 17
**2.2. Reset Password** ..... 17
**2.2.1. Description of the view** ..... 17
**2.2.2. Screen Images** ..... 17
**2.2.3. Objects and Actions** ..... 17
**2.3. Home Page** ..... 18
**2.3.1. Description of the view** ..... 18
**2.3.2. Screen Images** ..... 18
**2.3.3. Objects and Actions** ..... 18
**2.4. Posting System** ..... 18
**2.4.1. Description of the view** ..... 18
**2.4.2. Screen Images** ..... 19
**2.4.3. Objects and Actions** ..... 19
**2.5. Buzz** ..... 19
**2.5.1. Description of the view** ..... 19
**2.5.2. Screen Images** ..... 19
**2.5.3. Objects and Actions** ..... 20
**2.6. Searching System** ..... 20
**2.6.1. Description of the view** ..... 20
**2.6.2. Screen Images** ..... 21
**2.6.3. Objects and Actions** ..... 21
**2.7. User Profile** ..... 21
**2.7.1. Description of the view** ..... 21
**2.7.2. Screen Images** ..... 21
**2.7.3. Objects and Actions** ..... 22
**2.8. Chat Room** ..... 22

**2.8.1. Description of the view** ..... 22
**2.8.2. Screen Images** ..... 23
**2.8.3. Objects and Actions** ..... 23
**2.9. Hive Page** ..... 23
**2.9.1. Description of the view** ..... 23
**2.9.2. Screen Images** ..... 23
**2.9.3. Objects and Actions** ..... 23
**2.10. Administrator Management Page** ..... 24
**2.10.1. Description of the view** ..... 24
**2.10.2. Screen Images** ..... 24
**2.10.3. Objects and Actions** ..... 24

# 1. UML Design

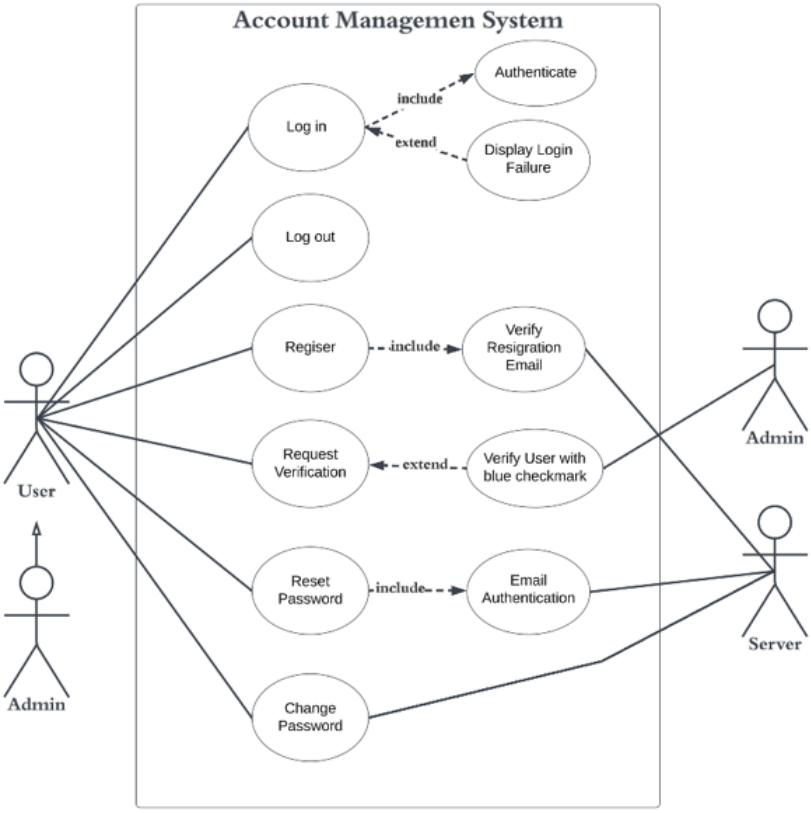
## 1.1. Account Management System

### 1.1.1. Structural Diagram

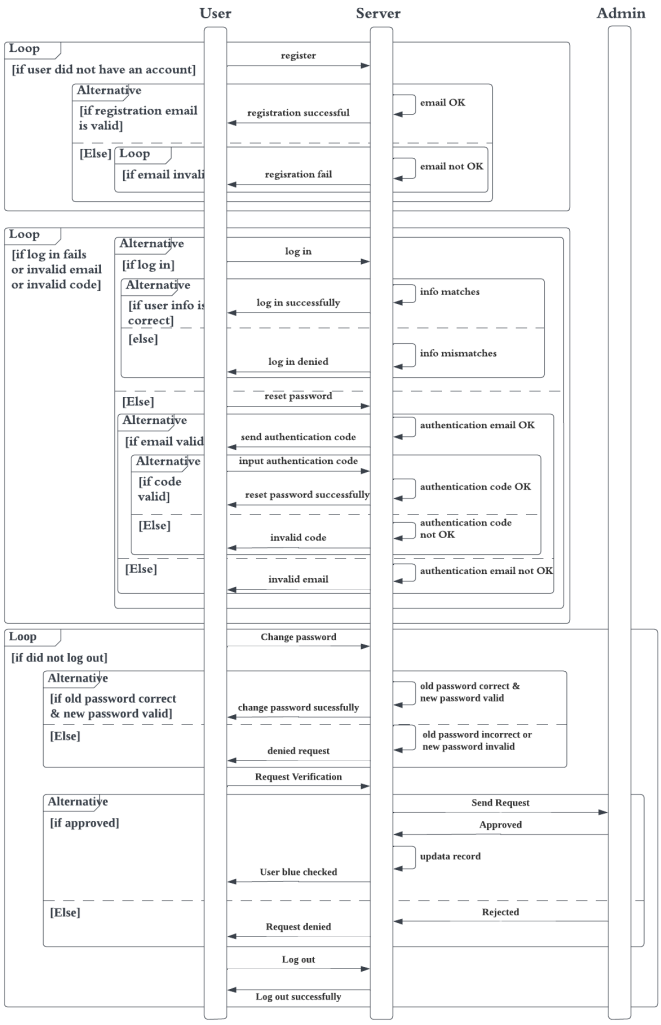



Class Diagram

### 1.1.2. UMLs




Use Case Diagram




Sequence Diagram

### 1.1.3. Functionalities

The Account Management System allows the users to manage their account. This component allows users to register a Buzzer account, log in, reset password, change password, request for verification, and log out.

### 1.1.4. Procedures and Functions

register(): This function allows users to registrate an account by entering a email address and. a password. This function will check whether the email entered is valid (valid when the email is not registered). If the email is valid, the registration successful message would be displayed; else the registration failure message would be displayed.

logIn(): This function allows users to log in by entering their registered email and password. This function will check if the entered user info matches that in the database. If it matches, user would be logged in and can perform Buzzer user operations; else, users log in request would be denied and let user re-enter email and password for login.

resetPassword(): This function allows users to reset their password in case if they forgot their. password. Users need to input their registered email, then an authentication code would be generated to their email by generateAuthenCode() and sendAuthenCode() to let them verify their identity. If they can authenticate their identity, they reset their password by entering a new password.

changePassword(): This function allows user to change their password after login. Users need to input their old password once and new password twice to be able to change their password. If the old password is wrong or the new password is invalid, the change password request would be denied; else the password would be changed successfully.

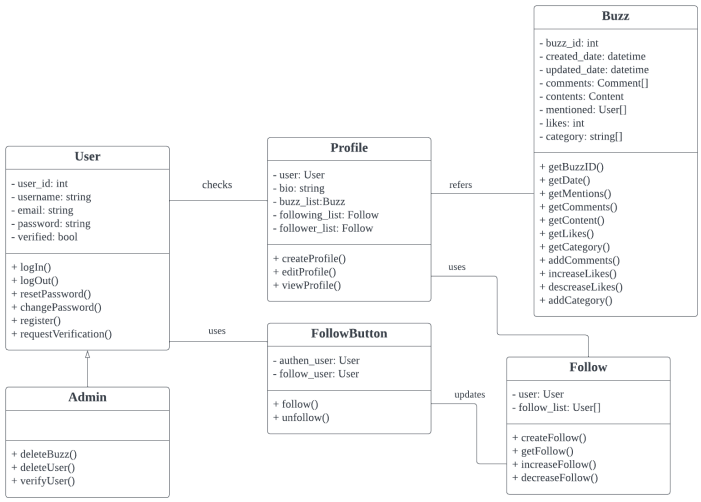
requestVerification(): This procedure allows user to send a request to the admin to verify them, where a check mark (tick) would be added next to the username of users after verification. Users are allowed to repeatedly request for verification, but only after request denied. If the users have their request approved, they are not allowed to request for verification anymore.

logOut(): This procedure allows users to log out by clicking the log out button. The log in token would also be deployed.

Note: All changes made by functions would also be updated in the database

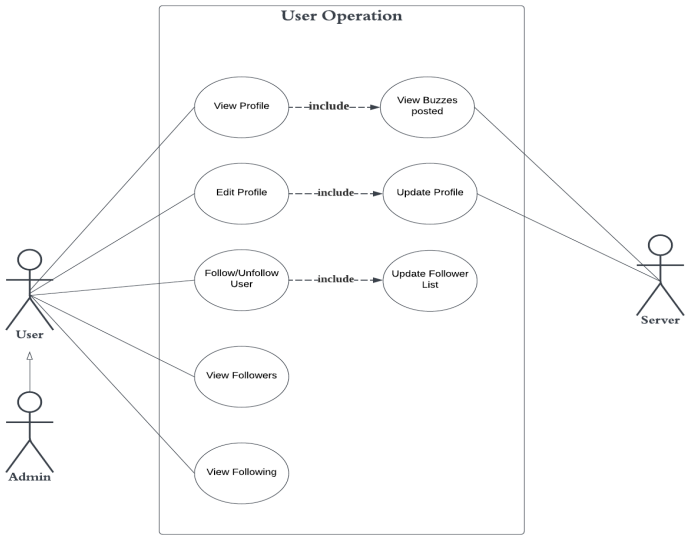
## 1.2. User Operation

### 1.2.1. Structural Diagram

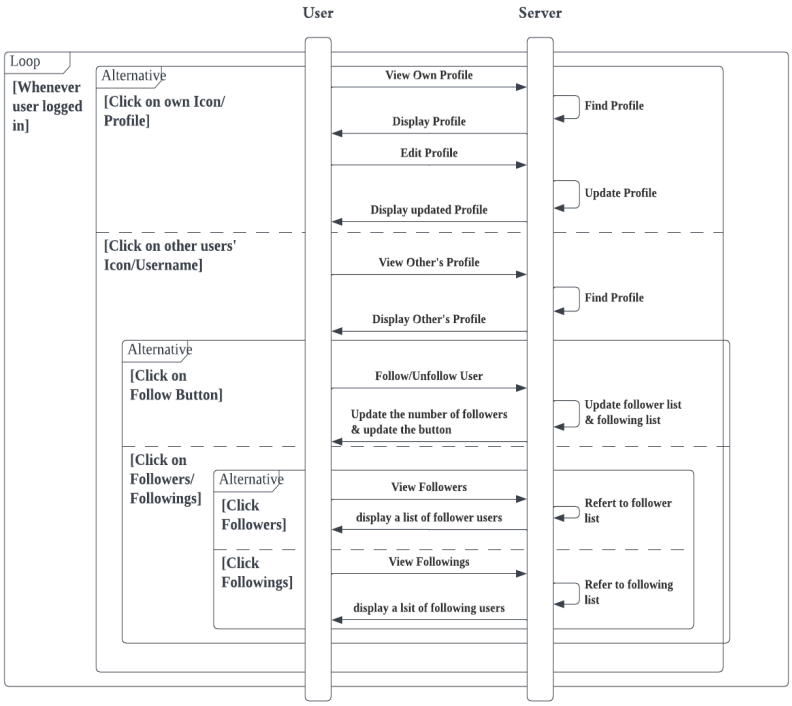



Class Diagram

### 1.2.2. UMLs




Use Case Diagram




Sequence Diagram

### 1.2.3. Functionalities

The User Operation allows users to perform basic user operation. Basic user operations include edit one's own profile, view other users' profile, follow/unfollow a user. Users can see the buzzes posted by a user from the user profile page.

### 1.2.4. Procedures and Functions

viewProfile(): Users are allowed to view other users' profile by clicking on the other users' icon or their username. Users are also allowed to view own profile by clicking on their own profile icon. User biography, followers, following, posted buzzes can all be seen in the profile page.

editProfile(): Users are allowed to edit their own profile such as their personal information and self-biography.

follow(): Users can follow a user through the profile page. The number of followers of the user would be real time increased.

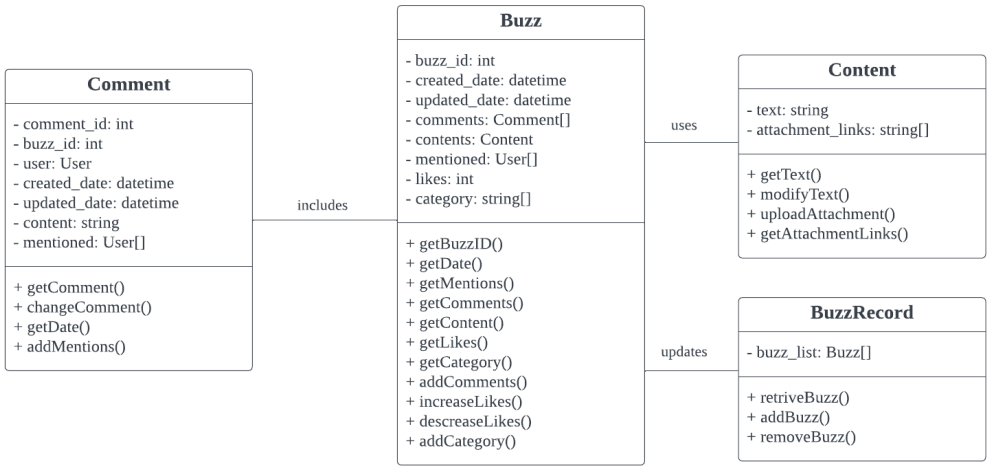
unfollow(): Users can unfollow a user through the profile page. The number of followers of the user would be real time updated.

getFollow(): This function helps to display the follower list, or the following list to users according to user request.

Note: All changes made by functions would also be updated in the database

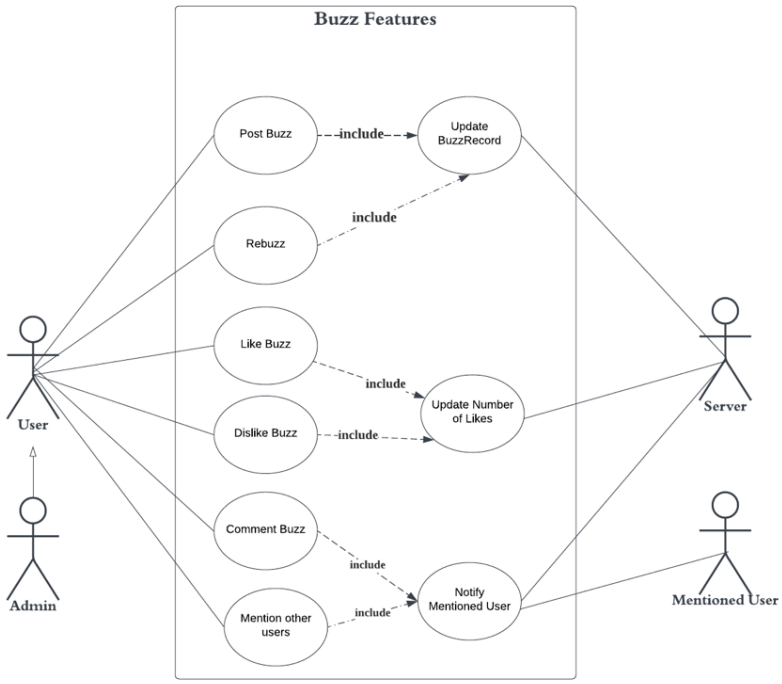
## 1.3. Buzz Features

### 1.3.1. Structural Diagram

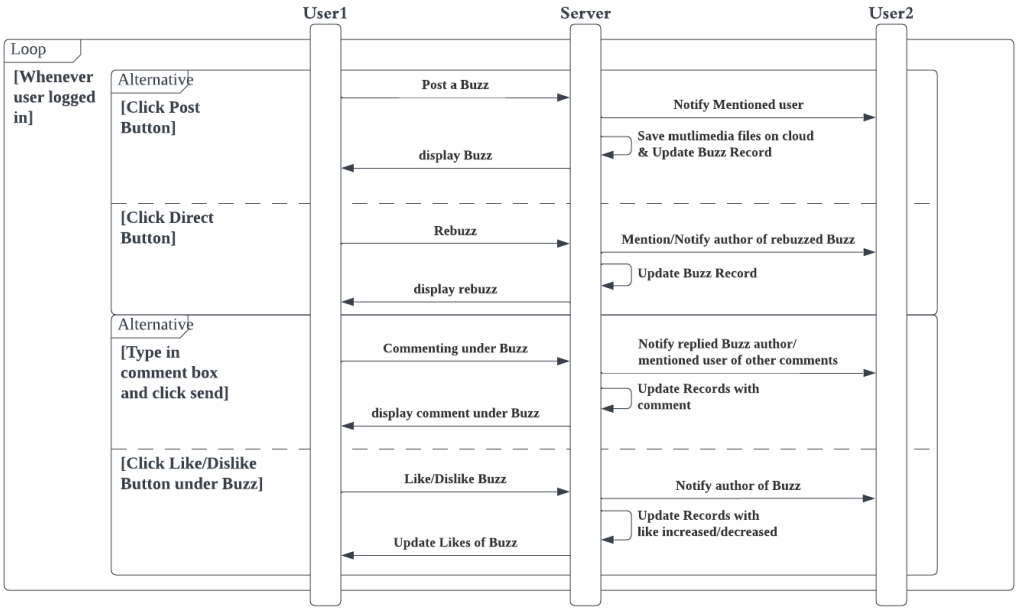



Class Diagram

### 1.3.2. UMLs




User Case Diagram




Sequence Diagram

### 1.3.3. Functionalities

The Buzz Features allows uses to perform Buzz operations. Buzz operations include posting buzzes, commenting under buzzes, like/dislike the buzzes, like/dislike the comments, and rebuzz.

### 1.3.4. Procedures and Functions

post(): This function allows users to enter their desired content in the text box and click the post button to post a Buzz in Buzzer.

rebuzz(): This function allows users to repost other users' buzzes as their own Buzz by clicking the "direct" button.

increaseLike(): This function increases the number of likes, display the increased likes, and color the "thumbs up" button of the buzz after the user clicked the "thumbs up" button.

decreaseLike(): This function decreases the number of likes, display the decreased likes, and color the "thumbs down" button of the buzz after the user clicked the "thumbs down" button.

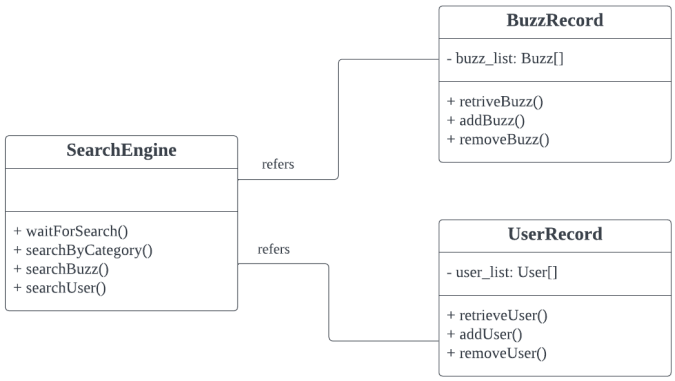
addComment(): This function adds the comment entered by users under a buzz and shows it in real time.

addMention(): This function hyperlinks the mentioned users' username (to their profile page) and notify the user that they are mentioned.

Note: All changes made by functions would also be updated in the database

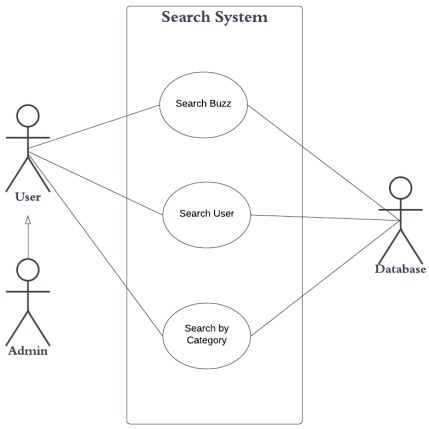
## 1.4. Searching System

### 1.4.1. Structural Diagram

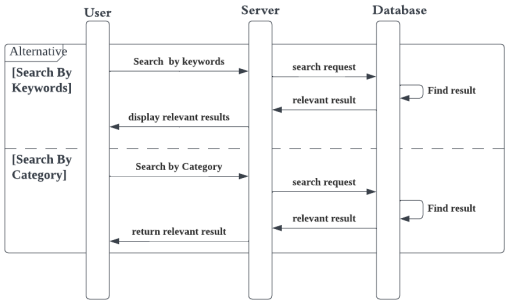



Class Diagram

### 1.4.2. UMLs




Use Case Diagram




Sequence Diagram

### 1.4.3. Functionalities

The Search System allows the user to search for buzzes, users, or buzzes and users with some specified category. Users can search for buzzes by entering keywords, search users by username or user_id, or search by clicking on the categories provided.

### 1.4.4. Procedures and Functions

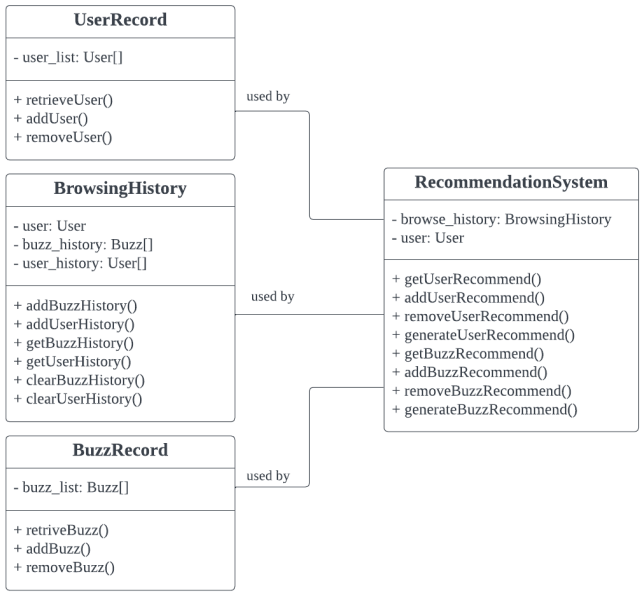
seacrhUser(): This function shows a list of users that contains the keyword entered by the user by searching from the database

seacrhBuzz(): This function shows a list of relevant buzzes that contains the keyword entered by searching from the database.

searchByCategory(): This function shows the list of related users and buzzes corresponding to the selected category.

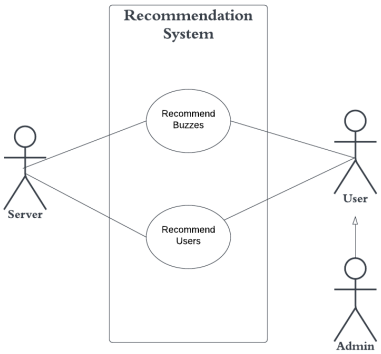
## 1.5. Recommendation System

### 1.5.1. Structural Diagram

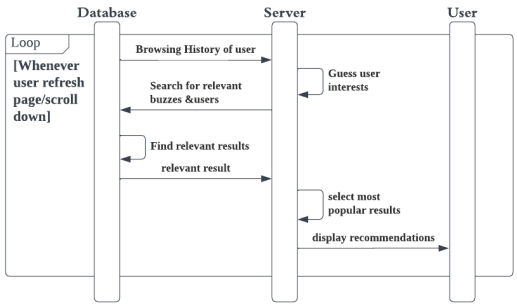



Class Diagram

### 1.5.2. UMLs




Use Case Diagram




Sequence Diagram

### 1.5.3. Functionalities

The Recommendation System would recommend users buzzes and users that they might be interested in.

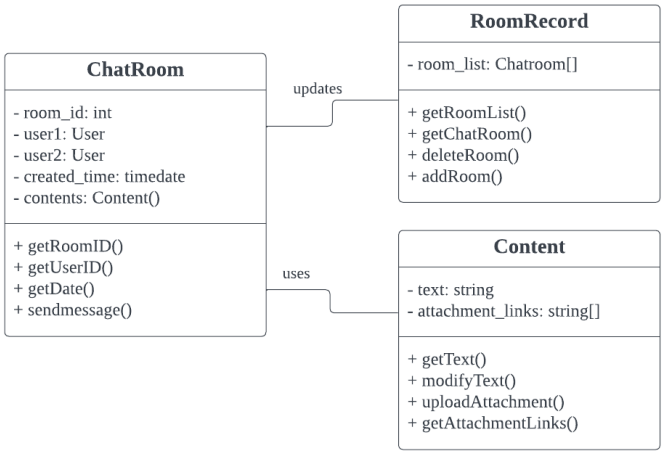
### 1.5.4. Procedures and Functions

generateUserRecommend(): This function recommends users a list of users that they may be interested according to their browsing history retrieved form the database.

generateBuzzRecommend(): This function recommends users a list of users that they may be interested according to their browsing history retrieved form the database.

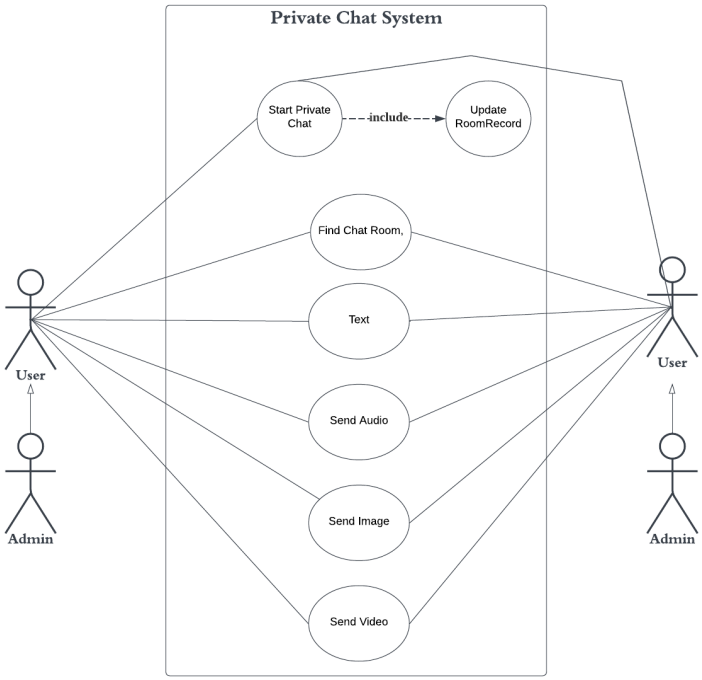
## 1.6. Private Chat System

### 1.6.1. Structural Diagram

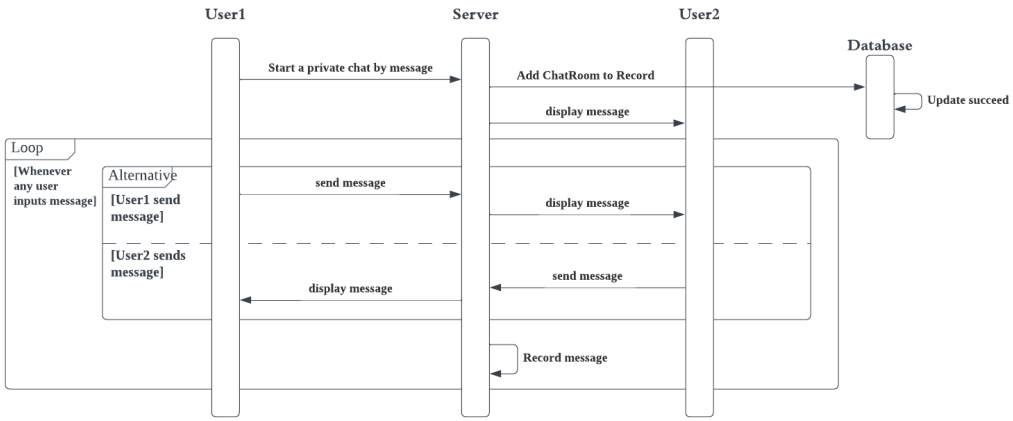



Class Diagram

### 1.6.2. UMLs




Use Case Diagram




Sequence Diagram

### 1.6.3. Functionalities

The Private Chat System allows users to start a private chat with other users. Users will build a private chat room between each other, and can send messages that includes text, audio, video, and images.

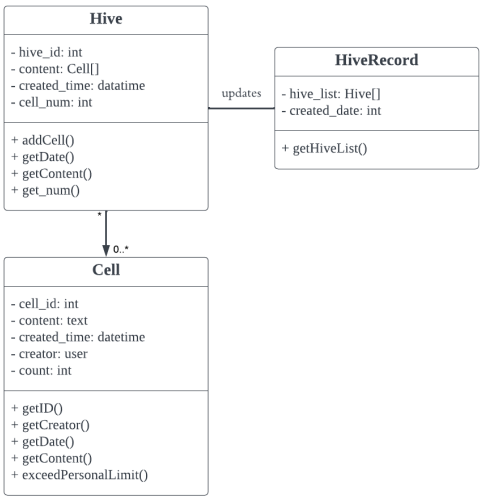
### 1.6.4. Procedures and Functions

addRoom(): This function helps building the private chat room and record this chat room in the database for reference.

sendMessage(): This function allows a user to send message to another user in the chat room. Multimedia documents are stored in cloud server and the links (to download them) are stored in database.

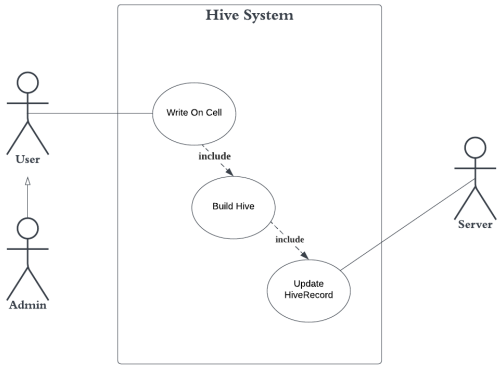
## 1.7. Hive System

### 1.7.1. Structural Diagram

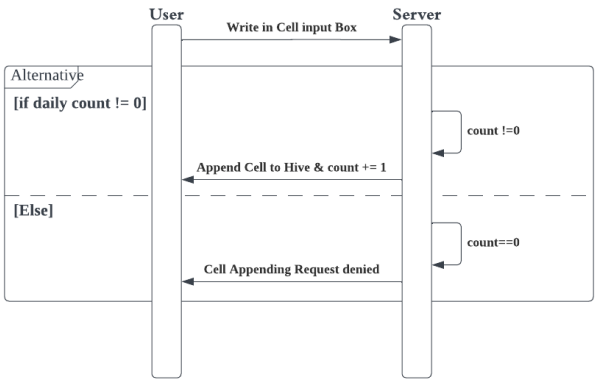



Class Diagram

### 1.7.2. UMLs




Use Case Diagram




Sequence Diagram

### 1.7.3. Functionalities

The Hive System allows the users to work together to build a hive like bees. Each user is only allowed to contribute to building the hive by adding one cell every day.

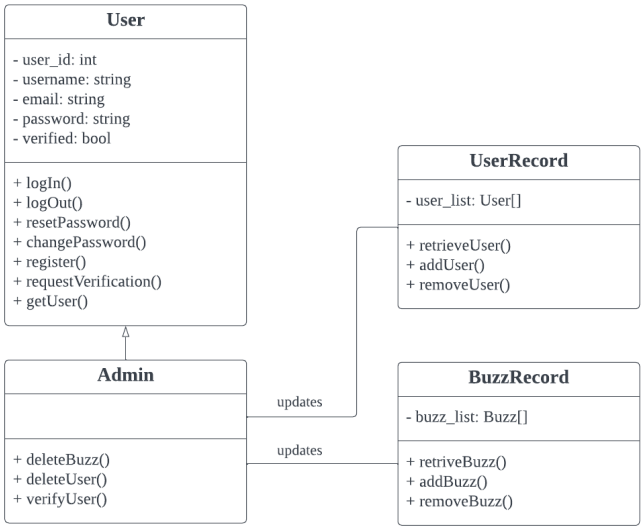
### 1.7.4. Procedures and Functions

addCell(): This function allows users to write their to-share text in the cell and merge it to the Hive. When the cell is added, it is also added to the database and increase the counter of the user to ensure only single cell can be posted per day.

buildHive(): This function initialize a daily Hive, and replace the old Hive. Every hive. initialized by this function would be also recorded by the database for history tracing purpose.

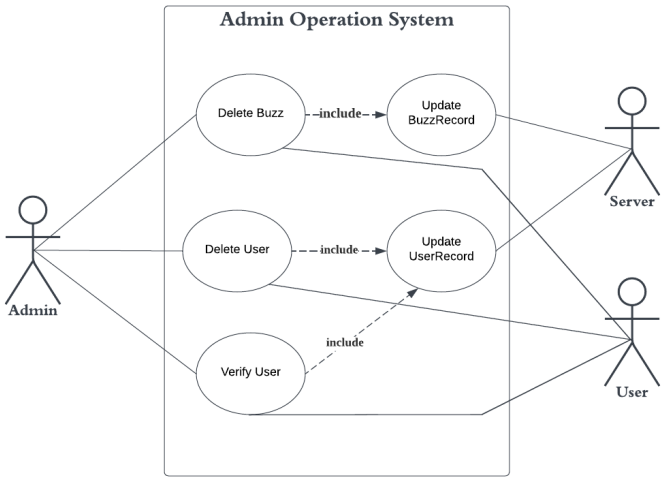
## 1.8. Admin Operation System

### 1.8.1. Structural Diagram

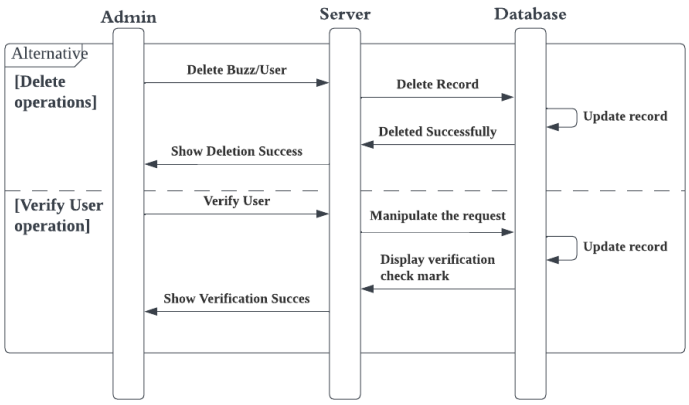



Class Diagram

### 1.8.2. UMLs




Use Case Diagram




Sequence Diagram

### 1.8.3. Functionalities

The Admin Operation System allows the admins to manage the web platform. Admin can delete inappropriate buzzes/user and verify users with a check mark.

### 1.8.4. Procedures and Functions

deleteUser(): This function allows admin to delete the corresponding user from database.

verifyUser(): This function allows admin to verify a user. User record would also be updated in the database indicating the user is verified.

deleteBuzz(): This function allows admin delete the corresponding buzz from database.

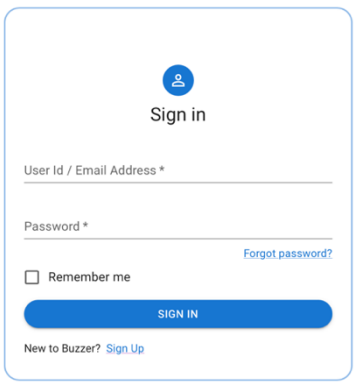
# 2. UI Design

## 2.1. Sign in & Registration

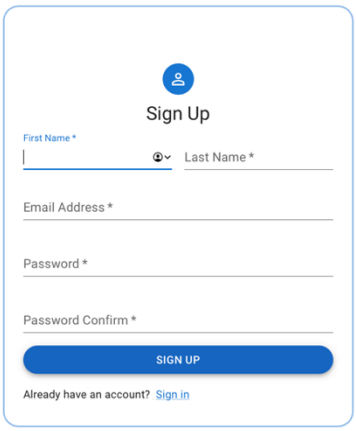
### 2.1.1. Description of the view

This view shows the sign in and registration page. Users should input their user id or email address with their corresponding password to log in. If users do not have an account, they can click the “Sign Up” button to register an account with submitting the required information.

### 2.1.2. Screen Images




Sign In UI




Sign Up UI

### 2.1.3. Objects and Actions

Sign In:

- Users can log in by inputting their user id or email address with password in the corresponding textbox, and then clicking the “SIGN IN” button (users would be directed to a Buzzer main page).
- If users did not have a Buzzer account yet, users can click the “Sign Up” button to register for a Buzzer account. Users can go back to sign in page by clicking the “Sign In” button.
- If users forgot their password, users could click the “Forgot password?” button. Then, users would be directed to the reset password page.

Sign Up:

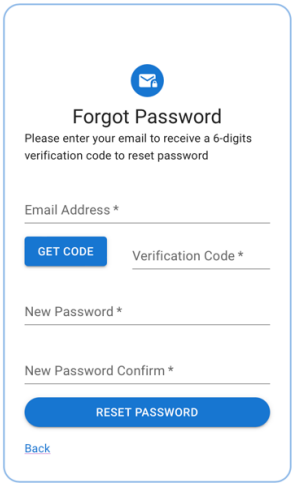
- Users can register an account by inputting their username, email, user id and repeated password into the corresponding place shown and then clicking the “SIGN UP” button (users would be directed to a Buzzer main page).

## 2.2. Reset Password

### 2.2.1. Description of the view

This view shows the reset password page where the users input their registration email to verify their identity for resetting their password.

### 2.2.2. Screen Images




### 2.2.3. Objects and Actions

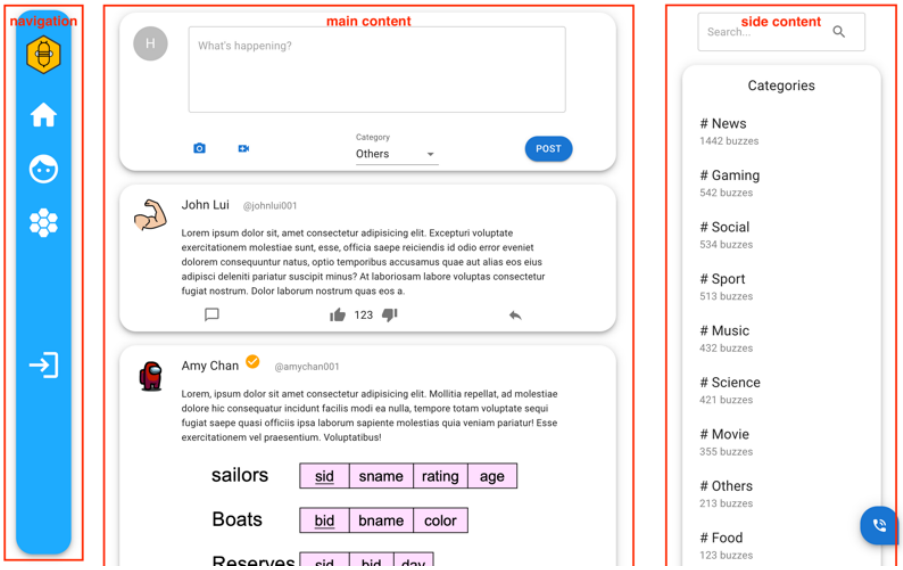
- Users should input their registration email of their Buzzer account and click the “GET CODE” button. Then, users would receive a 6-digit verification code in their email inbox, which is used for identity verification. After entering the verification code, users can input the new password and click the “RESET PASSWORD” button in order to reset password.
- Users can go back to previous page by clicking the “Back” button.

## 2.3. Home Page

### 2.3.1. Description of the view

This view shows the home page of the website. The home page divides into three major parts: navigation, main content and side content. The main content contains the posting system and buzzes. The side content contains the searching system, popular buzzes and category search.

### 2.3.2. Screen Images




### 2.3.3. Objects and Actions

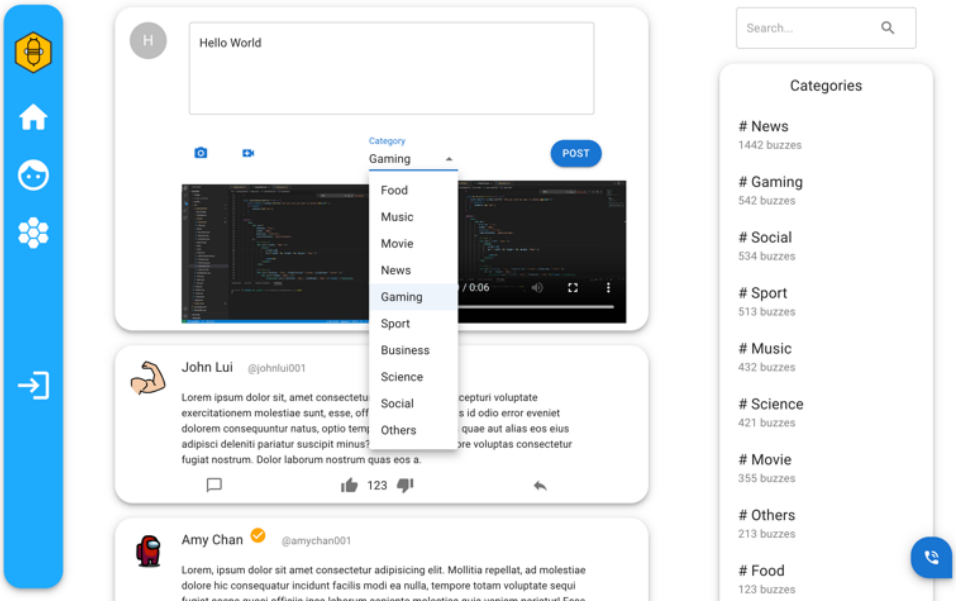
- Users can link to different other pages through navigation.
- Users can post their own buzzes in the posting system (details below).
- Users can click on the “Search” bar to activate the search system in the site content (details below). Then, users can click any place out of the search system in order to exit.
- Users can simply scroll down the main page to view different buzzes, including buzzes posted by users they followed/recommended buzzes.
- Users can open the chat box by clicking the “call” button.

## 2.4. Posting System

### 2.4.1. Description of the view

This view shows the posting system in the main content in home page. Users can post their buzz with content, image, video and category.

### 2.4.2. Screen Images




### 2.4.3. Objects and Actions

- Users can input the buzz content in the input box.
- Users can import an image and a video by clicking the “Camera” button and “Film” button respectively. Then, users can select an image and video from the local folder and the image and video will be previewed below.
- Users can select a category for their buzz by selection.
- Users can click the “POST” button in order to post a buzz.

## 2.5. Buzz

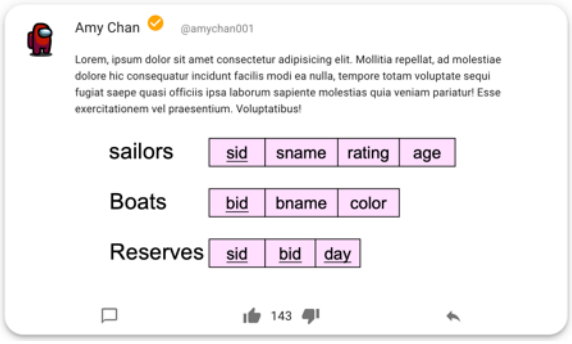
### 2.5.1. Description of the view

This view shows the Buzz Preview and an independent Buzz Page with a comment system.

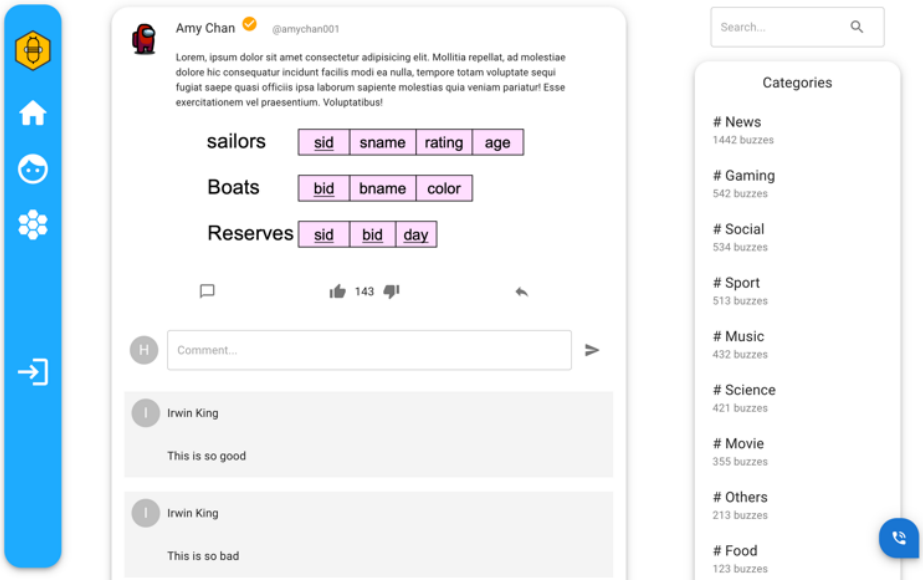
The Buzz Preview contains the author’s information and the buzz content.

The Buzz Page includes the buzz itself, the comment system and other users’ comments.

### 2.5.2. Screen Images




Buzz Preview




Buzz Page

### 2.5.3. Objects and Actions

Buzz Preview:

- Users can like/ dislike and re-buzz the buzz post by clicking the corresponding buttons.
- Users can click the author name or icon to navigate to the required user profile page.
- Users can click the comment button, then navigate to the corresponding buzz page.

Buzz Page:

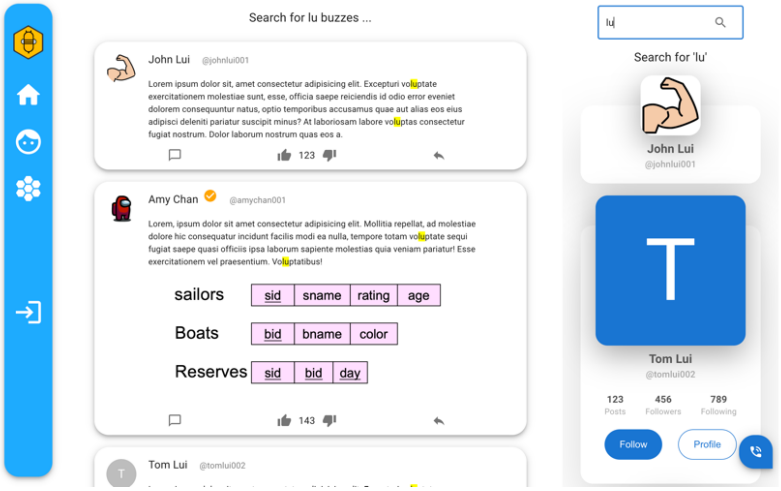
- Buzz Page includes the functions in Buzz Preview.
- Users can leave a comment in the input box and then click the “send” button.
- Users can browse other users’ comments below the comment system.

## 2.6. Searching System

### 2.6.1. Description of the view

This view shows the search system of the website. The search system includes the users searching and buzzes searching. The user's search results will be displayed in the side content dynamically based on the user’s inputs. The buzzes searching results will be displayed in the main content with highlighted the searching keyword in the content.

### 2.6.2. Screen Images




### 2.6.3. Objects and Actions

Searching for users:

- Users can search for a user by simply inputting the user's name in the search box.
- Users can check for the users' profile preview by moving the mouse to the corresponding user's item.

Searching for buzzes:

- Users can search for buzzes by inputting the keywords in the search box and then clicking the search icon or pressing enter keyboard button.

## 2.7. User Profile

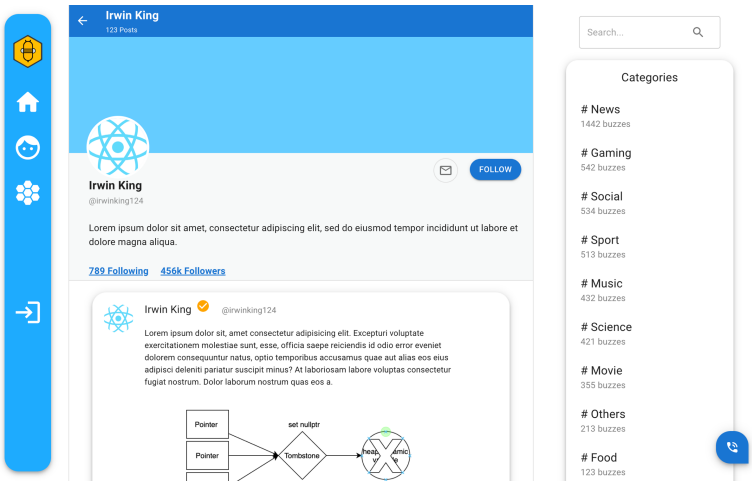
### 2.7.1. Description of the view

This view shows the user profile page and following/follower list in the main content.

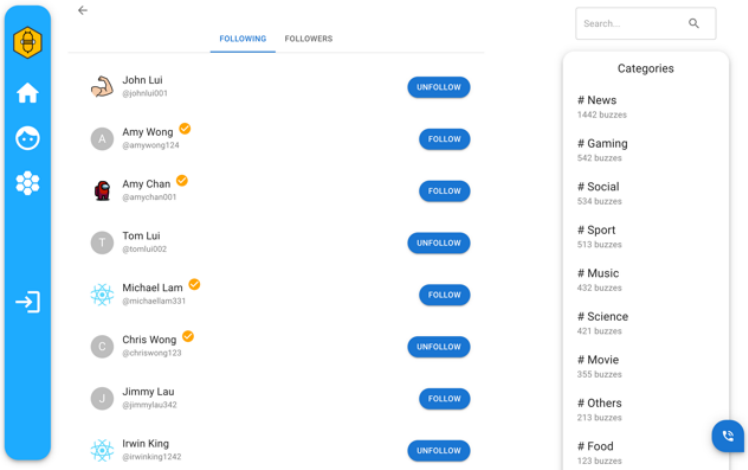
The user profile contains user information, including the username, user id, description and the buzzes posted by that user. If the user viewing the profile is the same as the user being viewed, the button is "EDIT". Otherwise, the button is "FOLLOW" or "UNFOLLOW".

The following/follower list shows the corresponding list of users with the "follow" state.

### 2.7.2. Screen Images



User Profile Page




User Following / Followers Page

### 2.7.3. Objects and Actions

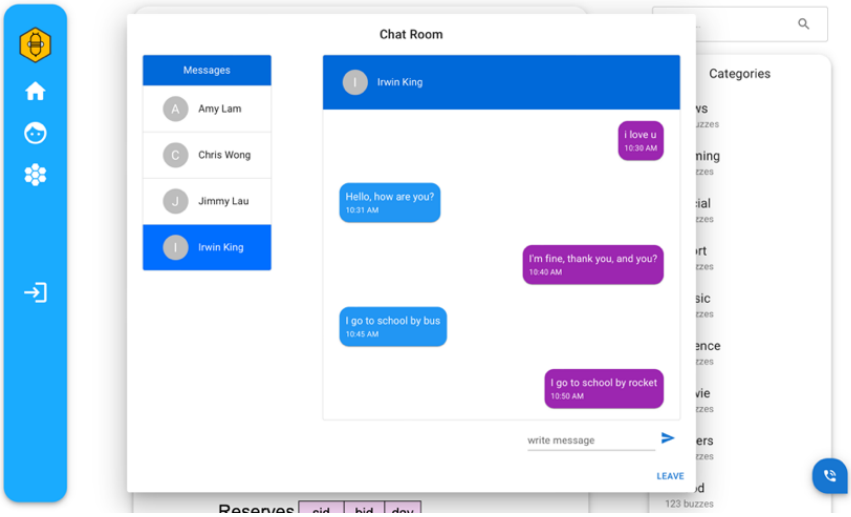
- If the user viewing the profile is the same as the user being viewed, they can edit their user information by clicking the “EDIT” button.
- If the viewing user is different from the user being viewed, the “EDIT” button is replaced with a “FOLLOW” / “UNFOLLOW” button, and the user can click the button to follow or unfollow this user.
- Users can click the “xxx Following” and “xxx Followers” buttons in the user profile to browse the following and follower list.
- In the following/follower list, users can click the “Following” and “Followers” buttons to switch the corresponding user list.
- Users can click the “Mail” button and start a private chat with the corresponding user.

## 2.8. Chat Room

### 2.8.1. Description of the view

This view shows the chat room system. Users can start a private chat with other users by following that user.

### 2.8.2. Screen Images




### 2.8.3. Objects and Actions

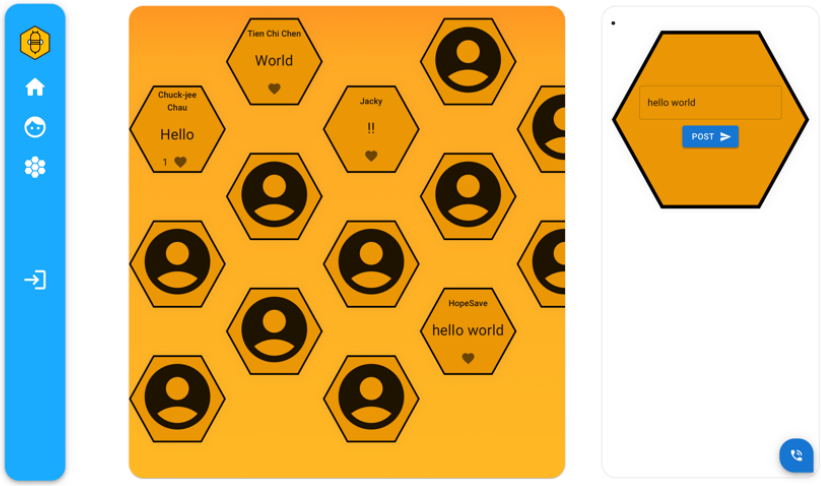
- Users can enter the chat room by clicking the call button, and leave the chat room by clicking the "LEAVE" button.
- Users can select the person they want to chat with in the Messages List.
- Users can send a message to another user by simply typing the content into the input box and clicking the send button.

## 2.9. Hive Page

### 2.9.1. Description of the view

This view shows the Hive that is built by all users. Users can contribute to building up the public hive by adding a hexagon per day (users are limited to adding one hexagon a day).

### 2.9.2. Screen Images




### 2.9.3. Objects and Actions

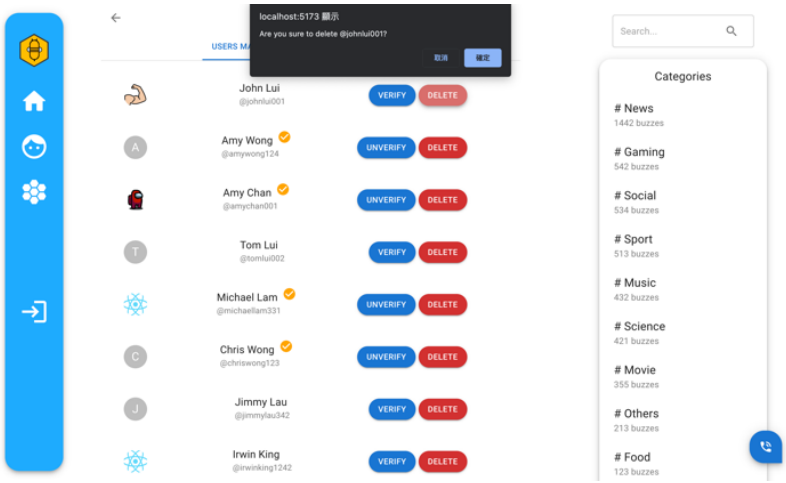
- Users can create a hive by inputting the content in the input box and clicking the “POST” button in the side content.
- Users can view and like other users’ hives by clicking the hidden hexagon.

## 2.10. Administrator Management Page

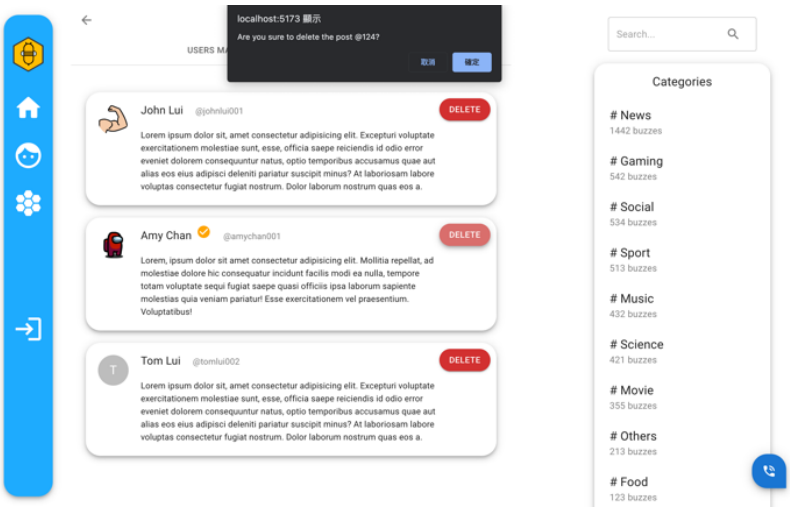
### 2.10.1. Description of the view

This view shows the management page of users with administrator right. Administrators have the responsibility to maintain Buzzer, so administrators have more rights over ordinary users. Administrators are allowed to delete inappropriate buzzes, verify a user, and delete a user.

### 2.10.2. Screen Images




User Management




Buzz Management

### 2.10.3. Objects and Actions

Admin can switch between Users and Buzzes Management by clicking the top buttons.

Users Management:

- Admin can delete a user by clicking the “DELETE” button and verify a user by clicking the “VERIFY” button. After clicking the button, the browser will alert a message for getting the admin confirmation.

Buzzes Management:

- Admin can delete a buzz by clicking the “DELETE” button. After clicking the button, the browser will alert a message for getting the admin confirmation.