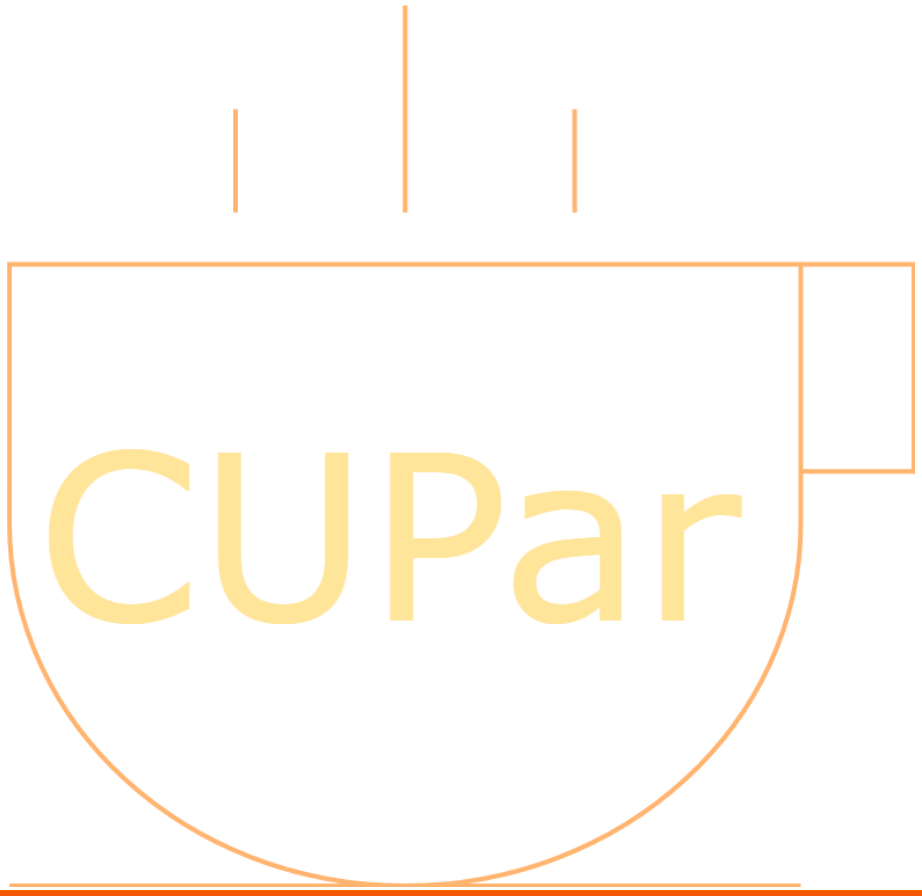


CUPar Final Report

Group01

LI Yuxin	**1155107874**
WEI Qi	**1155107666**
WEI Wang	**1155107718**
XIAO Tianyi	**1155107819**
ZHAO Feng	**1155107824**

CSCI3100

5/4/19

Contents

1 INTRODUCTION

1.1 Project Overview

CUPar is a platform helping CUHK students to find all kinds of partners. For CUHK students, there are mainly two kinds of partners they need to find: roommate and teammate. Besides, one may want to find friends to travel together, partners to join the workshop as a team, or someone to play sports after class. CUPar is designed to let students find roommate, teammate, and any kind of partners in an easier way.

1.2 Objective

CUPar aims to provide CUHK students with a quicker approach to find all kinds of partners. Today, when living in halls or having classes, due to limited social circle, students often encounter the problem that they can't find their ideal roommate or teammate. What's more, their roommate or teammate may be quite annoying, like freeriders. Using CUPar, students can get to know other students that have similar living habits or the same course project. They can also decide whether to accept or refuse after chatting. For teammate, if freeriders are encountered, they can report him or her after the project, and the system will put this person in the last of the rank list. Besides, CUPar also provides a forum to allow students to find other partners they may want to find. With the help of CUPar, students will have a better experience when dealing with their partners.

1.3 Highlights

1.3.1 User-friendly Interface

Our user interface is simple but beautiful and powerful. To maximize users' experience, we focus on making our website handy and convenient to use. Users do not need to pay more effort to too many inputs, but with just clicks, users can choose the specified inputs, which also benefits the system to modify the inputs.

1.3.2 Specially Designed for the CUHK Students

Our intended users are CUHK students. Users should use the CUHK email address (xxxxxxxxx@link.cuhk.edu.hk) to sign up for a new account. Specially designing for CUHK students can improve the users' experience by narrowing the scopes of users and making the matching results more precise. Also, using CUHK email address to sign up will prevent our system from malicious attack to some extent.

1.3.3 Full-featured Functions

Our system not only provides a platform to find project partners but also roommates and some partners for workshops, travelling, etc. Fully featured functions include but not limited to a complete account system includes, finding partners system, forum system and a simple real-time web-chat system, which will be all introduced in much more details in the report.

1.4 Project Statistics

1.4.1 Lines of Codes

Type of Files	Files	Lines of Blank	Lines of Comment	Lines of Codes
HTML (.html)	74	2019	56	10356
hbs (.hbs)	14	379	171	2965
ejs (.ejs)	6	15	6	288
JavaScript (.js)	51	1268	617	4869
CSS (.css)	16	64	146	6164
Total	161	3745	996	24720

1.4.2 McCabe number

(number of if-else and button)

Module / Component	Predicate Nodes	McCabe Complexity
Account System	108	109
Forum	29	30
Chat	14	15
Roommate	54	55
Teammate	176	177
Total	381	383

2 SYSTEM ARCHITECTURAL DESIGN by DFD

2.1 System Architecture

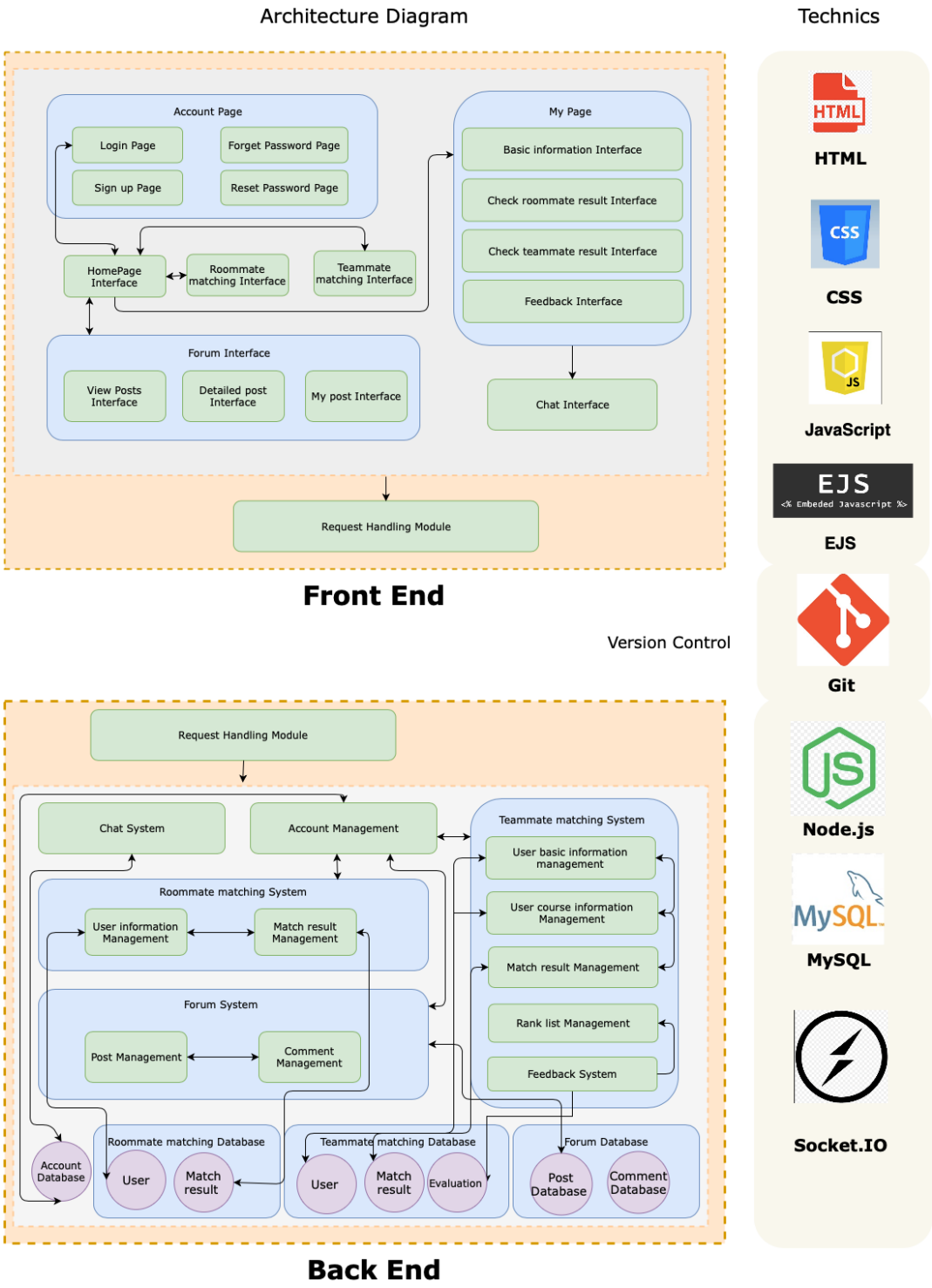


Figure 1 System Architecture

2.2 DFDs

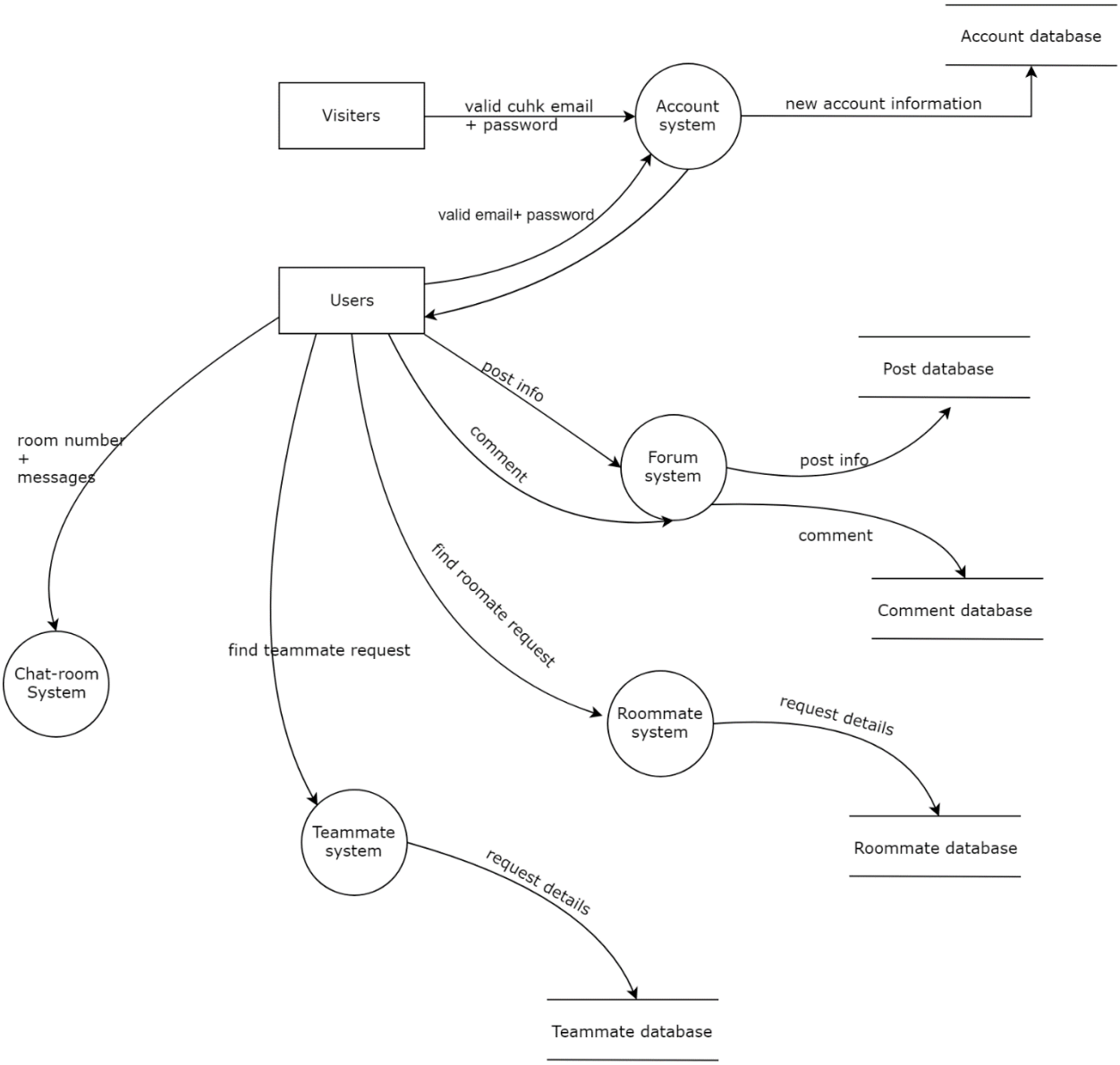


Figure 2 System DFD

2.2.1 Account Management

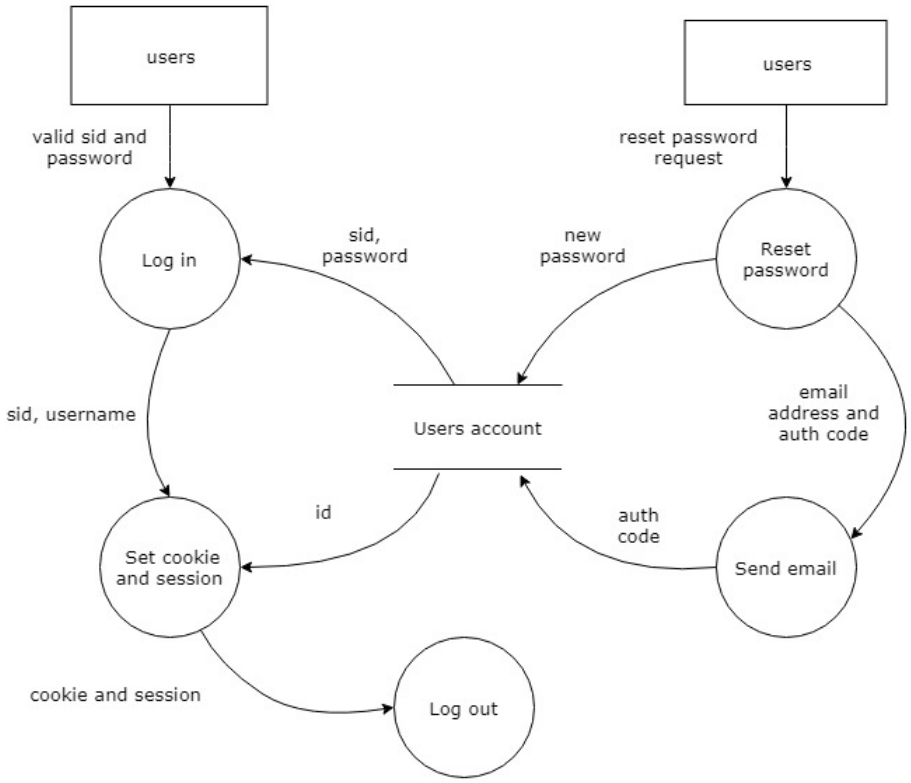


Figure 3 Login and Reset Password

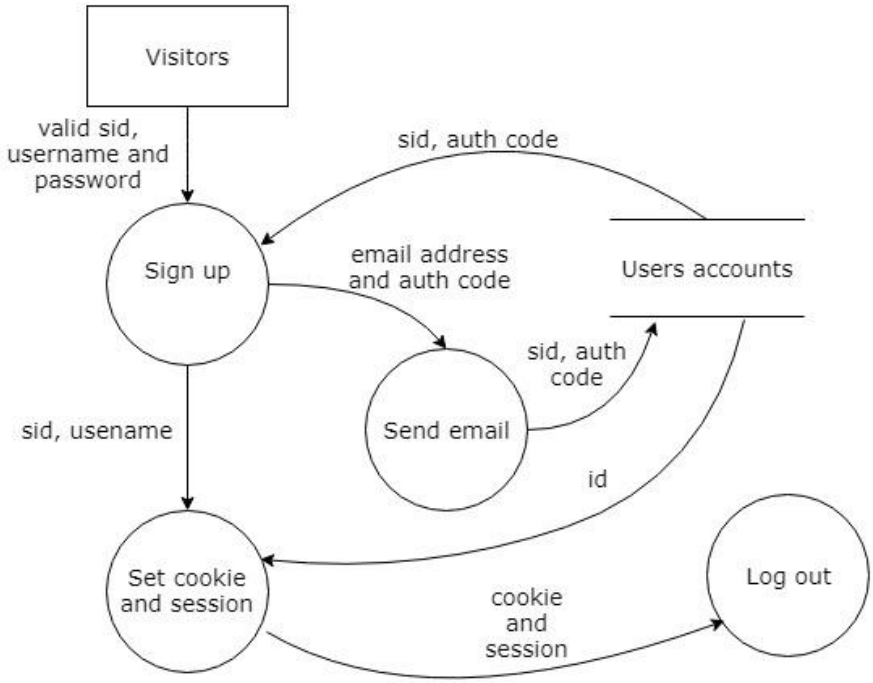


Figure 4 Sign Up

2.2.2 Chat-room Data Flow Diagram

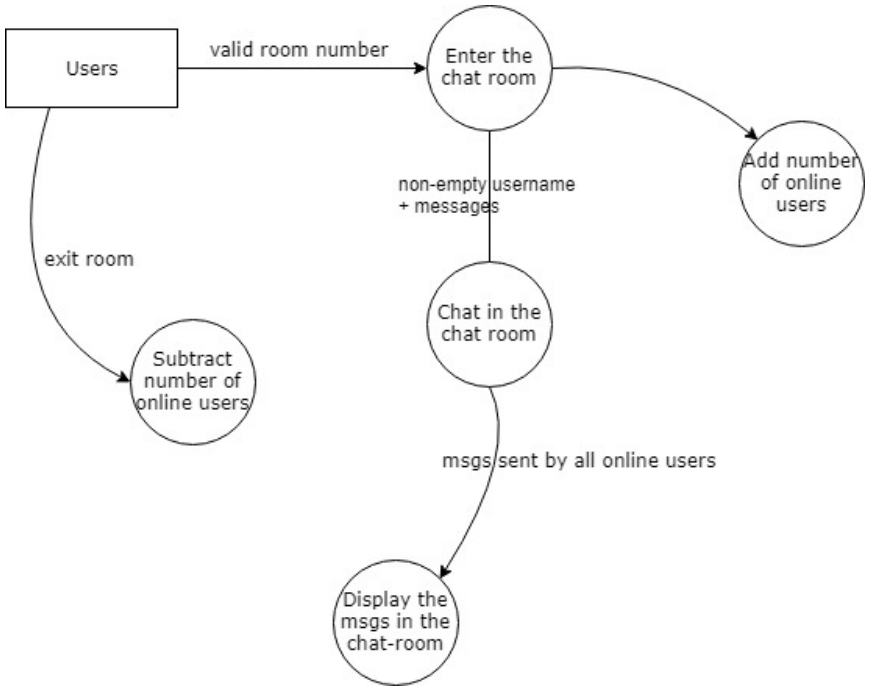


Figure 5 Chat System

2.2.3 Forum Data Flow Diagram

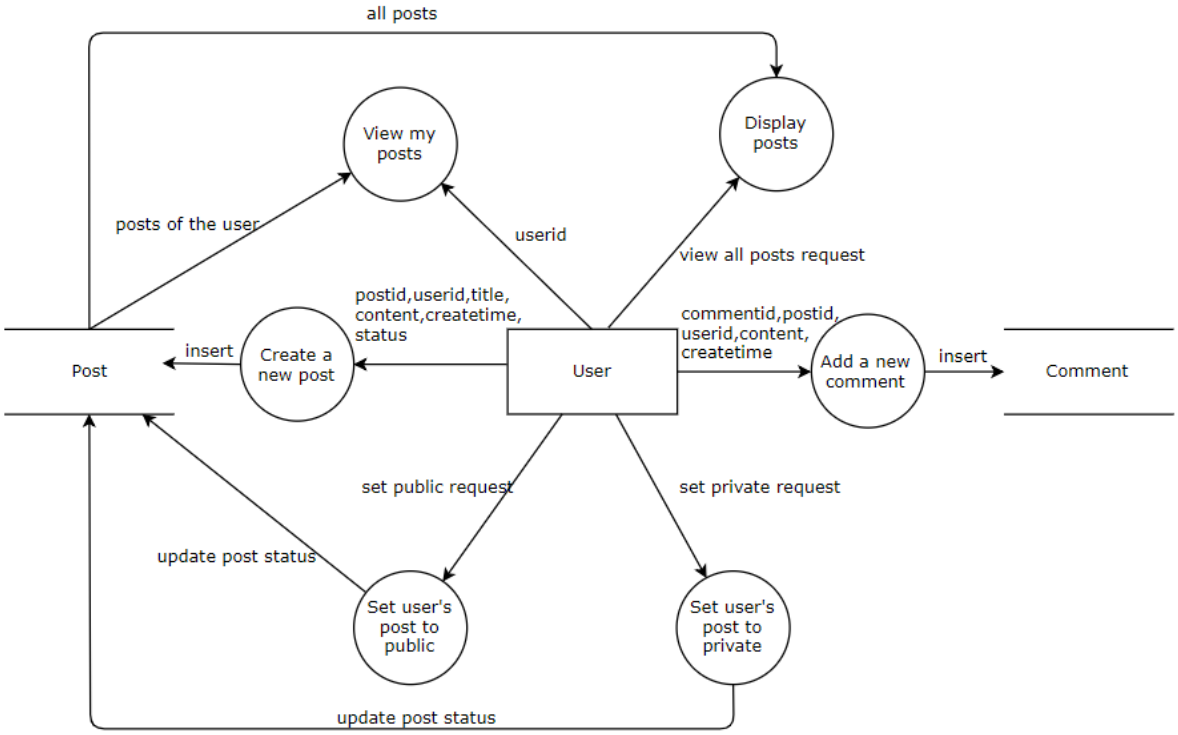


Figure 6 Forum System

2.2.4 Roommate Data Flow Diagram

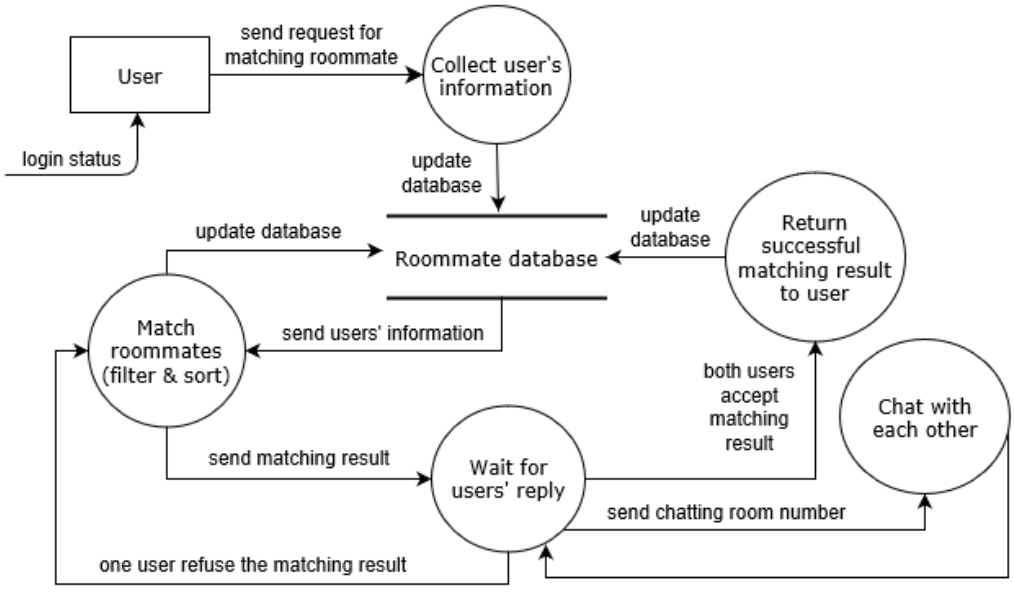


Figure 7 Find a Roommate

2.2.5 Teammate Data Flow Diagram

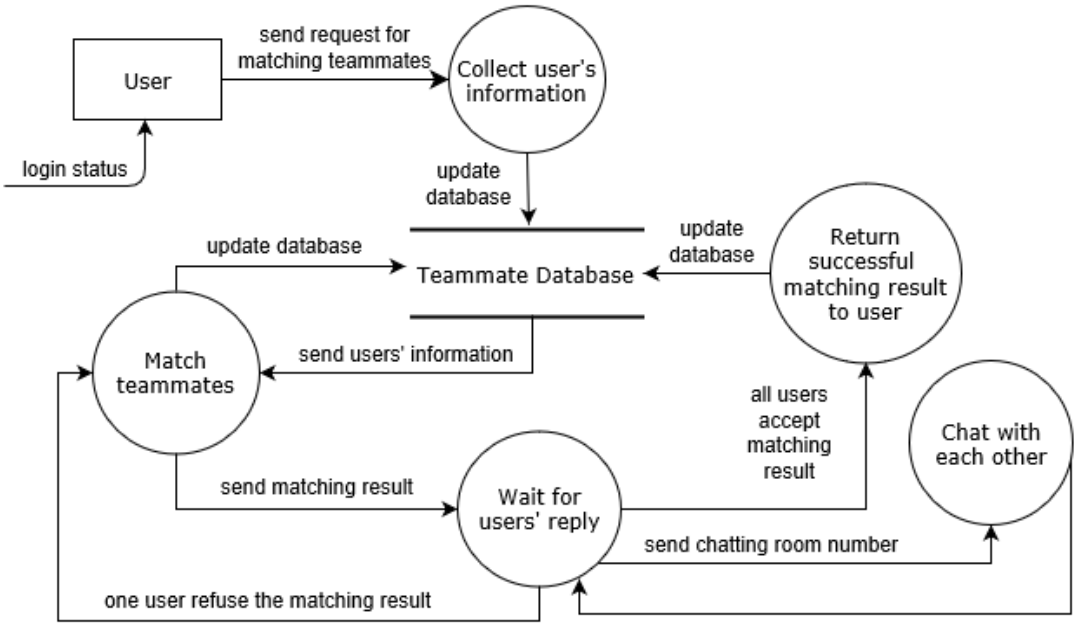


Figure 8 Find a Teammate

3 DETAILED DESCRIPTION OF COMPONENTS by UML

3.1 Account Component

3.1.1 Structural Diagram

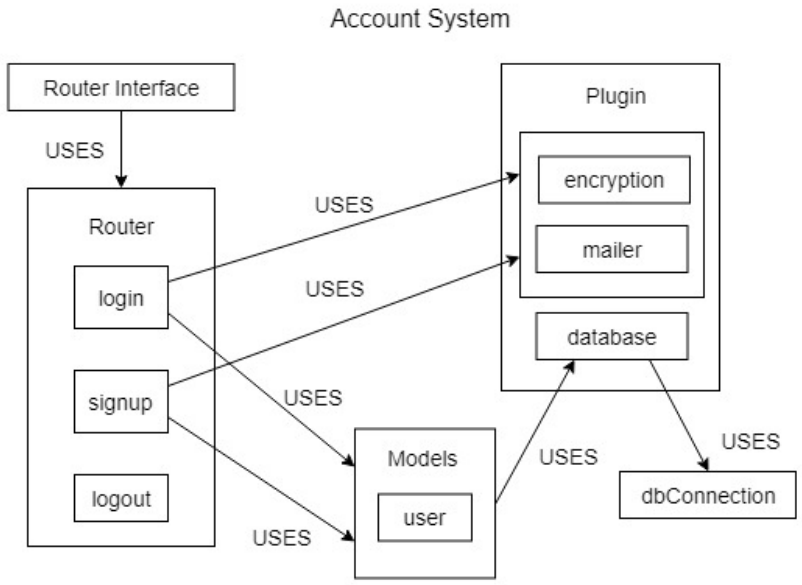


Figure 9 Structure of Account System

3.1.2 UMLs

3.1.2.1 UML Use-case Diagram

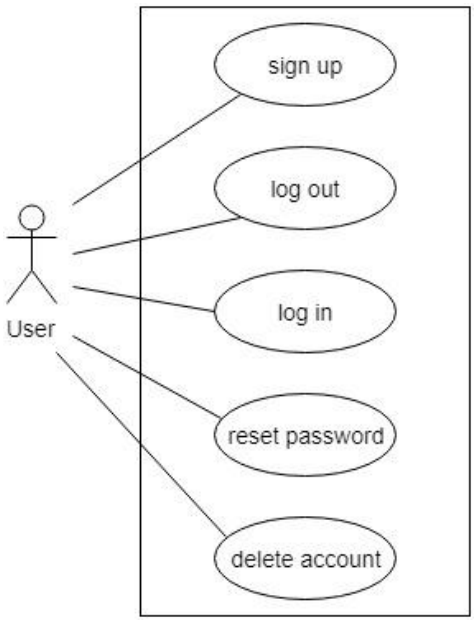


Figure 10 Use-case of Account System

3.1.2.2 UML Class Diagram

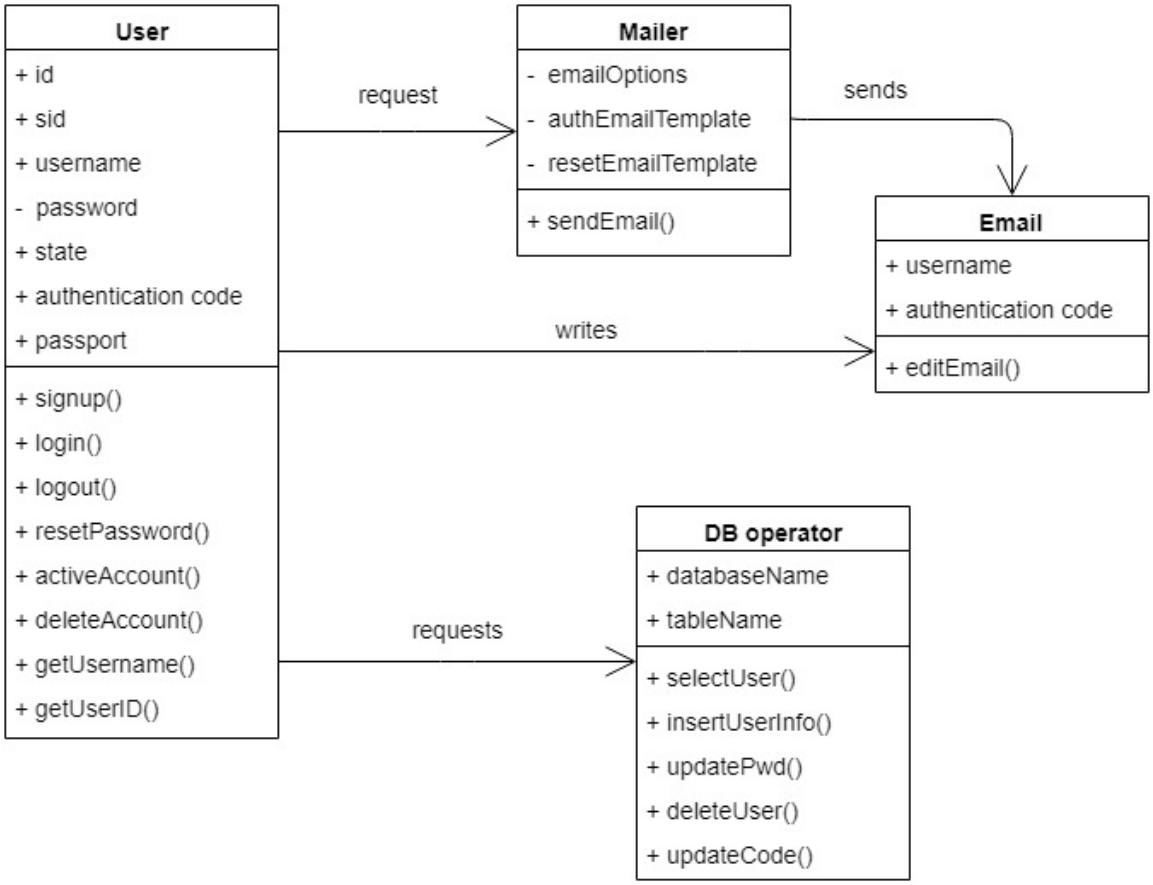


Figure 11 Class of Account System

3.1.2.3 UML Sequence Diagram

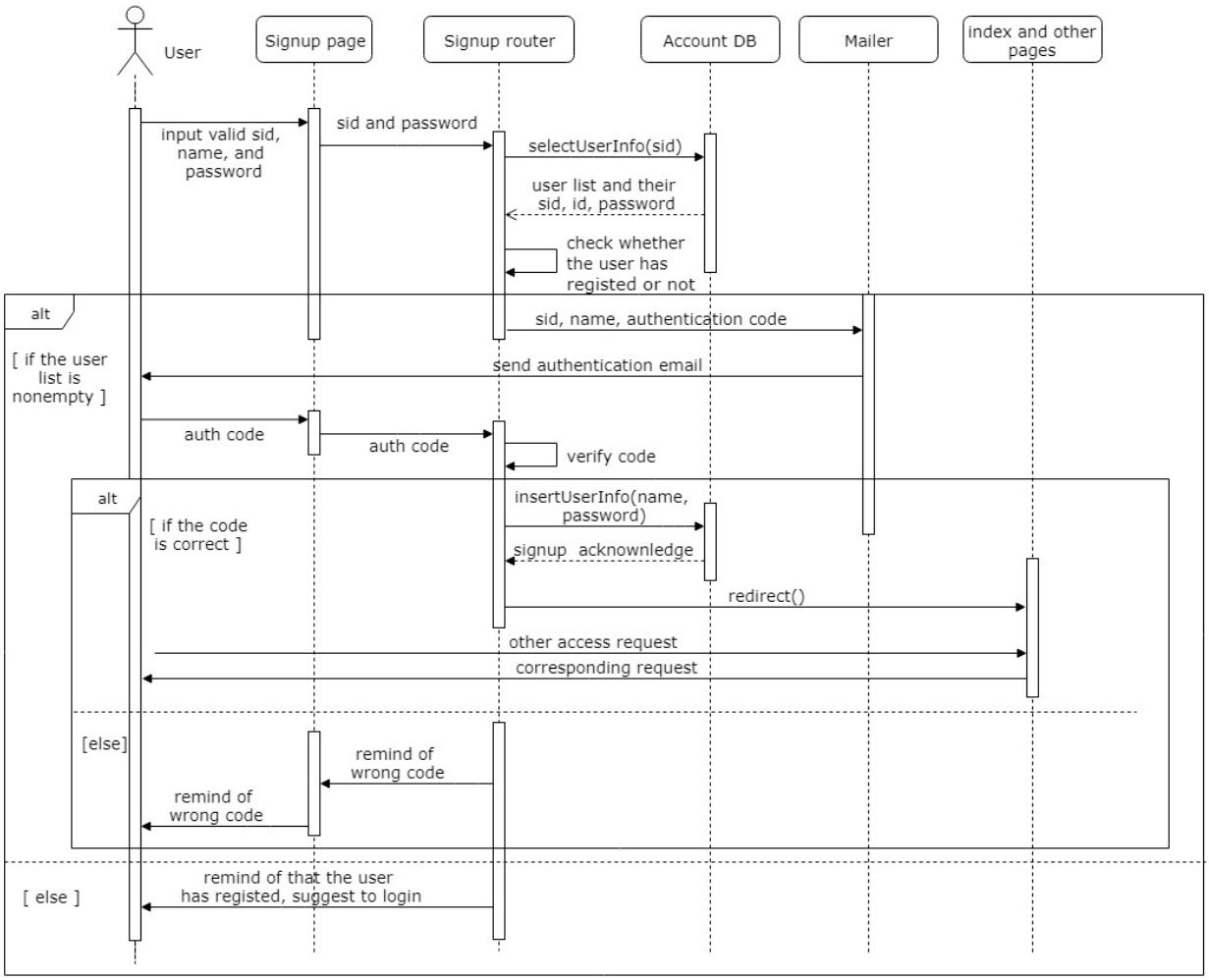


Figure 12 Sequence of Account System

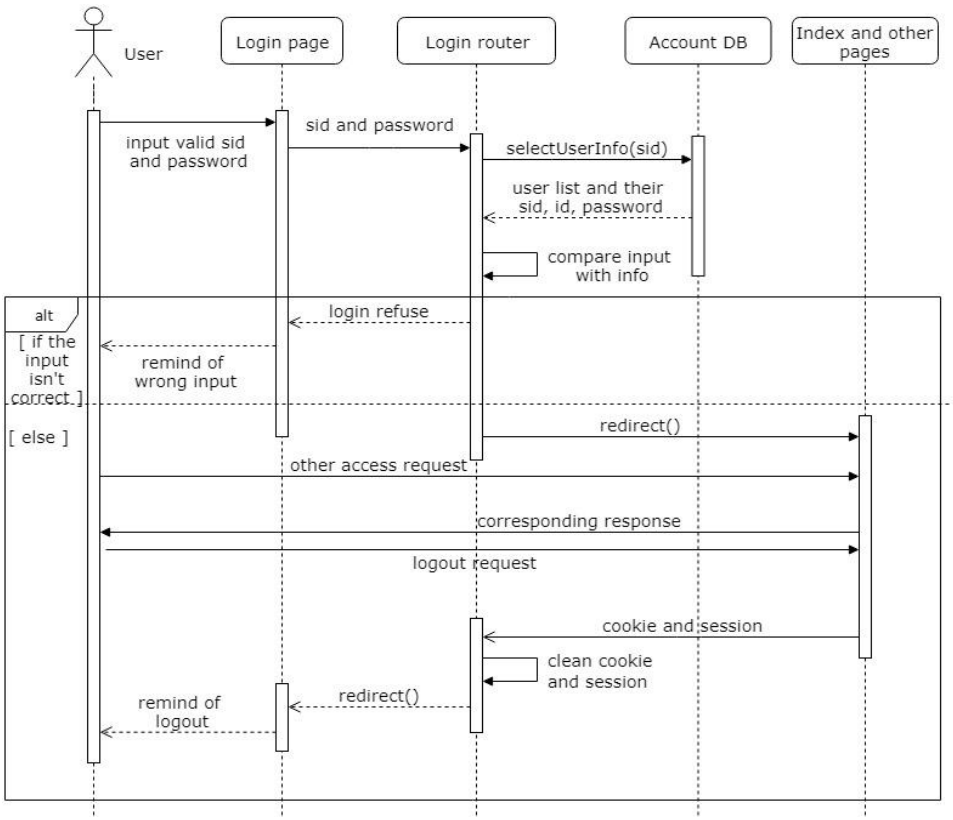


Figure 13 Sequence of Account System

3.1.2.4 UML Activity Diagram

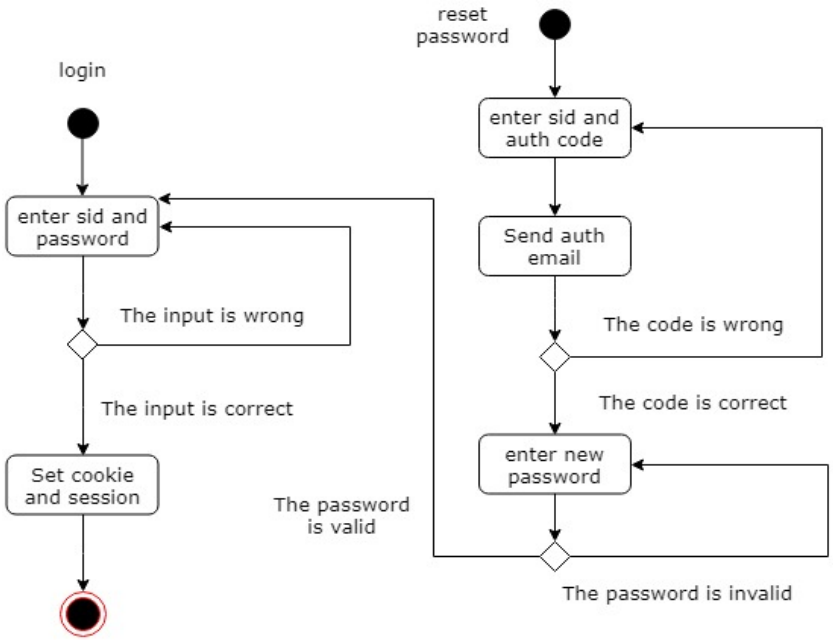


Figure 14 Activity Diagram of Account System

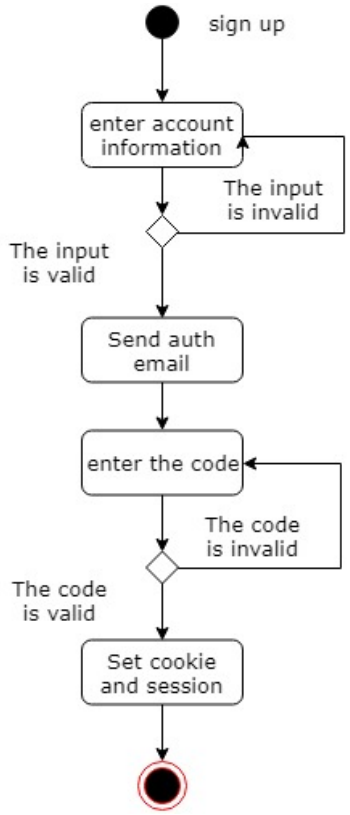


Figure 15 Activity Diagram of Account System

3.1.3 Functionality

This component handles all the operations regarding users accounts. The model of the user is defined in this module. The visitors can only access the index but any other pages or functions. Once users logged in the active account, all functions are available, and users don't need to re-login when they open a new tab or windows if the session is in idle no more than 1hour. Student IDs are required in all operations in the account system. The users are required to input the CODE sent to link-email to verify their identity in the procedure of signup, password reset, and account deletion.

3.1.4 Procedures and Functions

Function	Parameter	Description
signup	sid, username, password, passwordRP, CODE	Insert the information of the identified new user into the database. Assign a passport for the user and redirect to the index.
login	sid, password	Assign a passport for the user. Redirect to the index.
logout	passport	Clean all session and cookie of the user and redirect to the login page.
resetPassword	sid, CODE, new password	Update the password in the database with the new password. Redirect to the login page.
deleteAccount	sid, CODE, password	Delete the information of the identified user and redirect to the signup page.
activeAccount	sid, CODE, username, password, passwordRP	Update the state of the user to 1. If the user has not completed signup yet, they need to resign up again via this function with the new information and CODE to active the account.
sendEmail	email, address	Send the email address of the user an edited email template.
editEmail	username, CODE	Fill an email template with username and CODE.
selectUser	sid	Search the user's information with the sid.
insertUserInfo	sid, username, password, CODE	Insert the valid input of user into the database.

updatePwd	new_password	Search the user with the sid and update the password in the database with the new password.
deleteUser	sid, password, CODE	Search the user with sid and delete all information of the user in the database.
updateCode	new_CODE	A new generated CODE will be updated to the database.

The procedure of key functions:

Signup:

In the signup procedure, the user is required to input sid, username, CODE, and repeat the password to confirm the password. If the users have recorded in the database, they will be reminded of logging in their accounts. An email including CODE will be sent to their link-email. The sid and CODE will be inserted into the database before sending the email but the user can't log in because the account is inactive. If and only if the input is matched with the database, the other information will be updated to the database and account becomes active. After completing the signup, users will be assigned with a passport including their ids, sids, usernames and be redirected to the index.

Login:

In the login procedure, the user is required to input sid and password. If the users have not registered or active their accounts yet, they will be reminded of going to the signup page to sign up or resign up. If the input is matched with the database, a passport will be assigned to the user.

Reset Password:

In the procedure of resetting passwords, if the users have not registered yet, they can't reset. An email including CODE will be sent to their link-email. In addition, the state of the user is updated to 0 and the account becomes inactive. If and only if the identity of the user is verified by the CODE, a temporary session including sid is used to remember the identity of the user in the current windows, and the user can input the new password. After confirmed, the new password will be updated.

3.2 Forum Component

3.2.1 Structural Diagram

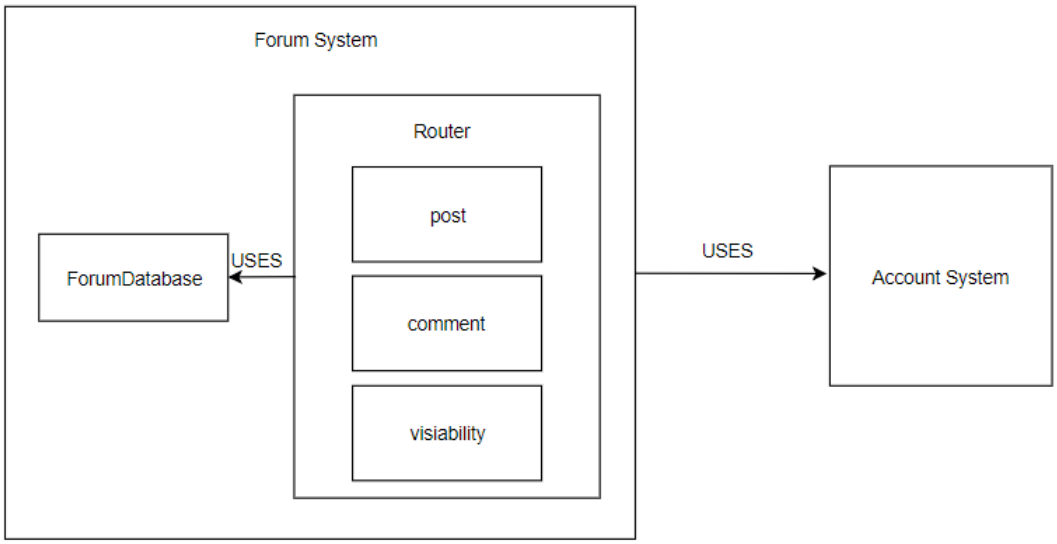


Figure 16 Structure of Forum System

3.2.2.1 UML Use-case Diagram

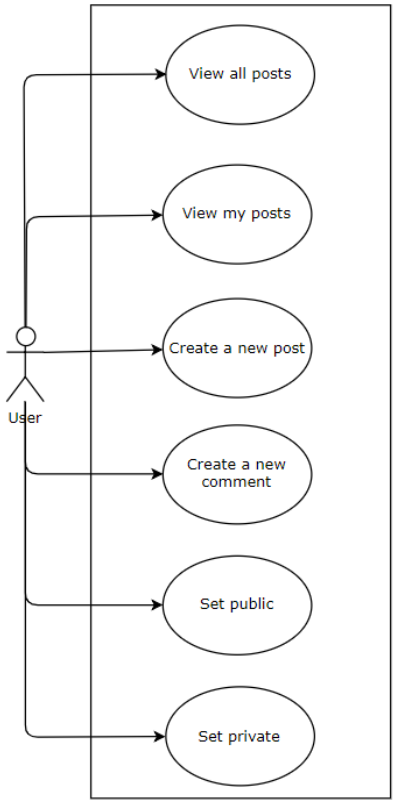


Figure 17 Use-case of Forum System

3.2.2.2 UML Class Diagram

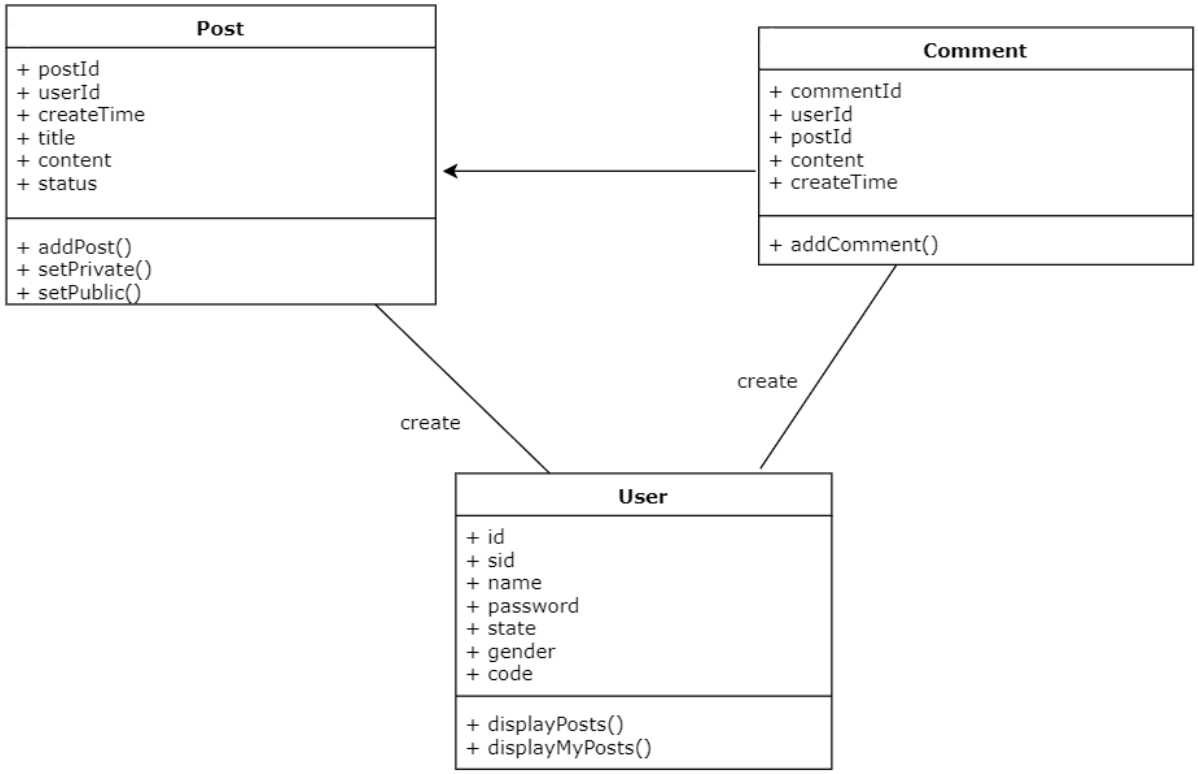


Figure 18 Class of Forum System

3.2.2.3 UML Sequence Diagram

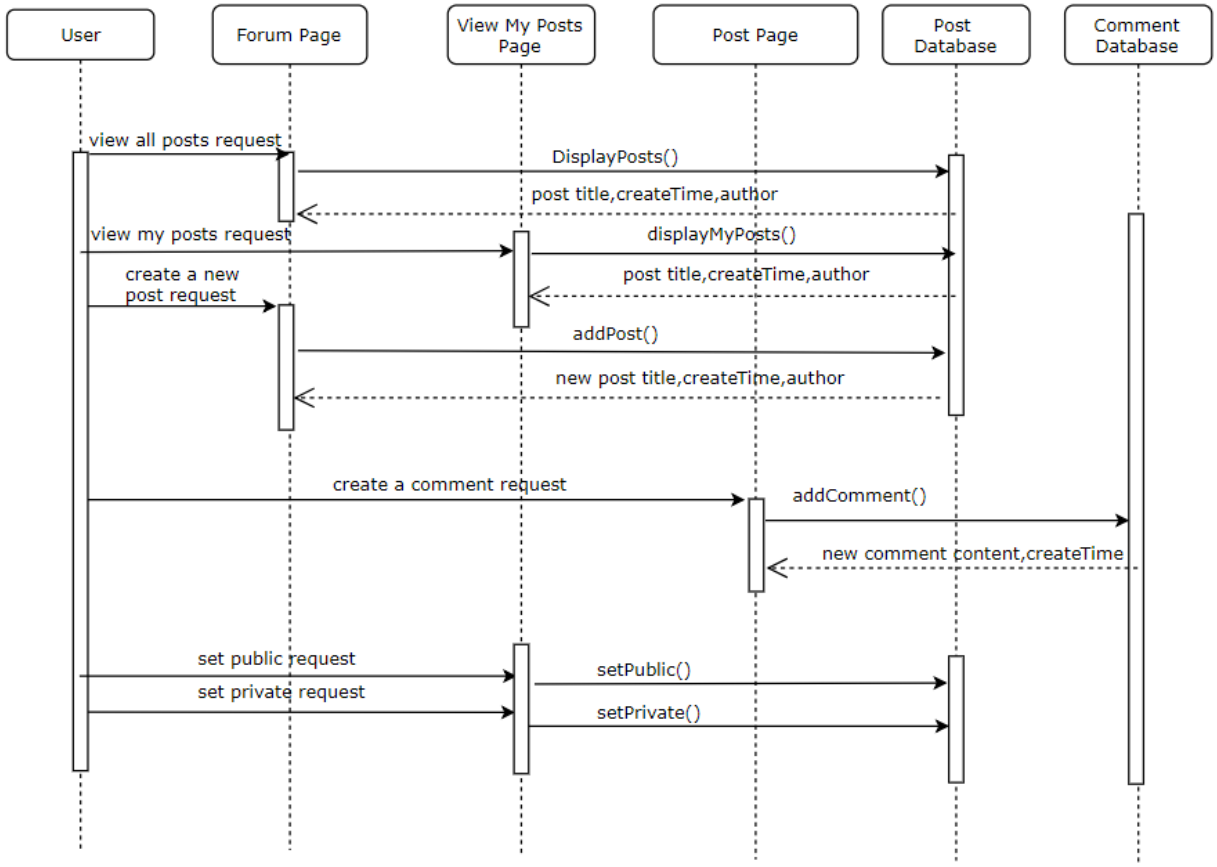


Figure 19 Sequence of Forum System

3.2.2.4 UML Activity Diagram

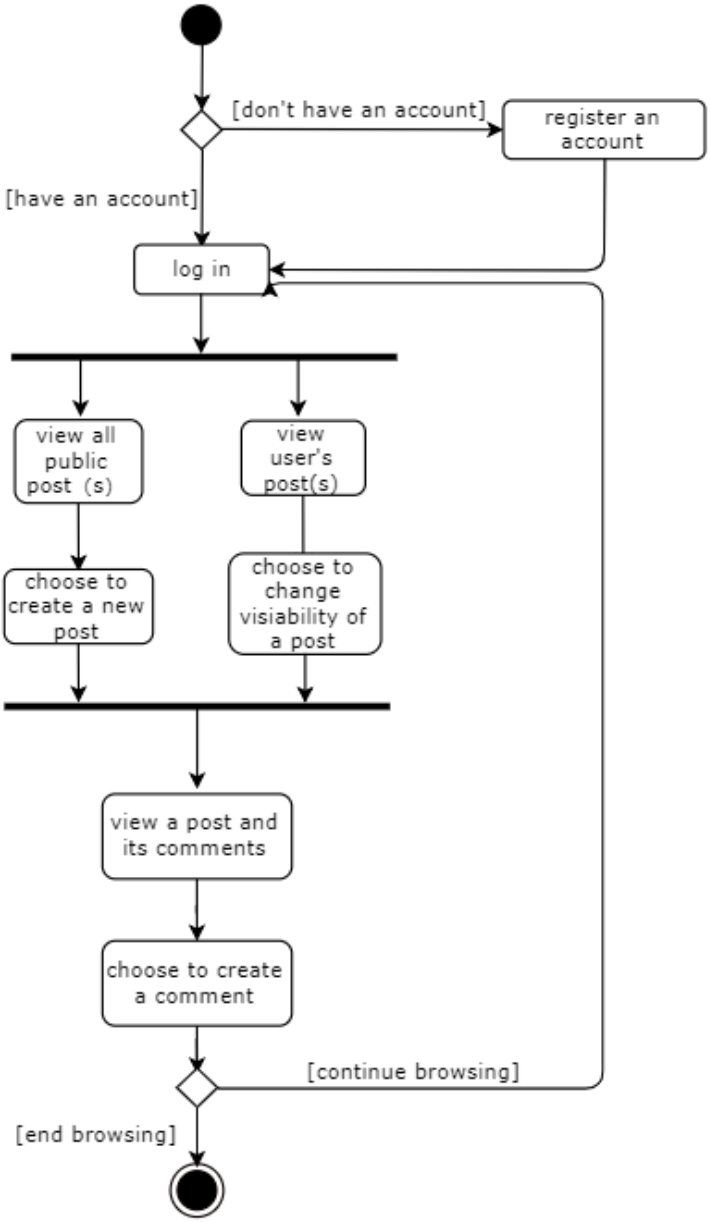


Figure 20 Activity Diagram of Forum System

3.2.3 Functionality

The forum module serves as an open place for CUHK students to share news concerning workshops, internships or even travel team formation, etc. The detailed functions will be stated in 3.2.4.

3.2.4 Procedures and Functions

Function	Parameter	Description
displayPosts	none	Return all public posts from the database
displayMyPosts	userId	Return the user's all posts
addPost	userId	Insert a new post into the "post" database
addReply	userId, postId	Insert a new comment into the "comment" database
getPost	postId	Return all information of the post whose id is identical to postId
getReply	postId	Return all comments of the post whose id is identical to postId
setPrivate	postId	Update the status of the corresponding post to 1
setPublic	postId	Update the status of the corresponding post to 0

*As for the status of a post, default value "0" means public, "1" means private.

The forum system mainly consists of three parts: forum index, individual post, user management. Forum index page provides an overview of all kinds of posts where the user could view all public posts and release a new post. As for the individual post page, users could browse the detailed information and its follow-up discussions. At the meanwhile, they could also release a new follow-up comment. What the user management part concerns are viewing his/her posts and change the visibility of a particular post.

3.3 Chat-room Component

3.3.1 Structural Diagram

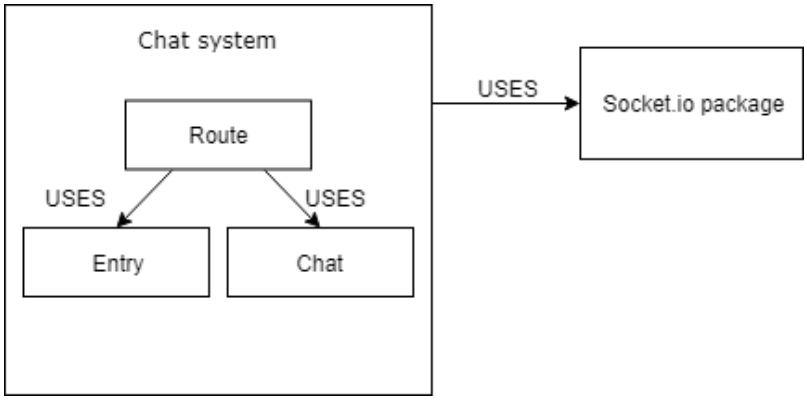


Figure 21 Structure of Chat System

3.3.2 UMLs

3.3.2.1 UML Use-case Diagram

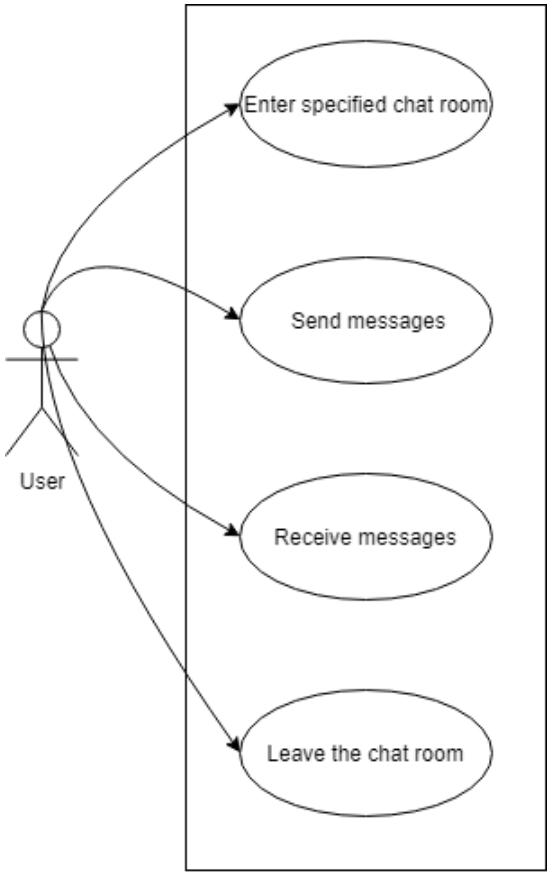


Figure 22 Use-case of Chat System

3.3.2.2 UML Class Diagram

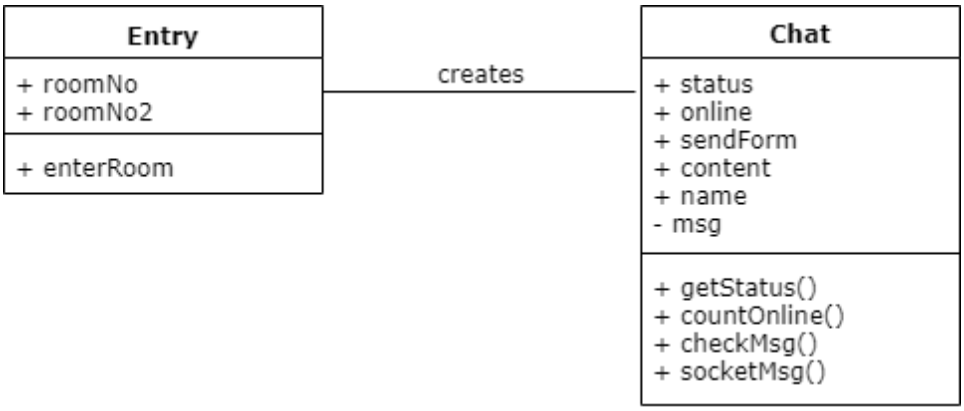


Figure 23 Class of Chat System

3.3.2.3 UML Sequence Diagram

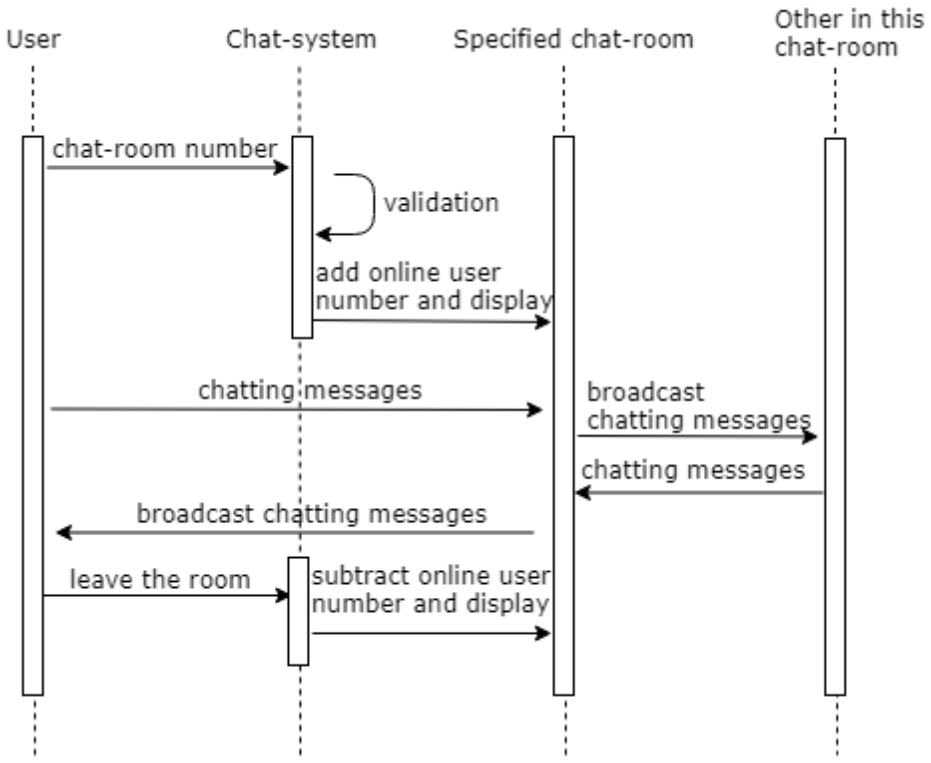


Figure 24 Sequence of Chat System

3.3.2.4 UML Activity Diagram

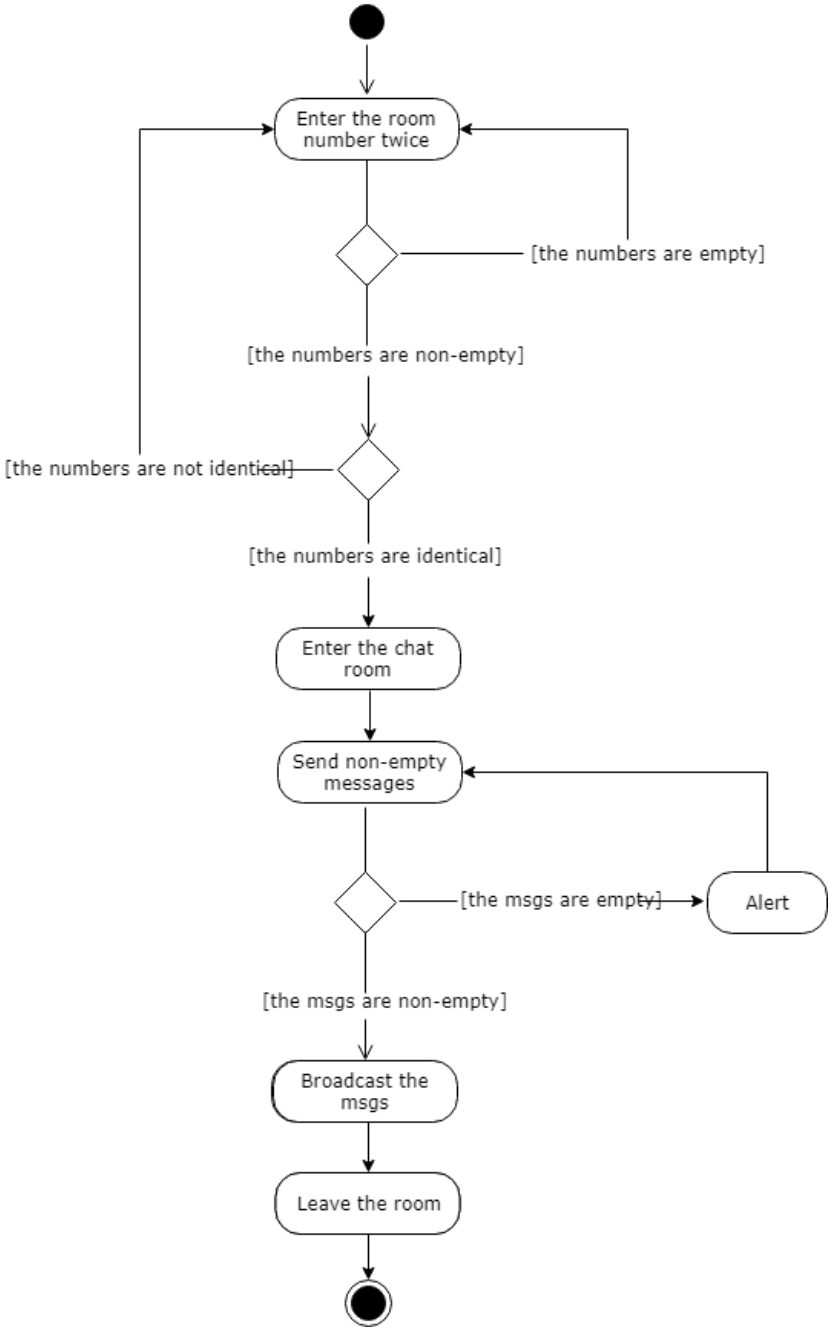


Figure 25 Activity Diagram of Chat System

3.3.3 Functionality

This component handles the chatting part for users after they see the matching result and obtaining a unique room number. The users use the room number to enter a specified chat room

to chat with their potential partner(s). Additionally, this component is an instant real-time chat in order to protect users' privacy.

3.3.4 Procedures and Functions

Function	Description
enterRoom	Triggered when a user tries to enter a room number to get into a specific chat room
socketConnection	Triggered when a user enters the chat room
socketMsg	Triggered when a user tries to send messages
socketDisconnect	Triggered when a user leaves the room

3.4 Roommate Component

3.4.1 Structural Diagram

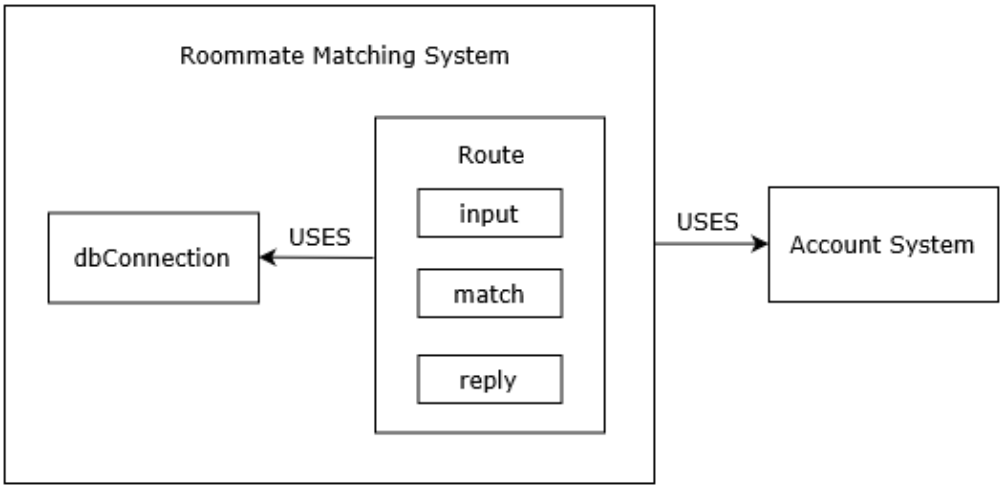


Figure 26 Structure of Roommate System

3.4.2 UMLs

3.4.2.1 UML Use-case Diagram

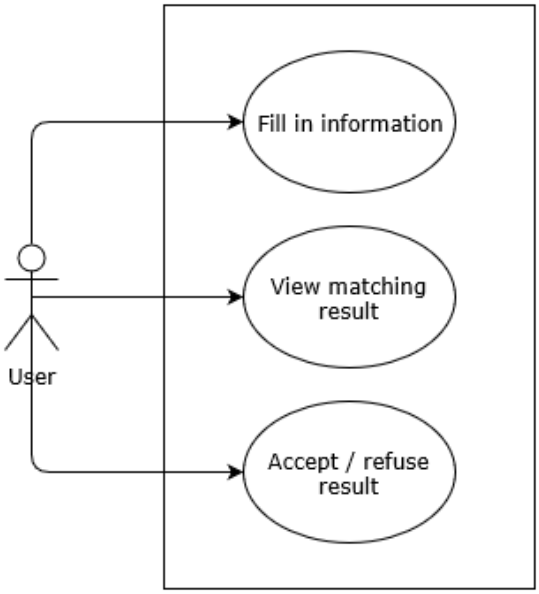


Figure 27 Use-case of Roommate System

3.4.2.2 UML Class Diagram

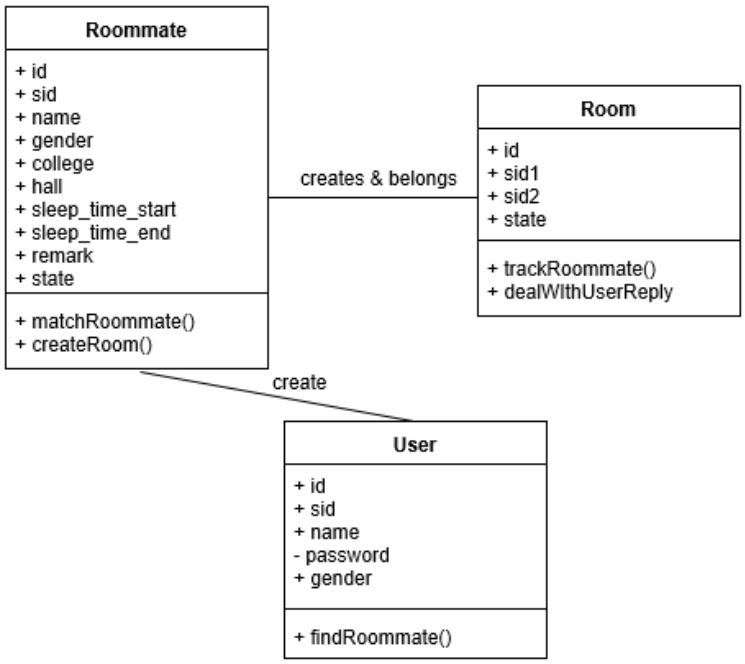


Figure 28 Class of Roommate System

3.4.2.3 UML Sequential Diagram

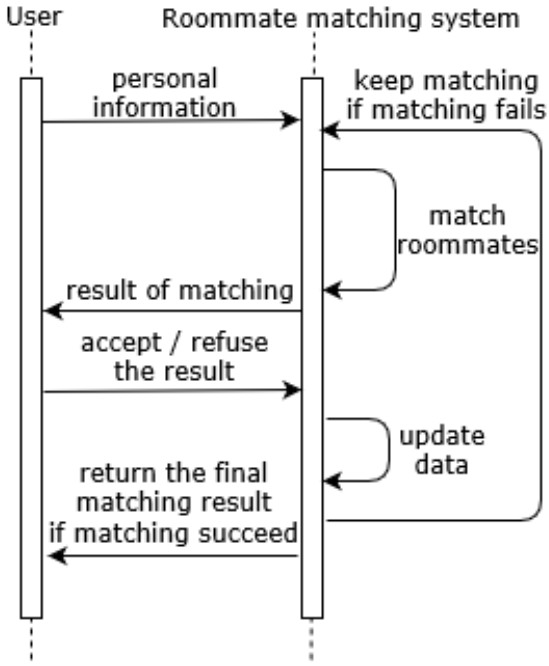


Figure 29 Sequence of Roommate System

3.4.2.4 UML Activity Diagram

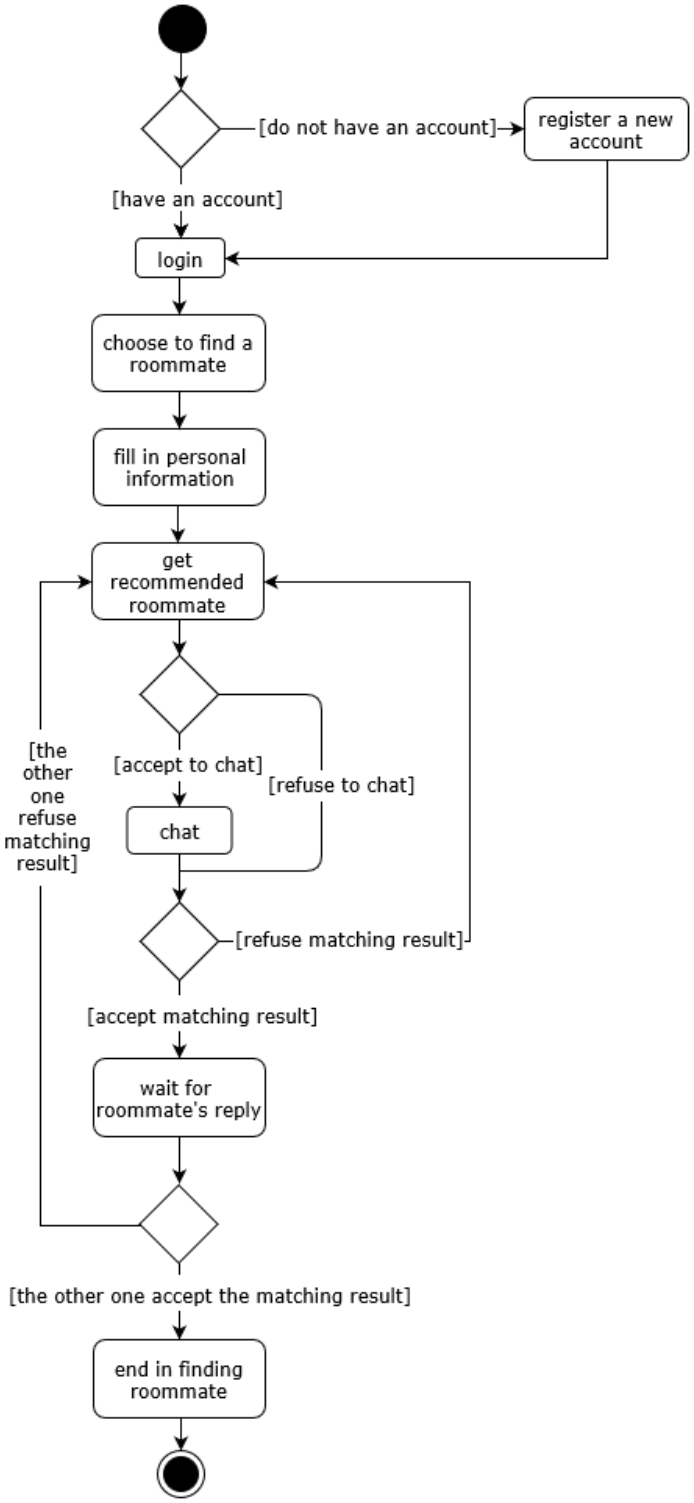


Figure 30 Activity Diagram of Roommate System

3.4.3 Functionality

The functionality of this module is to help the user find a roommate. The system will return a suitable roommate for the user based on his personal information like sex, college, hall, and so on. And details about how the system filter and rank for users are in the below part.

3.4.4 Procedures and Functions

Function	Parameter	Description
findRoommate	sid, sex, college, hall, sleep_time_start, sleep_time_end, remarks,	Users submit the information and start a request
matchRoommate	sid, sex, college, hall, sleep_time_start, sleep_time_end	The system matches a roommate for user
createRoom	sid	The information of two matched users are stored
chaseRoommate	sid	Find the relative roommate information from the room
dealWithUserReply	sid, reply (accept or refuse)	Deal with user's reply (accept or refuse)

In the roommate system, users could fill in the personal information to start a roommate matching request. When the user goes to view the matching result, the roommate system will search for other users whose information could be matched.

If other available users couldn't be found, users would be directed to a page that letting users know the roommate hasn't been founded. Else, the system will choose the fittest one according to their sleeping time, then return the matching result and also the chatting room number for both users to chat with each other.

Then the system will wait for users' reply. If one user refuses the result, the system would search for another user matched when next time the user views the result. If both users accept the result, then the matching is successful.

3.5 Teammate Component

3.5.1 Structural Diagram

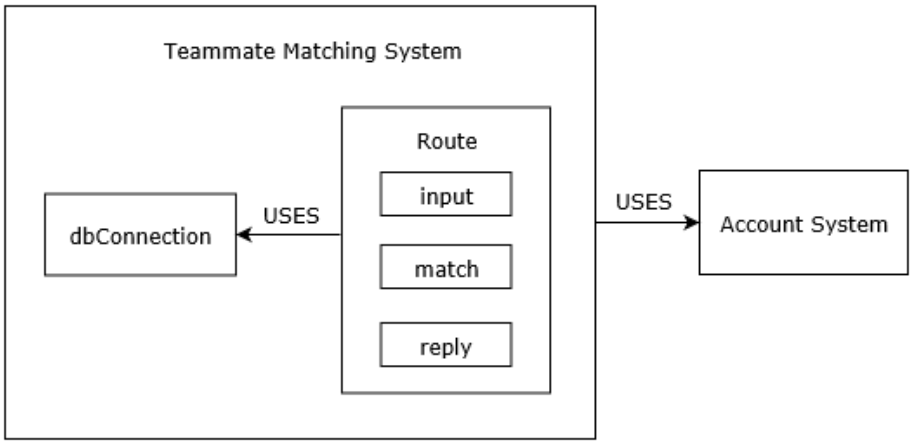


Figure 31 Structure of Teammate System

3.5.2 UMLs

3.5.2.1 UML Use-case Diagram

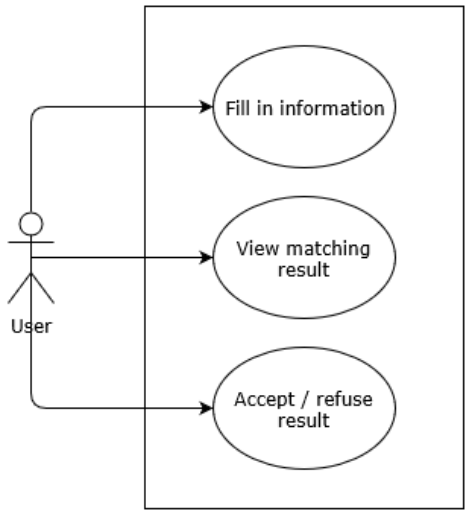


Figure 32 Use-case of Teammate System

3.5.2.2 UML Class Diagram

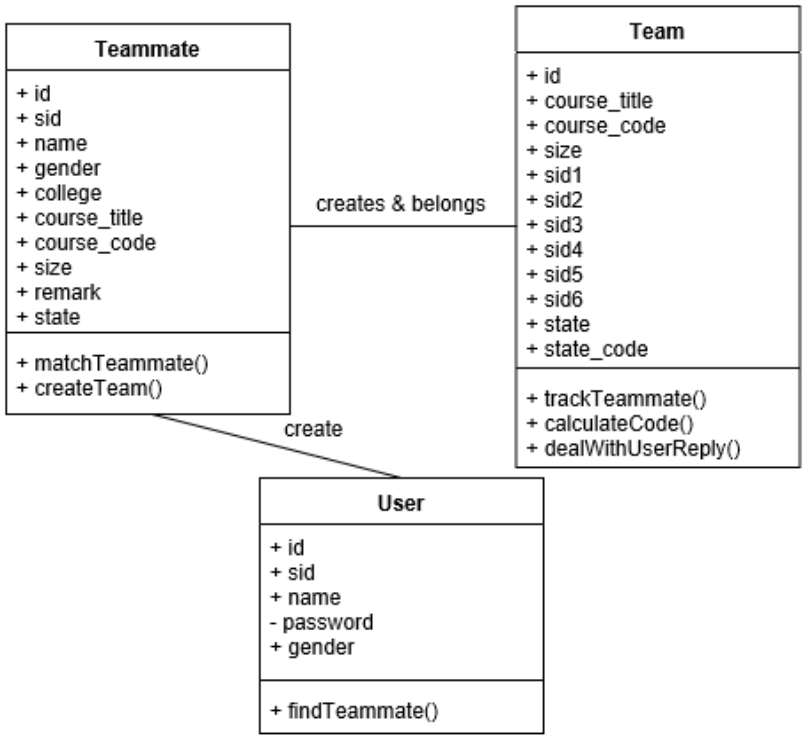


Figure 33 Class of Teammate System

3.5.2.3 UML Sequential Diagram

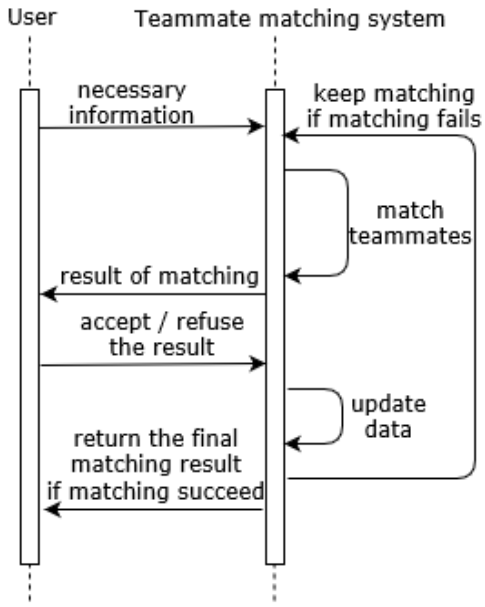


Figure 34 Sequence of Teammate System

3.5.2.4 UML Activity Diagram

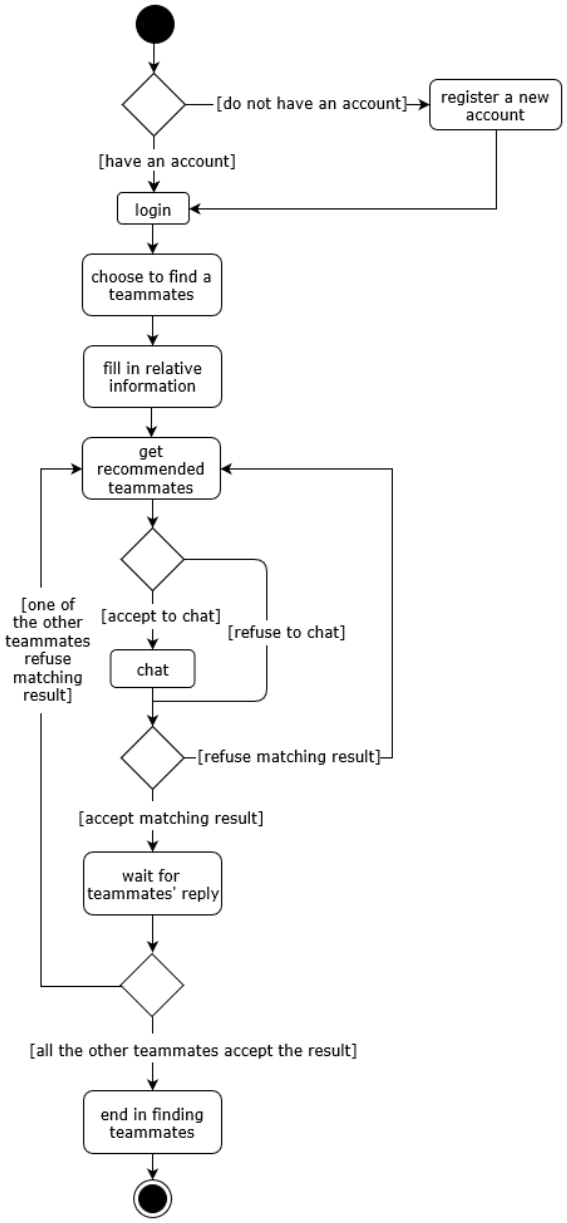


Figure 35 Activity Diagram of Teammate System

3.5.3 Functionality

The functionality of this module is to help the user find teammates. The system will return suitable roommates for the user based on his personal information like course title, course code, size of the team and so on. And details about how the system filter and rank for users are in the below part.

3.5.4 Procedures and Functions

Function	Parameter	Description
findTeammate	sid, sex, college, course_title, course_code, size, remarks,	The user submits the information and start a request
matchTeammate	sid, sex, college, course_title, course_code, size	The system matches a team for the user and adds the user into it
createTeam	sid, course_title, course_code, size	Create an empty team and add the user into it
chaseTeammate	sid, course_title, course_code	Find the relative teammate information from the team
dealwithUserReply	sid, reply(accept or refuse), team_id	Deal with user's reply (accept or refuse)
calculateCode	reply(accept), sid	Calculate the code showing the status about who have accepted the matching result

In the teammate system, users could fill in the personal information to start a teammate matching request. When the user goes to view the matching result, the teammate system will search for teams in the database that match the user's requirement.

If available teams couldn't be found, a new team related to the user's requirement would be constructed. Else, the user would be added into an existing incomplete team. If the team is still incomplete after the user joins it, the user would be directed to a page that letting the user know the teammate hasn't been founded. Else, the system will return the matching result and also the chatting room number for both users to chat with each other.

Then the system will wait for users' reply. If one user refuses the result, the system would search for another team matched when next time the user views the result, and this team would be dismissed. There is a code to show which user has accepted the result, in case that there are users repeat to accept it. If all users accept the result, then the matching is successful.

4 USER INTERFACE DESIGN

4.1 Description of the User Interface

The design refers to two templates with a lot of changes. For login related pages, we chose the simpler template so that the interface will be clearer to the user and the user can easily get the idea of what they need to do. Photos of CUHK are used as a background as the target client of our project is all CUHK students. For function related pages, we add more decorations (pictures and buttons) to the interface to make it more interesting.

To ensure user-friendliness, every page user can click the buttons and go to other pages directly. Also, when the user hasn't registered or hasn't logged in, the landing page allows the user to view our functions.

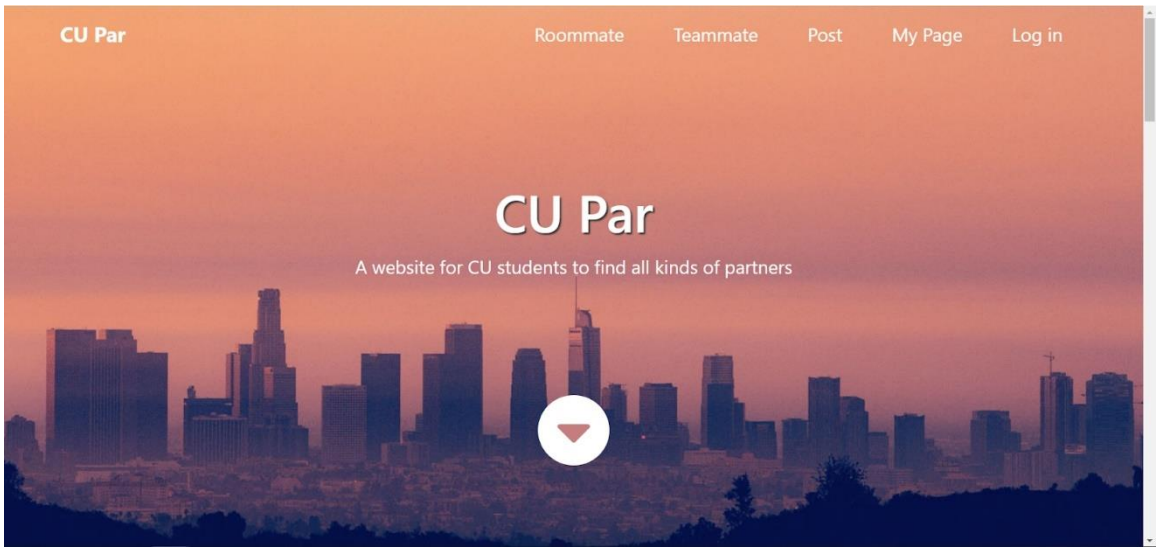
In the same time, to keep privacy, one is not allowed to use the functions (ie. one cannot go to the forum to view the posts) when not logged in. At the landing page, no matter which button user clicked on, he or she would be directed to the login page.

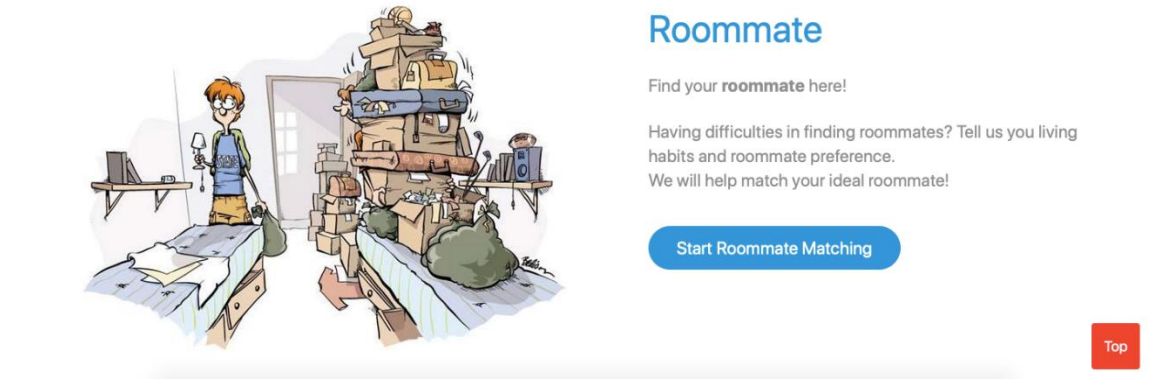
All pages and buttons are designed in a simple and clear way to provide users with an easy understanding when using the website. At the bottom (or right top) of every page, there is the introduction of web designers and the contact information. If the user encounters any problem, he or she can easily get in touch with developers.

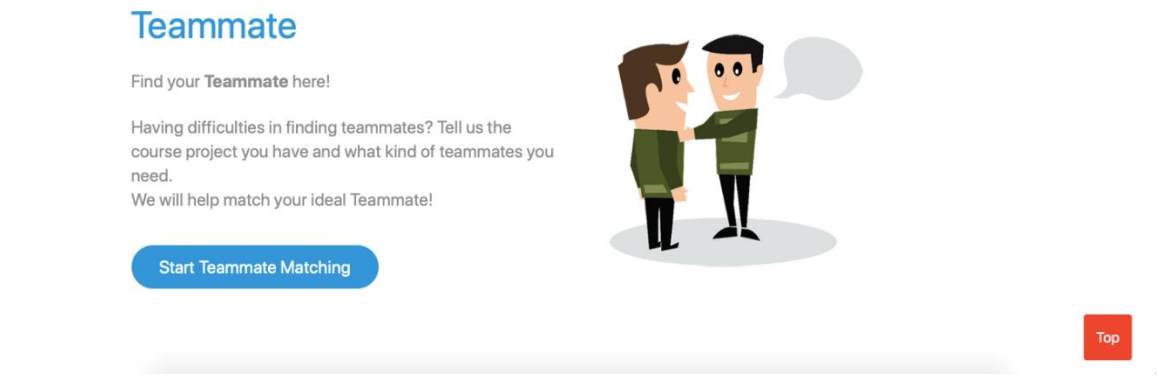
4.2 Objects and Actions with Screen Images

4.2.1 Landing Page

This page is the same as Home Page while it only allows users to view all the functions when not login. If the user clicks on the button, the page will redirect to the login page.


CU Par
Roommate
Teammate
Post
My Page
Log in
CU Par
A website for CU students to find all kinds of partners


Roommate
Find your roommate here!
Having difficulties in finding roommates? Tell us you living habits and roommate preference.
We will help match your ideal roommate!
Start Roommate Matching
Top


Teammate
Find your Teammate here!
Having difficulties in finding teammates? Tell us the course project you have and what kind of teammates you need.
We will help match your ideal Teammate!
Start Teammate Matching
Top

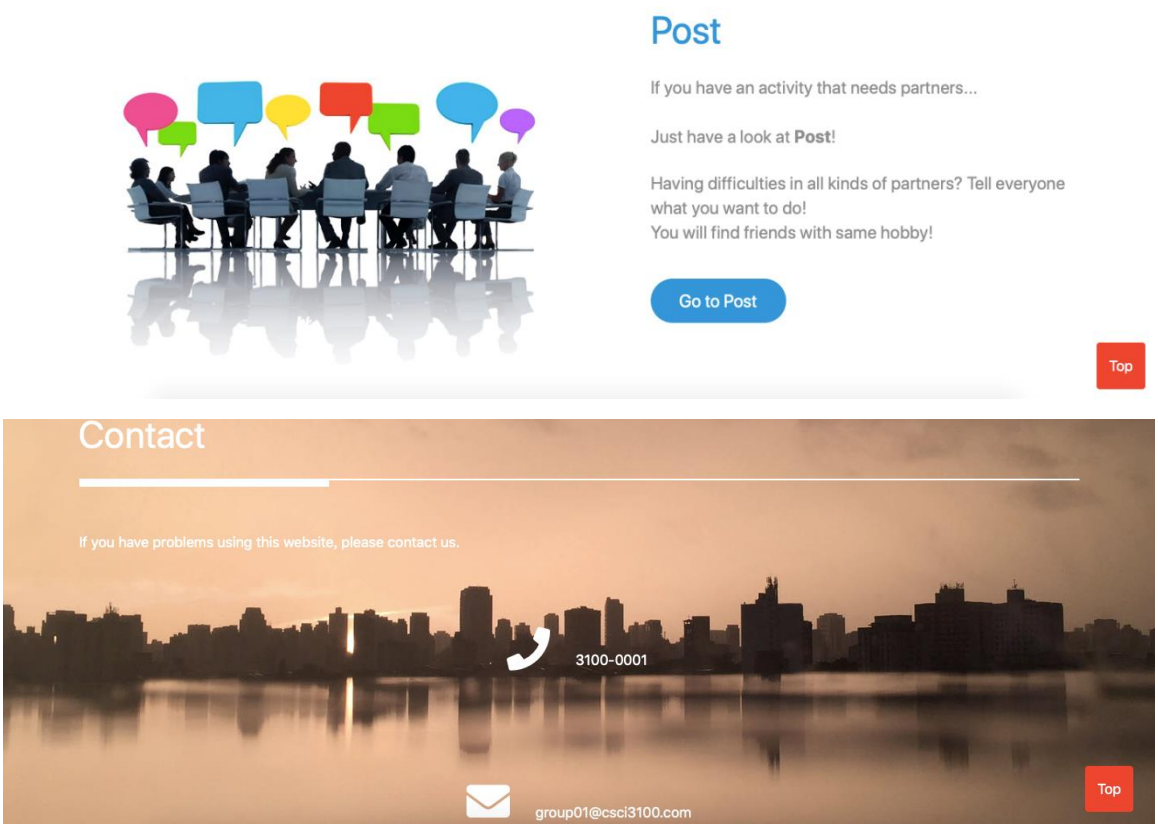


Figure 36 Landing Page

4.2.2 Sign up & Login

User needs to enter his or her CUHK student id number for verification. Then he or she can create a username and password. After the system has checked the user's CUHK identity, an email with verification code will be sent to user's CUHK link email account.

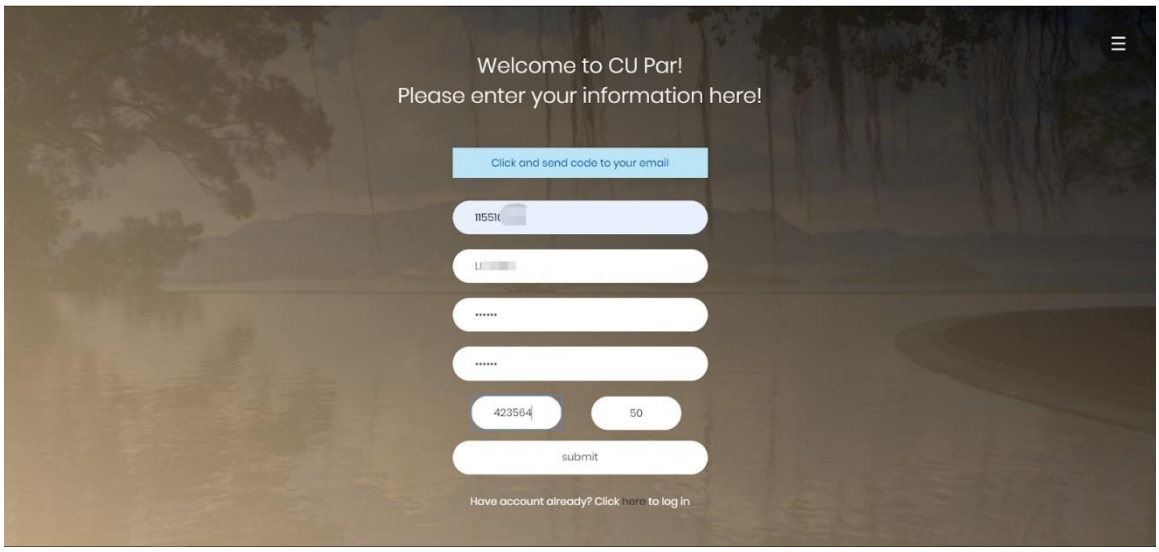


Figure 37 Sign up

If the user has already registered, he or she can log in directly with CUHK sid and CUPar password. If someone doesn't have an account, the login page can be directed to sign up page.

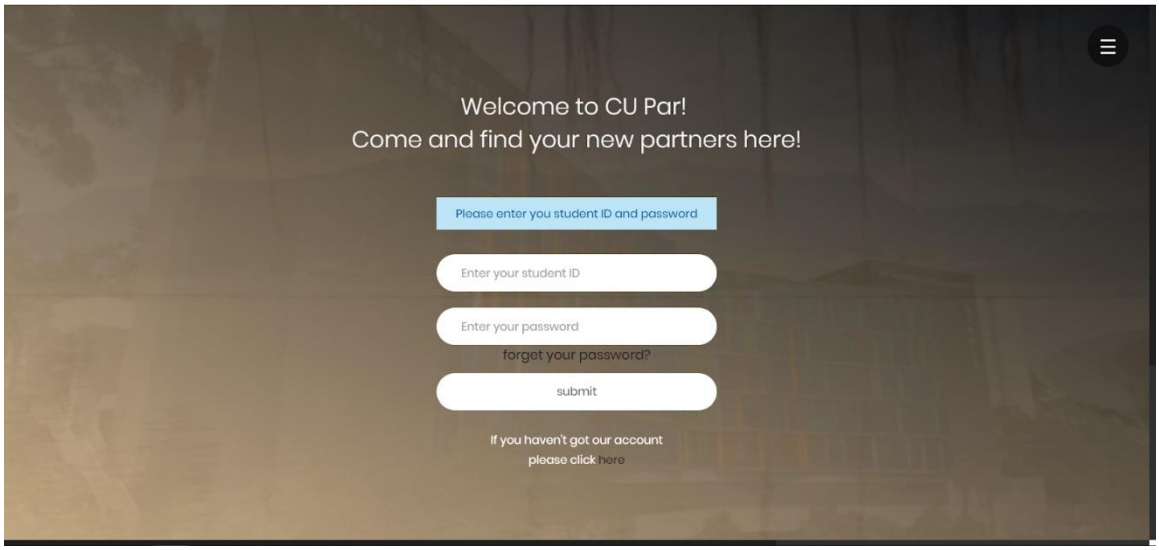


Figure 38 Login

If the user forgets his or her password, he or she can go to the reset password page. An email with verification code will first be sent to user's CUHK link email address. After verification, the user can reset the password.

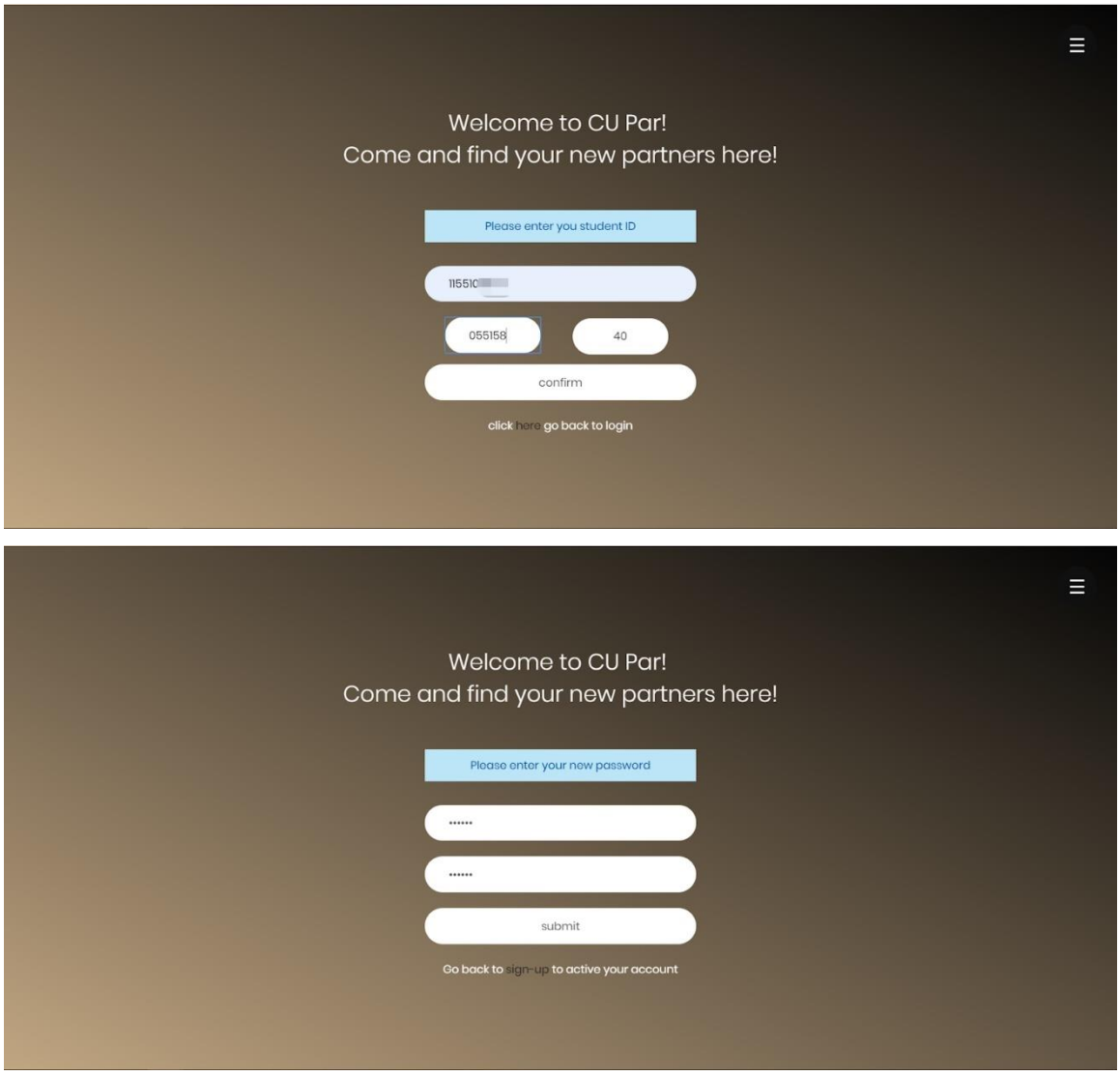


Figure 39 Reset the Password

When there's a problem with the information entered, different alerts will be sent out accordingly. If the process is successful, success notification will also appear.

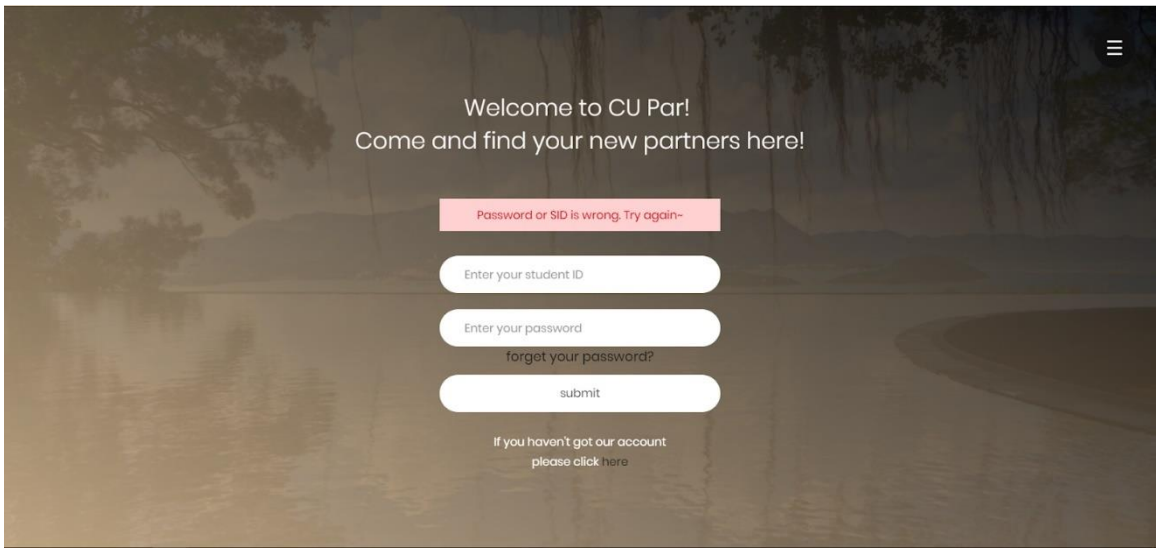


Figure 40 Login

4.2.3 Home Page

After login with CUPar account, the user will be led to Landing Page with his or her username on top. This time, the user can click on the buttons and choose the partners they want to find.

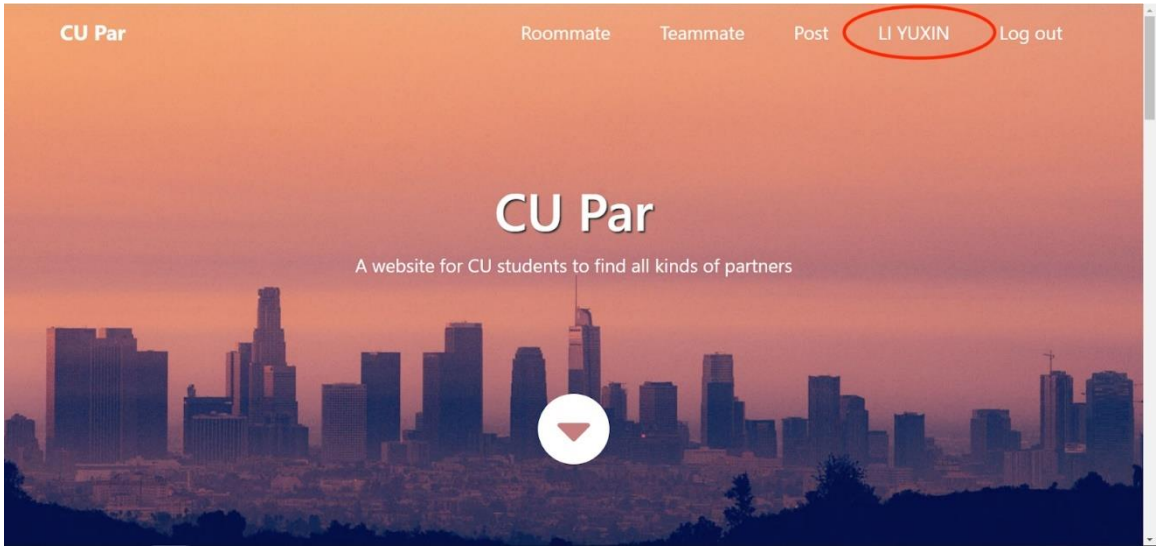


Figure 41 Home Page

There are three main functions: Match Roommate, Match Teammate and Forum. Match Roommate is to help find a suitable roommate. Match Teammate is to help find a suitable

teammate. The forum allows the users to post all kinds of activities that need partners and find interesting activities they want to join.

There's also a MyPage button that can let the user check his or her information as well as match results easily.

4.2.4 Match Roommate Page

If the user wants to find a roommate and clicks the Roommate button in Home Page, he or she will be directed to this page. User needs to fill in the basic information: gender, college, hostel.

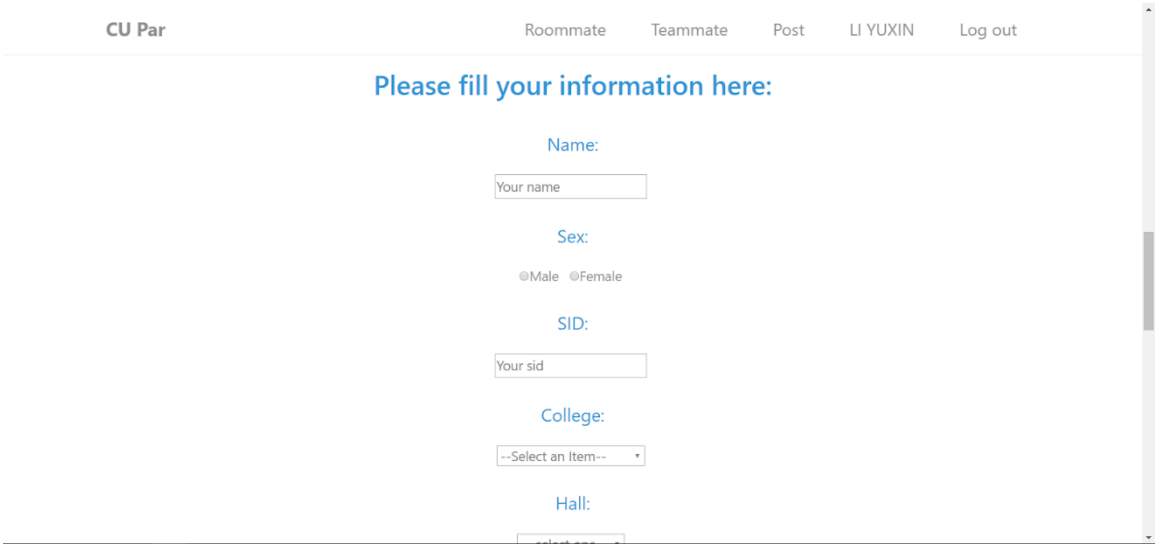


Figure 42 Match Roommate Page

There is also additional information required: sleep time, get up time, and other remarks. The information needed is designed to help achieve the result that students with similar living habits can be matched together.

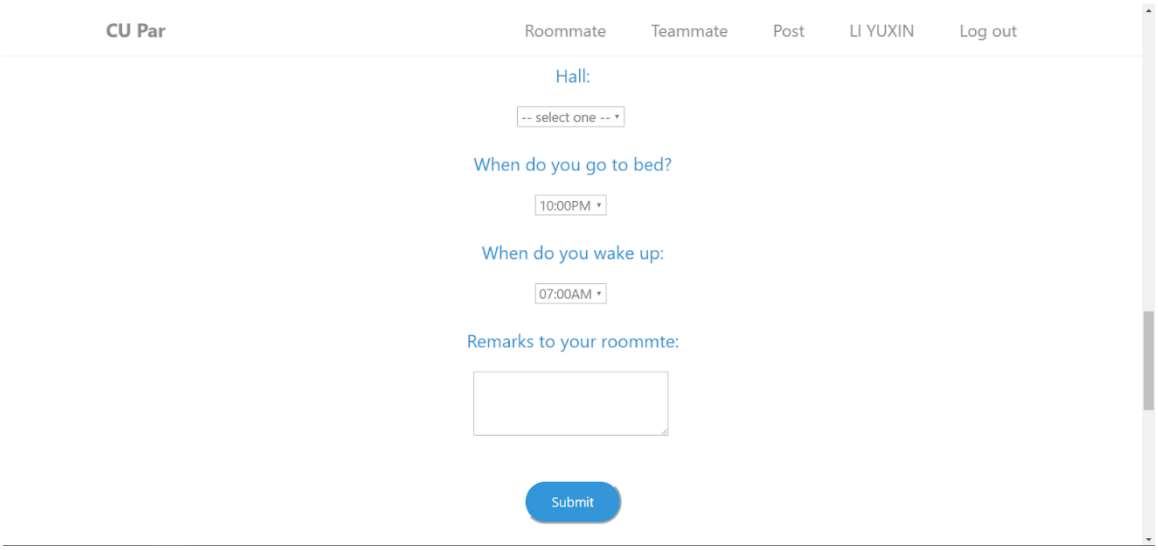


Figure 43 Match Roommate Page

4.2.5 Match Teammate Page

If the user wants to find a teammate and clicks the Teammate button in Home Page, he or she will be directed to this page. User needs to fill in the basic information: college, course, project name, no. of people needed in this project.

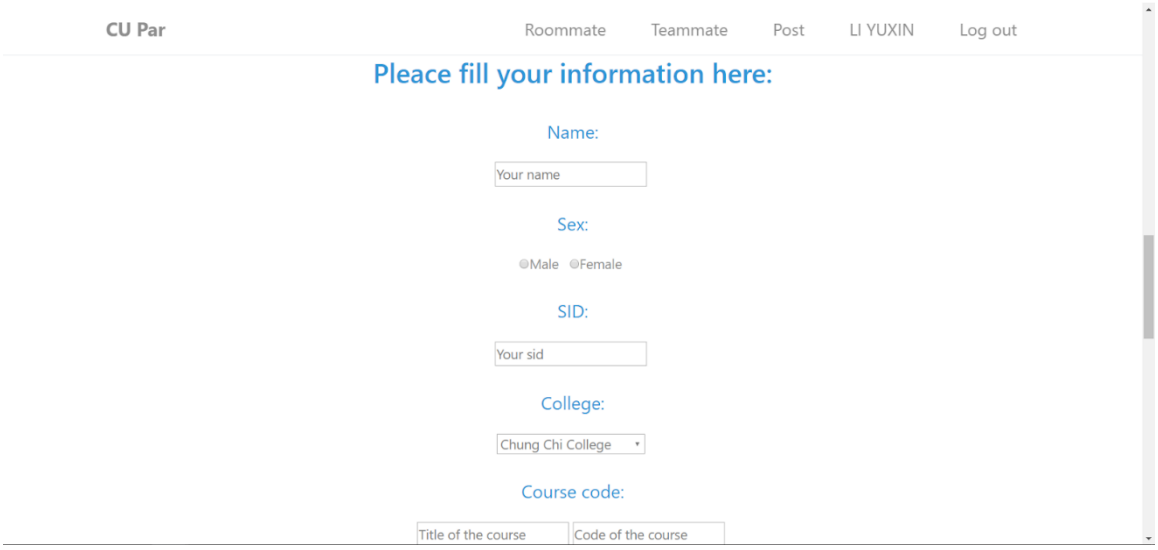


Figure 44 Match Teammate Page

User may also leave additional remarks. Our system will base on how many people needed and return the suitable match result.

4.2.6 Forum

The forum allows users to view and post all kinds of activities they want to find partners. If the user is interested in one of the posts, click the post and view the details. He or she can also leave comments after the post so that the person who created the post will know someone wants to join. When creating a post, it's recommended to put the type of the post (travel, sport, workshop, etc.) at the beginning of the post title. The user could also choose to set his or her own posts private or public at any time. If the user chooses "set private", the post will only be visible to the user.

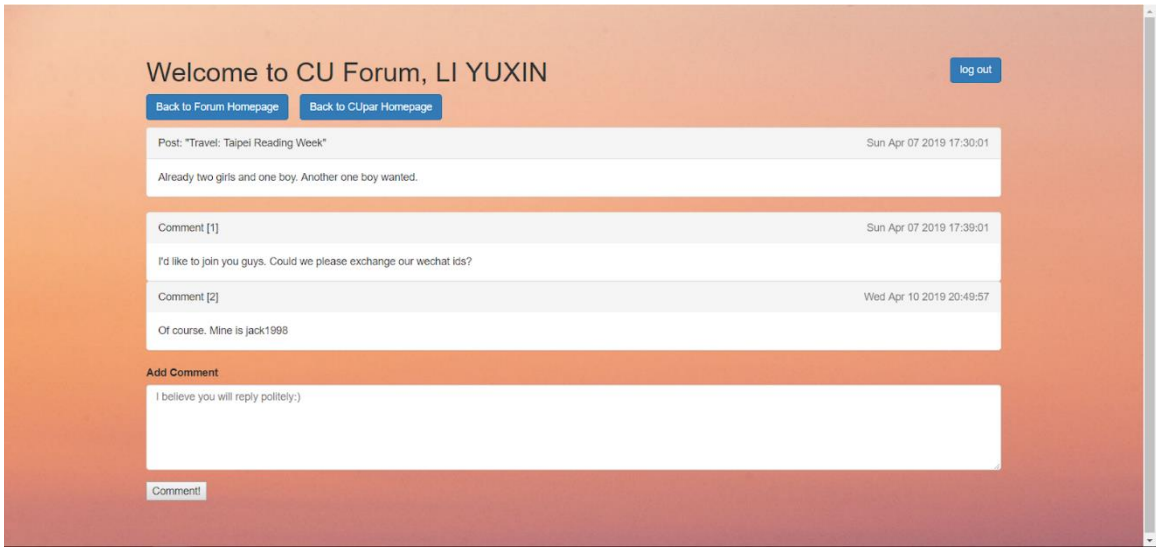


Figure 45 Forum Page (of LI YUXIN)

4.2.7 My Page

In this page, the user can view his or her information and check the result of roommate and teammate matching.

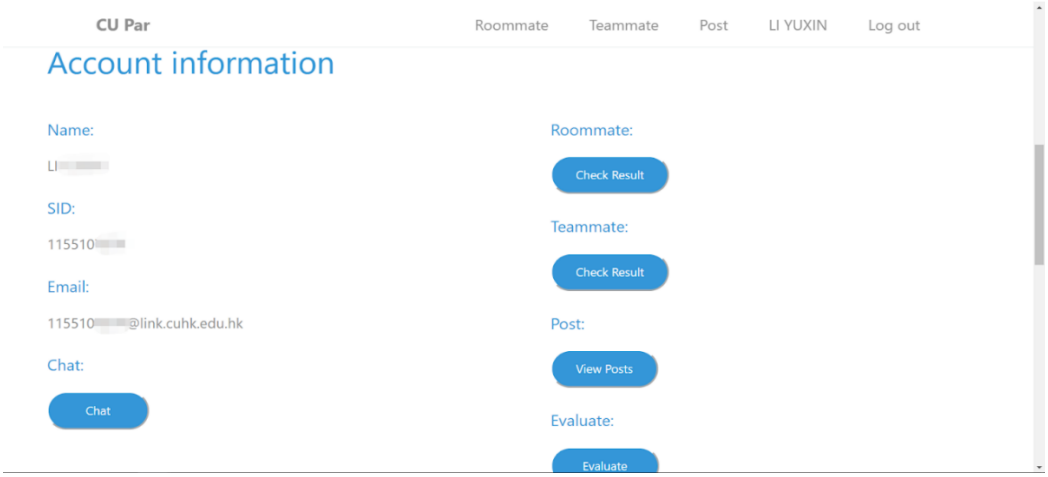


Figure 46 My Page

If the system returns a result, a chat room will be created for the user to chat with his or her potential partner. Click on the "chat" button for chatting (or one may skip this part).

In the chat page, enter the room number and start a chat with a potential partner. For privacy, only when both users are online, the messages from one side are visible to another side.

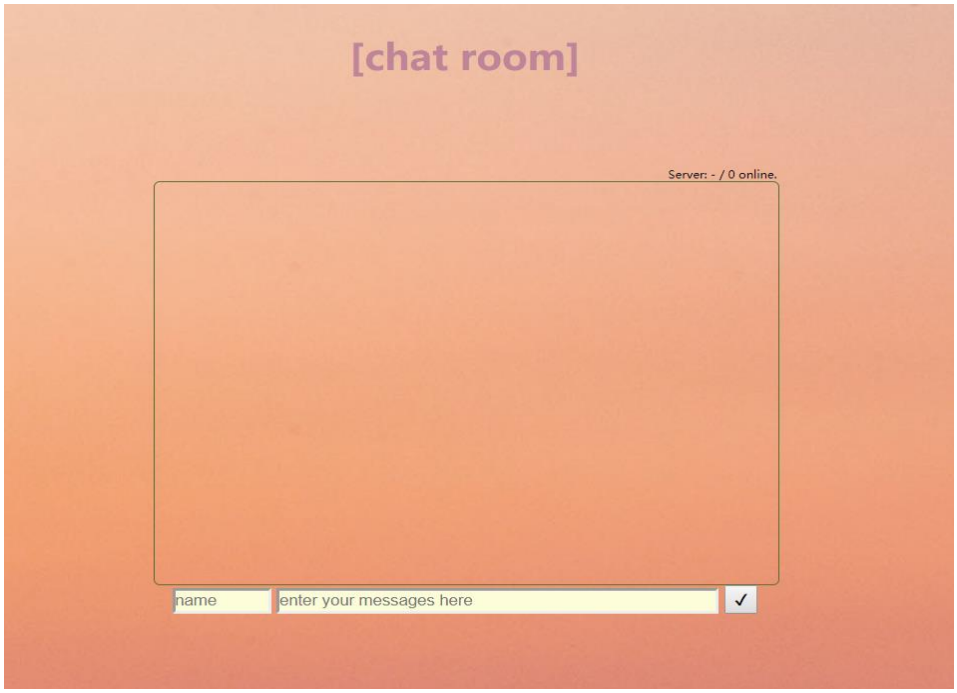


Figure 47 Chat Room

After chatting, if they decide to accept this matching result, choose “accept”. Otherwise, choose “refuse”.

For teammate matching, the user can choose to evaluate his or her teammate by click the “evaluate” button. Enter the partner’s sid and give an evaluation. Choose the star rate report freerider if he or she is. User may also leave additional comments.

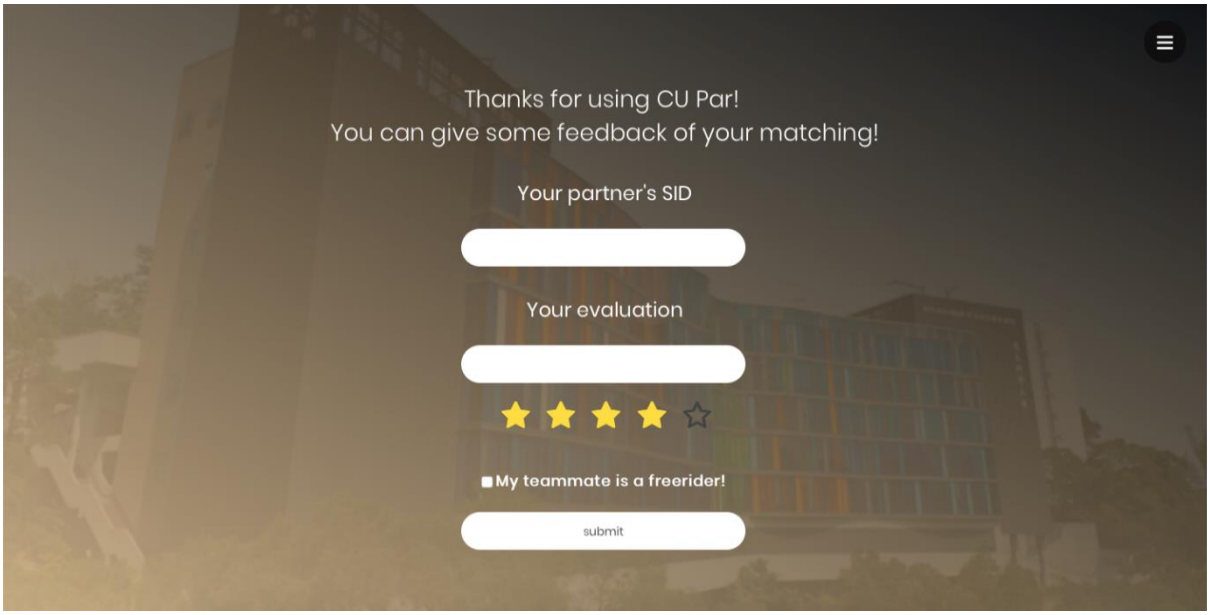


Figure 48 Rate and Feedback

5 TESTS

5.1 Test Overview and Test Plan

As CUPar is a platform with three main functions and a lot of detailed features, our test tries to cover all kinds of possibilities. During the test, we treat the user as a beginner and know nothing about our website. We expect users will input random context and try to simulate different situations. In general, our website should be able to run normally when encountering different problems during testing.

The testing is mainly separated into modular testing and integrated testing. The following test cases are designed base on black box testing. And we also use white box testing to reduce the

number of bugs. We try to cover all the statements of our project. Tested features are Account, Post, Chat, Roommate and Teammate.

Test environment:

- Command (Server Side)
- MySql (Database)
- WorkBench (GUI of Database)
- Chrome (Client Side)
- VS code (Code Editor)

5.2 Case – Account

5.2.1 Purpose

Validate whether it is a complete account system allowing users to log in, sign up, log out, reset password. Verify whether the account system is reliable enough to handle valid operations, refuse invalid operations and protect users' private and prevent hacking.

5.2.2 Inputs

Case1:

directly access the index without signing up or logging in

Case2:

directly access the page except for the index, login and signup page without logging in or signing up.

As for signup on the sign-up page: (You should log out first)

Case3:

SID: 1144100000 Username: Ryan Password: zxczxc Code: 123456

PasswordRP: zxczxc

Case4:

SID: 115510000 Username: Ryan Password: zxczxc Code: 123456

PasswordRP: 123456

Case5:
SID: 1155001111 Username: Michael Password: zxczxc PasswordRP: zxczxc click the send button

Case6: (use your link-email to signup)
SID: 1155107874(Your sid) Username: Ll Yuxin Password:lyxnb2333 PasswordRP: lyxnb2333 click the send button Code: 00000(different from the code you received)

Case7: (use your link-email to signup)
SID: 1155107874(Your sid) Username: Ll Yuxin Password:lyxnb2333 PasswordRP: lyxnb2333 click the send button Code: 00000(The code you received)

As for login on the login page: (You should log out first)
Case8:
SID: 1155109999 Password: zxczxc

Case9:
SID: 1155100000 Password: zxczx9

Case10:
SID: 1155107875 Password: lyxnb2333

Case11:
SID: 1155100000 Password: zxczxc

As for reset password on reset page: (You should register first)
Case12:
SID: 1155109999 click the send button

Case13:

SID: 1155107875 click the send button

Case14:

SID: 115510xxxx (You have registered before) click the send button Code: 000000(you received)

As for reset password on the resetPwd page:(You should pass the reset page)

Case15:

Password: asdasd PasswordRP: asdasc

Case16:

Password: asdasd PasswordRP: asdasd

5.2.3 Expected Outputs & Pass/Fail Criteria

Case1: The index shall be accessed but the button on the upper right is MyPage

Case2: The pages cannot be accessed and the window is redirected to the login page

Case3: It shall remind of “SID is invalid!”

Case4: It shall remind of “Two passwords are not the same!”

Case5: It shall remind of “You have registered, please log in”. The button of sending email is unavailable for 60 seconds. Each time click the button is the same. The following cases will omit the performance of the button.

Case6: It shall remind of “Wrong code, check or send it again”

Case7: It shall sign up successfully and redirect to the index page. All pages are available, and the button on the upper right on each page is LI Yuxin.

Case8: It shall remind of “You have not registered yet, please sign up”

Case9: It shall remind of “Password or sid is wrong, try again~”

Case10: It shall remind of “Account has not been activated, please return to sign-up page”

Case11: It shall log in successfully and redirect to the index page. All pages are available, the button on the upper right on each page is Tester 2

Case12: It shall remind of “You have registered yet, please sign up”

Case13: It shall remind of “Account has not been activated, please return to sign-up page”

Case14: It shall redirect to the resetPwd page.

Case15: It shall remind of “Please confirm your password”

Case16: It shall redirect to the login page and remind of “reset successfully, please log in”

5.3 Case – Forum

5.3.1 Purpose

Check if the forum system could support the function such as display all posts, display user posts, release a new post, release a new comment as well as edit the visibility. Test whether the main functions could run reliably given the extreme conditions.

5.3.2 Inputs

Case1: enter forum index page conditioning no posts in the database

Case2: enter forum index page conditioning there are 4 posts in the database

Case3: enter “view my posts” page conditioning that the user hasn’t released any post

Case4: add a new post at the forum index page with no input in title or content

Case5: add a new post at forum index page without empty input

Case6: click a particular post

Case7: add a new empty comment at the particular post page where there are no comments yet

Case8: add a new non-empty comment at the particular post page where there’s no comment yet

Case9: add a new non-empty comment at the particular post page where there are 3 comments

Case10: click “setPrivate” button of the first post’s in “view my posts” page and then click “Back to Forum Homepage”

Case 11: click “setPublic” button of the first post in “view my posts” page and then click “Back to Forum Homepage” (conditioning on case10)

5.3.3 Expected Outputs & Pass/Fail Criteria

Case1: alert “no posts” information and the forum index page display nothing but input region for a new post

Case2: All four posts are shown from top to bottom according to time sequence and there exists an input region in the bottom.

Case3: the “view my posts” page shows “no post, add one!” where clicking the “add one” could redirect to forum index page to add a new post

Case4: alert “input can’t be empty” and keep the input that the user has typed in

Case5: the post shows up within the forum index page and the inputs are removed

Case6: redirect to the particular post page and post details and follow-up comments are shown from top to bottom according to time sequence.

Case7: alert “input can’t be empty”

Case8: the comment shows up with comment id 1 and comments content and createTime below the post content and the input is removed

Case9: the comment shows up with comment id 4 and comments content and createTime below the post content and the input is removed

Case10: alert “set successfully” information and the original “setPrivate” button changes to “setPublic”, redirect to the forum index page and the edited post vanished

Case 11: alert “set successfully” information and the original “setPublic” button changes to “setPrivate”, redirect to the forum index page and the edited post appeared

5.4 Case - Chat

5.4.1 Purpose

Test whether users can use this chat-room to chat with their potential partners (roommates or teammates) secretly and synchronously.

5.4.2 Inputs

Enter the room number:

Case1: 1&none

Case2: none&1

Case3: 1&2

Case4: 1&1

Chat in the room:

Case5: Name: Esther Msgs:

Case6: Name: Msgs: hi~

Case7: Name: Esther Msgs: hi~

5.4.3 Expected Outputs & Pass/Fail Criteria

Case1 & 2: “please enter the room number” warning pops out.

Case3: “check the number you enter~ they do not match” warning pops out.

Case4: Users can enter the chat room successfully.

Case5 & 6: The messages cannot be sent out. And the text box will be red.

Case7: Use can send this message successfully and this will be broadcast to all users who are in the same chat room.

No blank input and different inputs of chat-room number are allowed. Only users with the matched chat-room number can have a chat in the chat room. In addition, the messages are instant in order to protect users’ secret.

5.5 Case - Roommate System

5.5.1 Purpose

Test whether users' information about finding roommates can be checked and stored correctly, and whether the server can match the roommates, return the matching result and deal with users' reply correctly.

5.5.2 Roommate System - Storing information

5.5.2.1 Inputs

Case1:

Name: (blank) Sex: Male SID: (blank) College: (blank) Hall: (blank)

Time (go to bed): 10:00 PM Time (wake up): 07:00 AM Remarks: (blank)

Case2:

Name: (blank) Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Case3:

Name: Jack Sex: Male SID: 1155107819 College: (blank) Hall: (blank)

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Case4:

Name: Jack Sex: Male SID: 115510 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Case5:

Name: Jack Sex: Male SID: 1234567890 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Case6:

Name: Jack Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

(This input has submitted before.)

Case7:

Name: Jack Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

5.5.2.2 Expected Outputs & Pass/Fail Criteria

Case1: The “Your SID is wrong!” warning pops out.

Case2: The “Please fill in all your necessary information!” warning pops out.

Case3: The “Please fill in all your necessary information!” warning pops out.

Case4: The “Your SID is wrong!” warning pops out.

Case5: The “Your SID is wrong!” warning pops out.

Case6: The “You have already filled your information!” warning pops out.

Case7: The input is legal. An alert with “Successful submit!” pops out and page is refreshed.

No blank in any field except remark, and the form of SID should be legal, which means there should be 10 digits in SID and the first 4 digits are 1155, and no repeated submission by the same user is allowed.

Additionally, there are default values for sex, Time (go to bed) and Time (wake up), which are ‘Male’, 10:00 PM and 07:00 AM, so these 3 values won’t be blank. And only after you choose the college, the user can choose the hall belongs to the college. In this situation, there will be a default value of the hall after the user chooses the college, which is different according to colleges.

5.5.3 Roommate System - Matching Information and Returning the Matching Result

5.5.3.1 Inputs

Below inputs describe the situation of the user and the database,

Case1:

The user hasn't submitted his/her personal information.

Case2:

User's information:

Name: Jack Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Database:

There isn't any other users' information in the database.

Case3:

User's information:

Name: Jack Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Database:

There is other users' information in the database. However, no users in the database that satisfies such conditions: Sex-Male, College-S.H.Ho, Hall-LQW.

Case4:

User's information:

Name: Jack Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Database:

There is one user whose information is:

Name: Louis Sex: Male SID: 1155106385 College: S.H.Ho Hall: LQW

Time (go to bed): 12:00 PM Time (wake up): 08:30 AM Remarks: Hi!

Case5:

User's information:

Name: Jack Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Database:

There is one user whose information is:

Name: Louis Sex: Male SID: 1155106385 College: S.H.Ho Hall: LQW

Time (go to bed): 12:00 PM Time (wake up): 08:30 AM Remarks: Hi!

However, these 2 users have matched before and one user refuses that matching.

Case6:

User's information:

Name: Jack Sex: Male SID: 1155107819 College: S.H.Ho Hall: LQW

Time (go to bed): 11:30 PM Time (wake up): 08:00 AM Remarks: Hello!

Database:

There are 2 users whose information is:

Name: Louis Sex: Male SID: 1155106385 College: S.H.Ho Hall: LQW

Time (go to bed): 12:00 PM Time (wake up): 08:30 AM Remarks: Hi!

Name: Tom Sex: Male SID: 1155105239 College: S.H.Ho Hall: LQW

Time (go to bed): 12:30 PM Time (wake up): 09:00 AM Remarks: Hi!

5.5.3.2 Expected Outputs & Pass/Fail Criteria

Case1-3,5: When the user checks the matching result, the user is directed to a page which shows "Sorry, we haven't found your roommate. You could check the result a few hours later."

Case4,6: When the user checks the matching result, the user is directed to a page which shows "Name: Louis Sex: Male SID: 1155106385 Remarks: Hi! ". And the status of the matching is "waiting for reply".

Only until there is another user in the database, whose sex and college and hall are the same as the user, then the user would be matched. Else, the user needs to wait for other users to satisfy the conditions show up.

If there are at least 2 users who satisfy the conditions, then the system will return the user whose sleeping time coincides most with the user. If some of the roommates' sleeping time coincide in the same way, then the system would choose the user whose submit the information earlier.

What's more, if the user has refused or been refused by one roommate, then that roommate won't be matched for the user again.

5.5.4 Roommate System - Dealing Users' Reply

5.5.4.1 Inputs

Case1: Refuse the matching result

Case2: Accept the matching result and then another roommate refuses it

Case3: Accept the matching result and another roommate accepts it

Case4: Accept or refuse the result when the status is "success"

5.5.4.2 Expected Outputs & Pass/Fail Criteria

Case1: The user can't see the matching result anymore until a new roommate is matched.

Case2: After the user accepts the result, the user can still check the matching result and its status is still "waiting for reply" until another user refuses it. Then the user can't see the matching result anymore until a new roommate is matched.

Case3: After the user accepts the result, the user can still check the matching result. Its status is still 'waiting for reply' until both roommates accept it, then the status will be changed into "Success".

Case4: The matching result and status don't change.

If one of user refuse refuses the result, then the matching result will be removed and the matching would restart. Else, the status of the matching would be changed into “success” and then the matching can’t be changed anymore.

5.6 Case - Teammate System

5.6.1 Purpose

Test whether users’ information about finding teammates can be checked and stored correctly, and whether the server can match the teammates, return the matching result and deal with users’ reply correctly.

5.6.2 Teammate System - Storing information

5.6.2.1 Inputs

Case1:

Name: (blank) Sex: male SID: (blank) College: CC Size: (blank)

Course_Title: (blank) Course_Code: (blank) Remarks: (blank)

Case2:

Name: Jack Sex: male SID: 1155107819 College: CC Size: 3

Course_Title: (blank) Course_Code: (blank) Remarks: (blank)

Case3:

Name: (blank) Sex: male SID: 1155107819 College: CC Size: (blank)

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Case4:

Name: Jack Sex: male SID: 115510 College: CC Size: 3

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Case5:

Name: Jack Sex: male SID: 1234567890 College: CC Size: 3
Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Case6:
Name: Jack Sex: male SID: 1234567890 College: CC Size: 3
Course_Title: CS Course_Code: 3100 Remarks: (blank)

Case7:
Name: Jack Sex: male SID: 1234567890 College: CC Size: 3
Course_Title: CSCI Course_Code: 310 Remarks: (blank)

Case8:
Name: Jack Sex: male SID: 1234567890 College: CC Size: 3
Course_Title: CSCI Course_Code: 0031 Remarks: (blank)

Case9:
Name: Jack Sex: male SID: 1234567890 College: CC Size: 1
Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Case10:
Name: Jack Sex: male SID: 1234567890 College: CC Size: 3
Course_Title: CSCI Course_Code: 3100 Remarks: (blank)
(This input has submitted before.)

Case11:
Name: Jack Sex: male SID: 1234567890 College: CC Size: 3
Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

5.6.2.2 Expected Outputs & Pass/Fail Criteria

Case1: The “Your SID is wrong!” warning pops out.

Case2: The “Please fill in all your necessary information!” warning pops out.

Case3: The “Please fill in all your necessary information!” warning pops out.

Case4: The “Your SID is wrong!” warning pops out.

Case5: The “Your SID is wrong!” warning pops out.

Case6: The “Course title should be 4 letters!” warning pops out.

Case7: The “Course code should be 4 digits!” warning pops out.

Case8: The “First digit of the code should between 1~6!” warning pops out.

Case9: The “Size of your team should between 2~6!” warning pops out.

Case10: The “You have already filled your information!” warning pops out.

Case11: The input is legal. An alert with “Successful submit!” pops out and page is refreshed.

No blank in any field except remark, and the form of SID should be legal, which means there should be 10 digits in SID and the first 4 digits are 1155. The size of the team should be between 2 and 6. Course title should be 4 letters. Course code should be 4 digits and the first digit should be between 1 and 6.

Additionally, there are default values for sex and college, which are ‘Male’ and CC, so these 2 values won’t be blank. And no repeated submission by the same user is allowed.

5.6.3 Teammate System - Matching Information and Returning the Matching Result

5.6.3.1 Inputs

Below inputs describe the situation of the user and the database,

Case1:

The user hasn't submitted his/her personal information.

Case2:

User's information:

Name: Jack Sex: male SID: 1234567890 College: CC Size: 3

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Database:

There isn't any team's information in the database.

Case3:

User's information:

Name: Jack Sex: male SID: 1234567890 College: CC Size: 3

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Database:

There is other teams' information in the database. However, no team in the database satisfies such conditions: Size-3, Couese_Title-CSCI, Course_Cose-3100.

Case4:

User's information:

Name: Jack Sex: male SID: 1234567890 College: CC Size: 3

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Database:

There is one team matched whose information is:

Team_id: 3 SID1: 1155106385 SID2: 1155106634

SID3:1155108075 Size: 3 Now_Size:3

Course_Title: CSCI Course_Code: 3100

Case5:

User's information:

Name: Jack Sex: male SID: 1234567890 College: CC Size: 3

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Database:

There is one team matched whose information is:

Team_id: 3 SID1: 1155106385 SID2: (blank)

SID3:(blank) Size: 3 Now_Size: 1

Course_Title: CSCI Course_Code: 3100

Case6:

User's information:

Name: Jack Sex: male SID: 1234567890 College: CC Size: 3

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Database:

There is one team matched whose information is:

Team_id: 3 SID1: 1155107819 SID2: (blank)

SID3:(blank) Size: 3 Now_Size: 1

Course_Title: CSCI Course_Code: 3100

Case7:

User's information:

Name: Jack Sex: male SID: 1234567890 College: CC Size: 3

Course_Title: CSCI Course_Code: 3100 Remarks: (blank)

Database:

There is one team matched whose information is:

Team_id: 3 SID1: 1155106385 SID2: 1155106634

SID3:(blank) Size: 3 Now_Size: 2

Course_Title: CSCI Course_Code: 3100

5.6.3.2 Expected Outputs & Pass/Fail Criteria

Case1-6: When the user checks the matching result, the user is directed to a page which shows "Sorry, we haven't found your teammate. You could check the result a few hours later."

Case7: When the user checks the matching result, the user is directed to a page which shows relative information of 2 users whose SID are 1155106385 and 1155106634. And the status of the matching is "waiting for reply".

When the user checks the teammate matching result, the system would execute different functions according to the situation.

For case1, if the user hasn't submitted his or her information, then he or she won't see any result.

If the user is already in a team and the team is incomplete, like case6, then the user would still need to wait for other users to join the team. If there isn't any team matched for the user, like case2 or case3, or there is team matched but the team is full, like case4, then a new team only with the user would be created. If there is already an incomplete team matched, then the user would join the team, and if the team is still incomplete, like case5, then the user still needs to wait for other users to join them. In all these situations above, the user won't see the matching result.

If the user's team become complete, like case7, then the user will see the matching result.

5.6.4 Teammate System – Dealing Users' Reply

5.6.4.1 Inputs

Case1: Refuse the matching result

Case2: Accept the matching result and then another teammate refuses it

Case3: Accept the matching result and all the other teammates accept it

Case4: Accept or refuse the result when the status is “success”

5.6.4.2 Expected Outputs & Pass/Fail Criteria

Case1: The user can’t see the matching result anymore until a new team is formed.

Case2: After the user accepts the result, the user can still check the matching result and its status is still “waiting for reply” until another user refuses it. Then the user can’t see the matching result anymore until a team is formed.

Case3: After the user accepts the result, the user can still check the matching result. Its status is still ‘waiting for reply’ until all teammates accept it, then the status will be changed into “Success”.

Case4: The matching result and status don’t change.

If one of user refuse refuses the result, then the matching result will be removed, and the matching would restart. Else, the status of the matching would be changed into “success” and then the matching can’t be changed anymore.

6 LESSONS LEARNED

Compared to the initial design, the actual website we developed has quite some differences. Some are improvement and highlights of the project compared to the initial design, others are minor drawbacks due to all kinds of reason.

Some details that were not included in the initial design are considered. For example, we designed the CUHK student verification process by sending verification email to CUHK email address. In the forum, one may be able to leave comments and set his or her own posts to be public or private. Multiple persons could chat together, and chat messages are only visible when both sides are online.

However, there are also some initial considerations we failed to realize. The functions of editing and deleting information are missing. Tags and rank list for matching are also abandoned. Helping students to avoid freeriders was considered as the main function at first but didn't reach our expectation at last. Further improvements like administrator and machine learning are not realized due to the limited timeline.

At the beginning of designing functions, some problems were considered while some didn't. When developing the website, the difficulties of different problems varied a lot. Therefore, the difference between the initial design and the final result appeared.

What we have learned includes but are not limited to the following three points.

6.1 Our Advantages

During the development of our project, we follow the principle of software development and we are also concerned about the qualities of the project.

One advantage of our project is we try to achieve modularity. Since different parts of work are divided, one can focus on developing his or her own concerns and therefore there's little repeated work. When discussing the project together, we only need to learn about the abstractions of other teammates' work and instead of trying to comprehend everything, which could be very efficient.

According to the functionalities, we modularize the code into different parts, i.e. routers, plugins and models. Some modular is reusable, etc. In this sense, the logic among the code becomes clear and debugging becomes efficient, which improve not only the maintainability but also the reliability.

Moreover, since the modular of each member is related to both front-end and back-end, everyone has an opportunity to learn about two kinds of knowledge and have experience in full stack development.

Another advantage is we focus on reliability. We try to consider all possible situations for our software. For security, we design some encryption mechanism to prevent maliciously hacking and apply email authentication to verify the identities of the user. We also paid a lot of attention to testing our codes in order to reduce bugs. Although bugs are difficult to avoid totally, we still try our best to reduce them. Till now, our software won't be troubled by bugs in most situations, and it will carry on in an acceptable way even when it faces some bugs.

6.2 Our Disadvantages

Version control is the biggest problem when developing the website. Since everyone is working on one part, when merging the changed code, one may fail to update other people's code at the same time. Therefore, when testing, some modifications may lose during version updating. Most of the time one of the groupmates made changes on several files and zipped the whole file together. Then other four groupmates need to download the whole file and do the unzipping. This happens two to three times a day and is quite time-consuming.

Timeline is another aspect we didn't manage properly in this project. We spent quite an amount of effort in discussing some theoretical problems and dividing the work at first. Later we didn't have enough time to modify and improve our code and therefore the final result didn't fully reach our expectation. Some further exploration like administrator and machine learning were not able to realize also because of this.

There are also other aspects we need to improve:
For the client-side, some parts of our website used other templates and were not easy to merge two templates together in the end. Therefore, there was some inconsistency in the user interface.

For the server-side, some details could be handled in a more efficient way. In the teammate matching system, the size of the team must be between 2 and 6, because the function to handle the matching request is separated in some way according to a different size. If the size if large, the function will be more complicated. In this sense, if the relative functions could be more general, the size of a team could be larger.

6.4 Future plan

If we redo the project, we will try to make more detailed time arrangement and push our progress faster. Also, we will surely need to find a better way for version control and improve our matching mechanism by implementing more advanced algorithms.

If we need to develop another website, we will pay more attention to the connection between client-side and server-side. The consistency of the user interface should be addressed as it affects users' experience. And we may try to use another database like MongoDB and another tool like JAVA Web to make some of our functionality easier to implement.

7 CONCLUSION

CUPar is the first software project our group developed. At the beginning of this semester, we only had little experience in software design. Through one semester's learning and practice, we successfully built a platform to let CUHK students find suitable partners of all kinds. Till the end of the project, the platform may look simple, but it has already realized a lot of functions.

During the whole semester, we learned a lot in class and tried to use them in our project. In the same time, the experience when developing the project greatly helped us on comprehending the notes in lecture notes. What's more, the project has let us know the importance of team cooperation. All the things we gained through the process will surely benefit us a lot in a future study.

The project doesn't end here, we will keep improving our project in the following summer vacation until we complete all expected functions. We do believe that we can learn more from further development.

8 APPENDIX

The terminology and abbreviation are listed as follows (in alphabetic order).

Name	Description
Auth email	In this report, auth email is the email including CODE sent to the user's link-email account, which is used for verifying the identity of the user.
CODE	The code or auth code, in this report, CODE is the authentication code sent to the user's link-email addresses with the authentication email, which is used for verifying the identities of the ownership.
passwordRP	In this report, passwordRP is the password the users enter again for confirming their new passwords.
passport	In this report, passport is one of the attributions of cookie, it contains id, sid and username of the user. The cookie will be expired after 1 hour if the user has no operation during this time.
Signup/register	In this report, signup or register means creating the accounts.
User(s)	In this report, users are the CUHK undergraduate students who have registered their accounts.
Visitor(s)	In this report, visitors are the CUHK undergraduate students who have not created their accounts.