

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

SC2006 SOFTWARE ENGINEERING

SCSE Buddy
SSP1 Group 2 (Team JKT)

Team Leader:

Cholakov Kristiyan Kamenov (U2123543B)

Team Members:

Ang Jia Ze (U2121818C)

Mo Jiayun (U2123962F)

Theodore Tan Wan Li (U2122077E)

Julian Wong Wei Sheng (U2123563J)

Gupta Tushar Sandeep (U2123144E)

Table of Contents

Table of Contents	2
Revision History	3
1. Introduction	4
1.1 Purpose	4
1.2 Product Scope	4
1.3 User and Stakeholders	4
1.4 Assumptions	4
1.5 Constraints	4
1.6 Initial UI Mockup	4
2. Functional Requirements	10
2.1 Functional Requirements	10
2.2 Use Case Diagram	14
2.3 Use Case Descriptions	15
2.4 Class Diagram	40
2.5 Sequence Diagram	41
2.6 Dialog Map	47
3. Non-functional Requirements	48
3.1 Usability Requirements	48
3.2 Performance Requirements	48
3.3 Security Requirements	48
4. Interface Requirements	49
4.1 Users	49
4.2 Software	49
4.3 Communication Interfaces	49
5. Architecture Design	50
5.1 System Architecture Diagram	50
5.2 Design Pattern	50
6. Data Dictionary	52
7. Testing	54
7.1 Black Box Testing	54
7.2 White Box Testing	65
Appendix A: Video Demonstration	68

Revision History

Name	Date	Reason For Changes	Version
Ang Jia Ze	6 oct 2022	Creation of documentation	1.0.0
Ang Jia Ze	7 oct 2022	Included introduction	1.0.1
Ang Jia Ze	11 oct 2022	Added initial UI mockup	1.0.2
Ang Jia Ze	26 oct 2022	Added functional requirement	1.0.3
Ang Jia Ze	28 oct 2022	Added diagrams to functional requirements	1.0.4
Ang Jia Ze	28 oct 2022	Added architecture design	1.0.5
Ang Jia Ze	28 oct 2022	Added non functional requirement	1.0.6
Ang Jia Ze	28 oct 2022	Added interface requirement	1.0.7
Mo Jiayun	29 oct 2022	Correction of use case descriptions	1.0.8
Ang Jia Ze	29 oct 2022	Correction of use case descriptions	1.0.9
Ang Jia Ze	30 oct 2022	Added data dictionary	1.1.0
Ang Jia Ze	31 Oct 2022	Added sequence diagram	1.1.1
Ang Jia Ze	1 Nov 2022	Added black box testing	1.1.2
Ang Jia Ze	2 Nov 2022	Added white box testing	1.1.3
Mo Jiayun	8 Nov 2022	Added design considerations and edited the SRS	1.1.4
Ang Jia Ze	8 Nov 2022	Edited the diagrams	1.1.5
Mo Jiayun	10 Nov 2022	Updated design pattern	1.1.6

1. Introduction

1.1 Purpose

Our SCSEBuddy app aims to be the guide for SCSE students in NTU. Users can find their way around SCSE with the map function of the app.

Users can create an account to view the course review and forum portion of the app. A course review section is there for users to post or view other users' reviews of the courses provided by SCSE. Users can mingle in the included forum part of the application, from looking for friends to asking for advice. Interactions are facilitated by creations of post, ability to comment on post as well as liking post.

1.2 Product Scope

Our SCSEBuddy app will run on the Android operating system.

There are 3 main scope of the app:

1. The first scope is to provide a direction guide for SCSE. Users can key in their current location and the location in SCSE they wish to go to. Our app will then guide them with pictorial images of the path they should take to get to their destination.
2. The second scope is to help users make a better decision in planning their courses throughout their academic life with a course review. Users can leave reviews of courses they have taken, allowing other users to see the consensus of SCSE students with regards to the course. Users can then make an informed decision as they plan which courses to take for their next semester.
3. The third scope is to make connections between SCSE students with a forum. Users can create posts under related topics. Other users can like and comment on said posts to allow connections between students in SCSE.

1.3 Users and Stakeholders

Stakeholders of SCSEBuddy consist of SCSE faculty, SCSE students using Android phones and SCSEBuddy. SCSE building layout and course information will be retrieved from NTU website. SCSE students using Android phones can use SCSEBuddy to go around SCSE buildings with the map. They can use SCSEBuddy to post or read both course reviews and forums. SCSEBuddy seeks to ease SCSE students' life by creating a platform where information can be gained for students without connections to their seniors.

1.4 Assumptions

Users should have an Android device capable of running the application.
Users should have active internet access to use the application.

1.5 Constraints

SCSEBuddy only allows unregistered users to use the map function.
SCSEBuddy only supports Android operating systems that are running on Android 10.0 and up. (API level 29)
SCSEBuddy is currently only available in the English language.

1.6 Initial UI Mockup

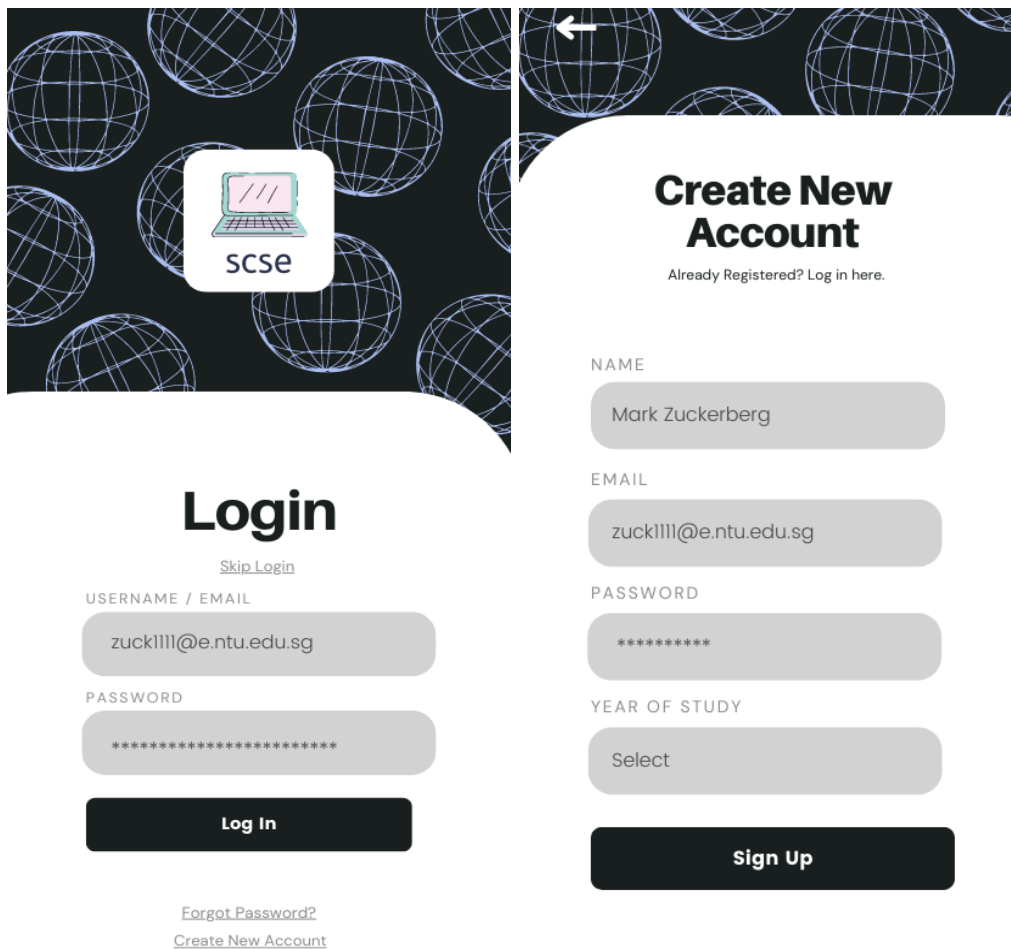


Figure 1: Login and account creation screen.

Figure 1 shows the user interface for login and account creation. The layout and design of the interface is kept minimalistic to ensure the user experience is intuitive. This will enable the users to create an account and log in smoothly.

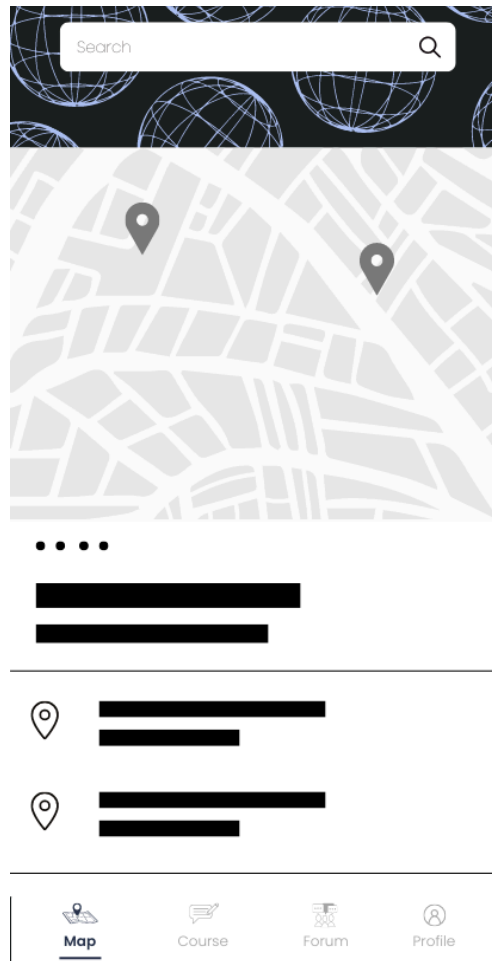


Figure 2: Maps navigation.

Figure 2 shows the home page after the user logs in or skips log in. The maps will be displayed to them. Users can view the map or key in their current and desired location in SCSE. Maps will then generate the best path for the user to reach their destination.

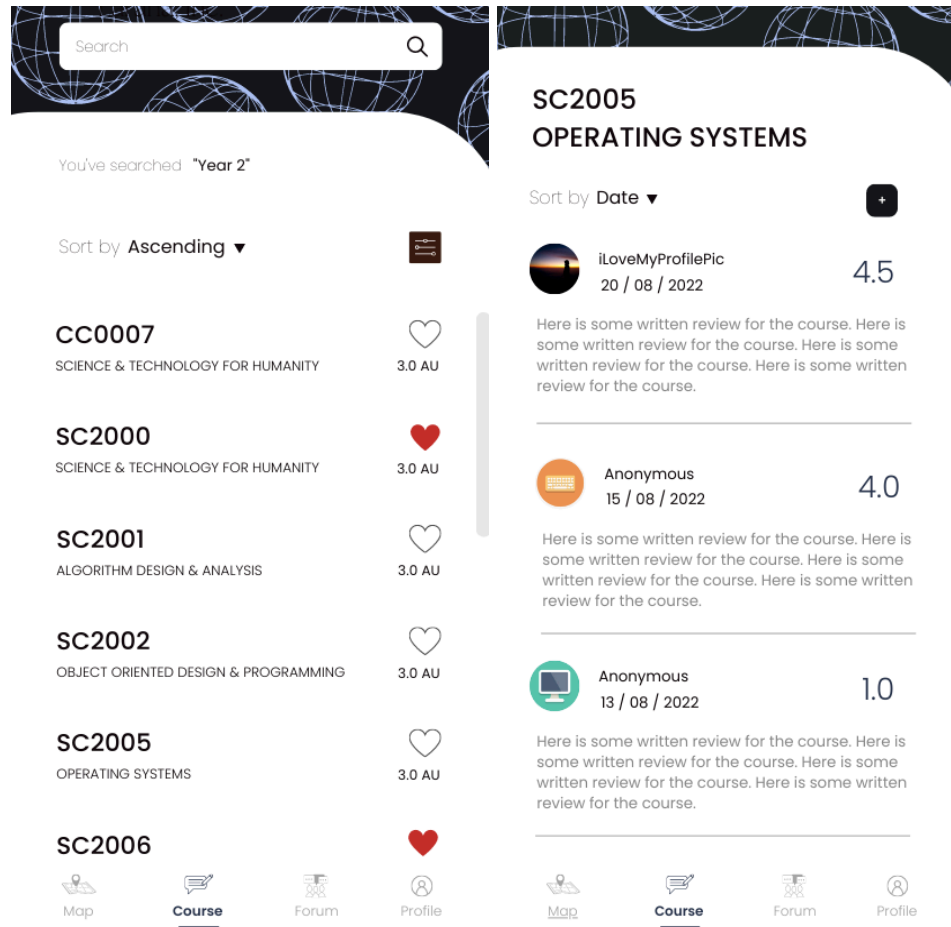


Figure 3: Course page and course reviews.

Figure 3 shows the course page and reviews left by users of SCSEBuddy.

In the course page, users can see the basic course information such as course number and the credits of the course. Users can favorite the courses as well.

If users press on the course in the course page, it shows the course review left by other users. Users can press on the '+' icon to add their own review of the course. Reviews can be left anonymously with a rating from 1 to 5 stars, accompanied by text description.

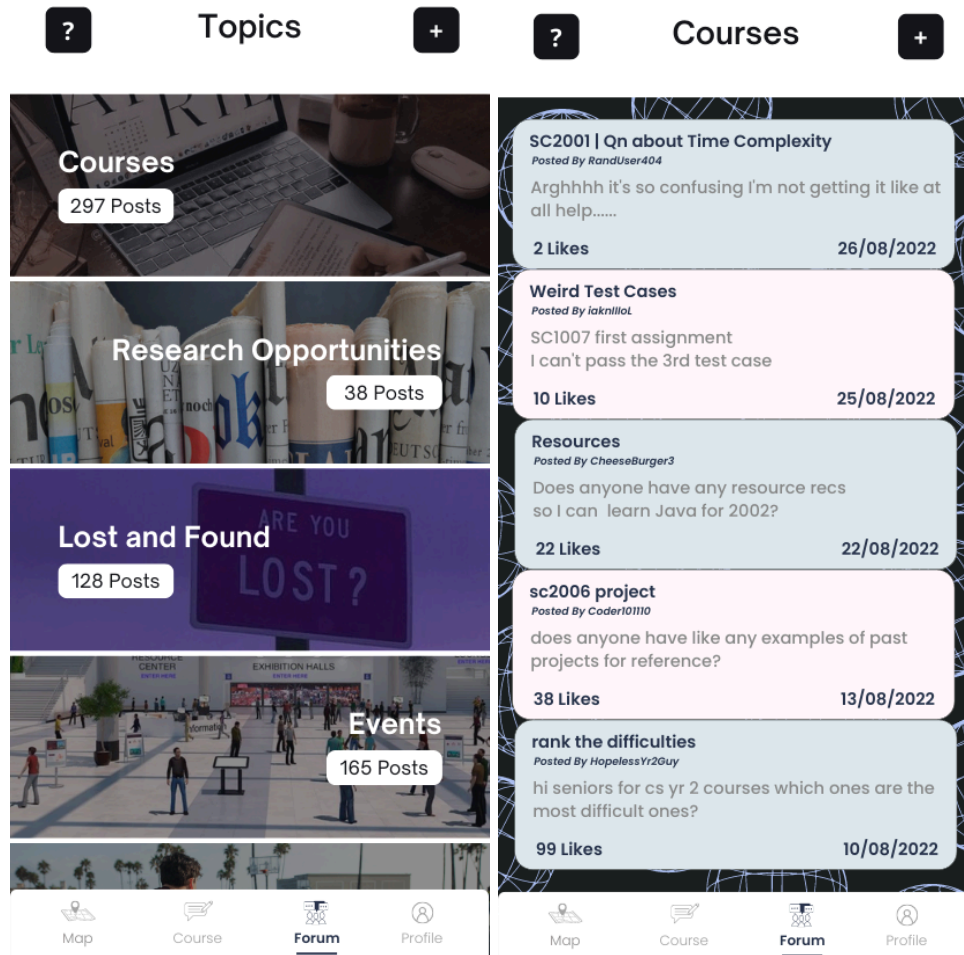


Figure 4: Forum and forum posts.

Figure 4 shows the forum and the posts in the forum topic.

For the main forum page, users can view any of the forum topics. Inside the forum topics, users can view, like and comment on the posts as well as create their own post. To create a post, users can press the '+' icon to open up the post creation page. Each post will contain the original poster's name, time of post, as well as number of likes.

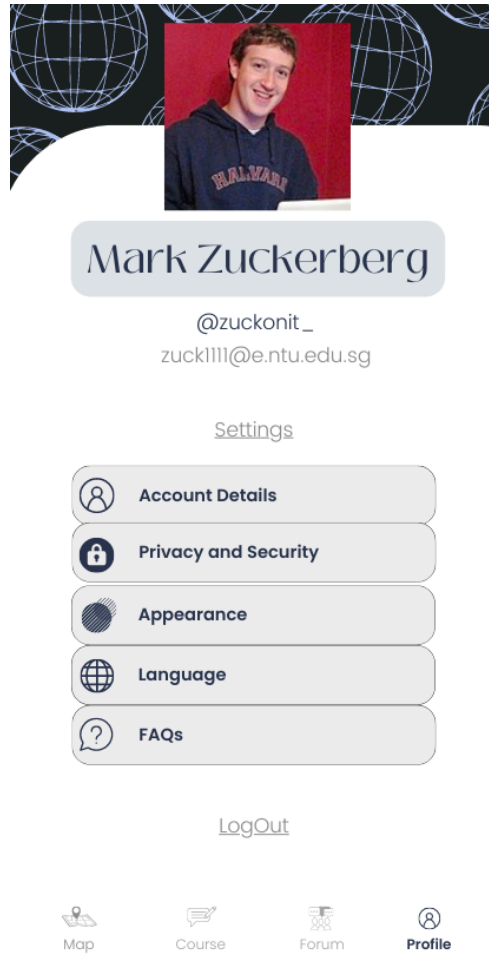


Figure 5: Profile page.

Figure 5 shows the profile page of the user. The profile page will show users' own name, profile photo, and email information.

In the FAQs, users can read the FAQ.

In account details, users can change their account details such as password, name and emails.

Users can choose to log out of their account as well.

2. Functional Requirements

2.1 Functional Requirements

1. Navigation/Map
 - 1.1. The navigation/map page must show the search function.
 - 1.2. The user shall be able to search for a particular room.
 - 1.2.1. When a searched room is found, the user shall be able to choose to navigate to it.
 - 1.2.1.1. The user shall be able to enter a start location.
 - 1.2.1.2. The navigation to room must appear as a combination of arrow pointing to the searched room
 - 1.2.1.2.1. If multiple floors are involved in the route, stair directions shall appear (e.g. "Go up/down to Xth floor").
 - 1.2.1.2.2. If multiple floors are involved in the route, arrow directions must be shown for every floor.
 - 1.2.1.2.2.1. Floor on which the only directions are to go up/down the stairs shall not be included.
 - 1.2.1.3. The user shall be able to see a preview of the route as a sequence of 360-degree photos.
 - 1.3. The user shall be able to input the room names.
 - 1.3.1. After inputting the room names, the app will be able to locate you.
 2. Course Reviews
 - 2.1. The app must have records of different SCSE courses.
 - 2.1.1. Every record must contain the course's name.
 - 2.1.2. Every record must contain the course's code.
 - 2.1.3. Every record must contain a set of reviews about it.
 - 2.2. The course review page must show all course records at first.
 - 2.2.1. The whole set of courses must be sorted in ascending order of the year in which the course is taken (based on SCSE curriculum structure.).
 - 2.2.2. The courses in the same year must be sorted in ascending order of their course codes (based on SCSE curriculum structure).
 - 2.3. Users shall be able to search among all of the available course records based on the course code.
 - 2.4. The search result must display all the courses containing the searched keyword sorted in ascending order of their course code.
 - 2.5. A sorting functionality for the search results shall be available to the user.
 - 2.5.1. The results shall be able to be sorted in ascending/descending order of their course code.
 - 2.5.2. The results shall be able to be sorted in ascending/descending order of their course name.
 - 2.6. The user shall be able to open the page of any course.
 - 2.6.1. The page must contain the reviews about the selected course.

- 2.6.1.1. The review record must include the course code.
- 2.6.1.2. The review record must include the date of publication of the review
- 2.6.1.3. The review record must include the name of the author, if any.
 - 2.6.1.3.1. If the review's author didn't include their name, the name shall appear as "Anonymous".
- 2.6.1.4. The review record must include text description of the review, if any.
- 2.6.1.5. The review record must include the numeric grade about the course.
 - 2.6.1.5.1. The numeric grade must be a number in the range of 1 to 5.
- 2.6.2. The page of the selected course must show all of the reviews for the selected course.
 - 2.6.2.1. The reviews must be sorted in an descending order of their publish date by default.
 - 2.6.2.2. The user shall be able to sort the reviews.
 - 2.6.2.2.1. The user shall be able to sort the reviews in ascending/descending order of the date the review is published.
 - 2.6.2.2.2. The user shall be able to sort the review in the ascending/descending order of the numeric grade given in the review .
- 2.6.3. The specific course page must allow the user to give a review.
 - 2.6.3.1. The user must enter a numeric grade for the course.
 - 2.6.3.2. The user shall be able to give text feedback about the course.
 - 2.6.3.3. The publish date and time of the review must be recorded.
 - 2.6.3.4. The user shall be able to select whether they want the review to be anonymous or not.

3. Forum

- 3.1. The application must contain a set of topics.
 - 3.1.1. Every topic record must contain the name of the topic.
 - 3.1.2. Every topic record must contain the set of all posts about the topic.
- 3.2. The forum page must show all the topics available.
 - 3.2.1. The topics must be sorted in descending order of the date that the last post about the topic was posted (default).
 - 3.2.2. The user should be able to sort the topics.
 - 3.2.2.1. The user shall be able to sort the topics in ascending/descending order of the number of posts in a topic.
 - 3.2.2.2. The user shall be able to sort the topics in ascending/descending order of their names.
 - 3.2.3. The user shall be able to search for a post by parts of its name.
 - 3.2.4. The result of the search must contain all the topics with names containing the search entry.
 - 3.2.5. The user shall be able to enter the individual page of every topic.
 - 3.2.5.1. The individual page of every topic must include all the posts in the topic.

- 3.2.5.1.1. Every post record must contain the date the post was published.
- 3.2.5.1.2. Every post record must contain the number of likes it has.
- 3.2.5.1.3. Every post record must contain a text description.
- 3.2.5.1.4. Every post record must contain the name of its author or display as anonymous.
- 3.2.5.1.5. Every post record should be able to contain a set of comments.
 - 3.2.5.1.5.1. Every post record must contain the name of its author or display as anonymous.
 - 3.2.5.1.5.2. Every comment record must contain a text description.
 - 3.2.5.1.5.3. Every comment record must contain the date of publishing it.
- 3.2.5.2. The user shall be able to like a post.
 - 3.2.5.2.1. The user shall be able to remove the like at any time.
- 3.2.5.3. The user shall be able to sort the posts.
 - 3.2.5.3.1. The user shall be able to sort the post in ascending/descending order of the number of likes.
 - 3.2.5.3.2. The user shall be able to sort the post in ascending/descending order of date published..
- 3.2.5.4. The user shall be able to publish a post.
 - 3.2.5.4.1. The user must enter the text description of the post.
 - 3.2.5.4.2. The name of the user must be recorded.
 - 3.2.5.4.3. The date of publishing must be recorded.

4. Sign In/Sign Up

- 4.1. Every user shall be able to create an account in the application
 - 4.1.1. The user account record must contain their name.
 - 4.1.2. The user account record must contain their email. (not visible to other users)
 - 4.1.3. The user account record must contain their password. (not visible to other users)
 - 4.1.4. The password used must match the re-enter password.
- 4.2. Every user must be able to login into their account.
 - 4.2.1. The user must enter an email and password to log in.
 - 4.2.2. The user shall be able to contribute to the application only if they have an account.
- 4.3. Every user must be able to reset their password by using the forget password.
 - 4.3.1. The user must key in the email used.
 - 4.3.2. The user must receive an email to reset the password.

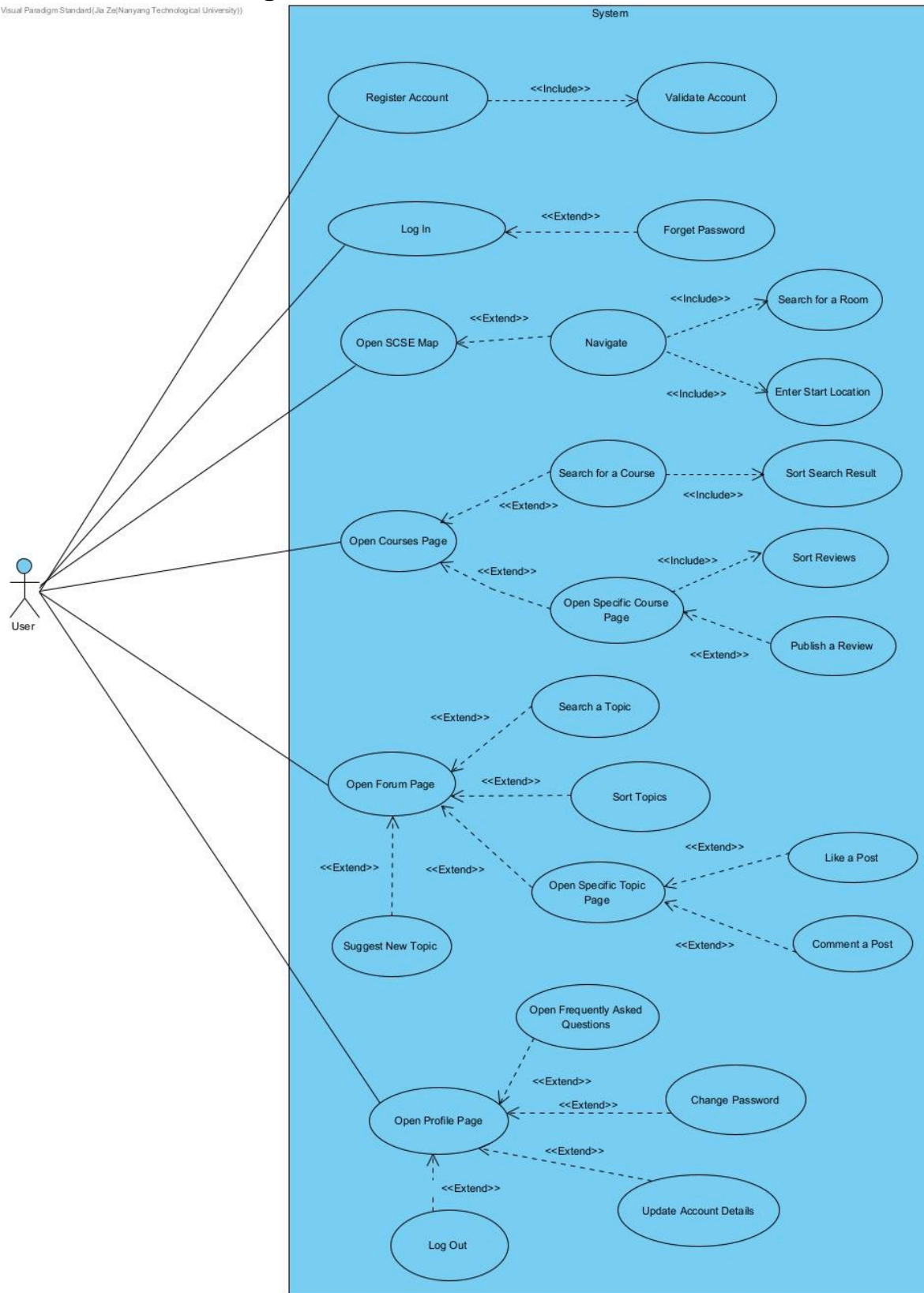
5. Profile

- 5.1. The user should be able to enter their profile page.
 - 5.1.1. The profile page must display their name.
 - 5.1.2. The profile page must display their email.

- 5.1.3. The profile page must display their profile photo.
- 5.1.4. The profile page must allow the user to update account details.
 - 5.1.4.1. The user must be allowed to update the name.
 - 5.1.4.2. The user must be allowed to update the email.
- 5.1.5. The profile page must allow users to change their password.
- 5.1.6. The profile page must allow users to read the frequently asked questions.
- 5.1.7. The profile page must allow users to log out.

2.2 Use Case Diagram

Visual Paradigm Standard (Jie Ze/Nanyang Technological University)



2.3 Use Case Descriptions

Use Case ID:	RA		
Use Case Name:	Register Account		
Created By:	Theodore	Last Updated By:	Ang Jia Ze
Date Created:	05/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	Registration for a new user account.
Preconditions:	<ol style="list-style-type: none"> 1. The user's account must not exist in the database. 2. There must be an internet connection.
Postconditions:	<ol style="list-style-type: none"> 1. A verification code must be sent to the user's email.
Priority:	High
Frequency of Use:	Once per lifetime.
Flow of Events:	<ol style="list-style-type: none"> 1. The user enters a valid email address, name, password and re-enter password. 2. The system validates the password and re-enter password fields are identical. 3. The system verifies the email format. 4. The System will send a 6 digit verification code to the registered email address. 5. The system adds the account into the database.
Alternative Flows:	-
Exceptions:	<p>RA.EX.1: System detects mismatch between 'password' field and 're-enter password' field.</p> <ol style="list-style-type: none"> 1. System display error message. 2. Return to step 1. <p>RA.EX.2: The user enters an invalid email address.</p> <ol style="list-style-type: none"> 1. System display error message. 2. Return to step 1. <p>RA.EX.3: No internet connection.</p> <ol style="list-style-type: none"> 1. The system displays an error message 2. The system waits for an internet connection.
Includes:	Validate Account (VA)
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	VA		
Use Case Name:	Validate Account		
Created By:	Theodore	Last Updated By:	Ang Jia Ze
Date Created:	05/09/22	Date Last Updated:	29/10/2022

Actor:	Users
Description:	System to validate the account availability and to verify the email using the verification code.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. Account must not already exist in the database.
Postconditions:	<ol style="list-style-type: none"> 1. The account's information is correct and in the correct format. 2. The user's account is added to the database.
Priority:	High
Frequency of Use:	Once per lifetime.
Flow of Events:	<ol style="list-style-type: none"> 1. The system checks the verification code. 2. System validates the account availability in the database. 3. System creates the account in the database.
Alternative Flows:	-
Exceptions:	<p>VA.EX.1: System detects that the user account already exists in the database.</p> <ol style="list-style-type: none"> 1. System displays an error message. 2. The user needs to enter a new username/email. 3. The user selects the register button to re-attempt registration. 4. Return to step 1. <p>VA.EX.1: The verification code is wrong.</p> <ol style="list-style-type: none"> 1. System will display an error message. 2. Give the user 2 more attempts to enter the code. 3. If all attempts are incorrect, return to step 1. <p>VA.EX.3: No internet connection.</p> <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	LI		
Use Case Name:	Log In		
Created By:	Kristiyan Cholakov	Last Updated By:	Mo Jiayun
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The system checks if the user is in the database and if their credentials are correct. If yes, the system grants them access to the functionality that requires an account.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must be registered to log in.
Postconditions:	<ol style="list-style-type: none"> 1. The user enters the application and accesses all the functionality that requires a previously registered account.
Priority:	High
Frequency of Use:	0-20 times per day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user enters their credentials. 2. The system checks if the user is registered. 3. The system lets the user in if they are registered.
Alternative Flows:	LI.AC.1: The user does not want to log in. <ol style="list-style-type: none"> 1. The user clicks on the 'Skip Login' button. 2. The user accesses only the map functionality of the application.
Exceptions:	LI.EX.1: The user's credentials are incorrect: <ol style="list-style-type: none"> 1. The system displays an error message to the user. 2. The user is returned to step 1. LI.EX.2: The user forgets the password. <ol style="list-style-type: none"> 1. The user clicks the 'Forgot Password' button. 2. The user inputs an email address. 3. The user clicks the 'Reset Password' button. 4. The system sends an email to the user with instructions. LI.EX.3: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	N		
Use Case Name:	Navigate		
Created By:	Kristiyan Cholakov	Last Updated By:	Mo Jiayun
Date Created:	05/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The system must be able to navigate the user when the user enters a start location and a destination.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The entered rooms must exist.
Postconditions:	<ol style="list-style-type: none"> 1. The system must show the user an overview of the shortest way to the destination. 2. The system must wait until the user confirms they want to start the navigation process. 3. The system must navigate the user via the shortest way. 4. The system must show a complete path of 360-degree photos to the final destination. 5. The system must mark the door of the final destination with a highlighted rectangle.
Priority:	High
Frequency of Use:	0-10 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user enters a start location. 2. The user searched for a destination location. 3. The system shows an overview of the shortest path. 4. The user confirms the path. 5. The system shows the path in 360-degree photos. 6. The system highlights the final destination's door.
Alternative Flows:	<p>N.AC.1: The user may only want to check the path to the destination, without navigating to it.</p> <ol style="list-style-type: none"> 1. Enter the start location and the final destination. 2. The user does not proceed to navigate (only looks at the path).
Exceptions:	<p>N.EX.1: No internet connection:</p> <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. <p>N.EX.2: Either one of the start location or the final destination is not existing/found:</p> <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system returns to step 1.
Includes:	Search for a Room (SR) & Enter Start Location (SL)
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	ESL		
Use Case Name:	Enter Start Location		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user enters a valid room number/name and confirms if this is the room after seeing the overview location. This room is the start location.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The entered room must exist.
Postconditions:	<ol style="list-style-type: none"> 1. The system must check if the room is found. 2. The system will display an overview of where the entered room is at. 3. The user will be able to verify if this is actually his starting location (by clicking on a "Are you here?" button).
Priority:	High
Frequency of Use:	0-10 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user enters the room name/number in the Search bar. 2. The system checks if the room exists. 3. The system shows an overview of the room location. 4. User verifies that they are next to this room. 5. The system waits for a destination room.
Alternative Flows:	-
Exceptions:	SL.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. SL.EX.2: The entered room is not found: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system returns to step 1.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	SR		
Use Case Name:	Search for a Room		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user searched a room by its name/number and if the room exists it is the final destination of the searched path. An overview of the path is presented to the user.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. A start location/room must be entered first. 3. The entered room must exist.
Postconditions:	<ol style="list-style-type: none"> 1. The system must check if the room exists. 2. Proceeds to Navigate with the correct start location and final destination.
Priority:	High
Frequency of Use:	0-20 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user enters the room name/number in the Search bar. 2. The system checks if the room exists. 3. The system proceeds to Navigate with the correct start location and final destination.
Alternative Flows:	-
Exceptions:	<p>SR.EX.1: No internet connection:</p> <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. <p>SR.EX.2: The entered room is not found:</p> <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system returns to step 1.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	OCP		
Use Case Name:	Open Courses Page		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to access the courses page and interact with the different courses information if they are logged in to their account.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The system must display all the user favorites at the top of the list in the ascending year of modules taken based on SCSE curriculum structure. 2. The system must then display all courses in the ascending year of modules taken based on SCSE curriculum structure. 3. The user must be able to open any specific course page.
Priority:	Medium
Frequency of Use:	0-20 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The system displays all the available courses. 2. The user can open any specific course pages.
Alternative Flows:	-
Exceptions:	OCP.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	SC		
Use Case Name:	Search a Course		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The system must allow the user to search for a course by its course code.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. Some courses with the searched information must exist.
Postconditions:	<ol style="list-style-type: none"> 1. The system must display all the courses containing the searched information. 2. The user should be able to open any specific course page from the results of the search. 3. The user should be able to sort the results from the search.
Priority:	Medium
Frequency of Use:	0-20 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user enters the information that they want the course to contain. 2. The system displays all the courses that contain the information. 3. The user can open any specific course page from the results of the search.
Alternative Flows:	-
Exceptions:	SC.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. SC.EX.2: No courses with the searched information: <ol style="list-style-type: none"> 1. The system displays a message to the user. 2. The system returns to step 1.
Includes:	Sort Search Result (SSR)
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	SSR		
Use Case Name:	Sort Search Result		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The system must allow the user to sort the result from the search of courses.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. There must be a result from the search.
Postconditions:	<ol style="list-style-type: none"> 1. The system must display the results from the search in a sorted order (ascending/descending) based on (year, code, name) according to the user's choice.
Priority:	Low
Frequency of Use:	1 time a day
Flow of Events:	<ol style="list-style-type: none"> 1. The user selects the way they want the results to be sorted 2. The system sorts the results in the chosen way.
Alternative Flows:	SSR.AC.1: Default sorting: <ol style="list-style-type: none"> 1. The system sorts the results in ascending order based on their course codes by default.
Exceptions:	SSR.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. SSR.EX.2: No results from the search: <ol style="list-style-type: none"> 1. The system displays a message to the user. 2. The system returns to step 1.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	OCP		
Use Case Name:	Open Specific Course Page		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user must be able to open the specific page to view the reviews of any selected course.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. A specific course must be selected.
Postconditions:	<ol style="list-style-type: none"> 1. The system must display all the reviews about the selected course.
Priority:	Medium
Frequency of Use:	0-20 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user selects a course. 2. The system displays all the reviews about the selected course.
Alternative Flows:	-
Exceptions:	OCP.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	Sort Reviews(RS)
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	RS		
Use Case Name:	Sort Reviews		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The system must allow the user to sort the reviews about a certain course.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. A specific course must be selected.
Postconditions:	<ol style="list-style-type: none"> 1. The system displays the reviews in the sorted order based on the default option or the user's choice (ascending, descending) (date published, numeric review grade).
Priority:	Low
Frequency of Use:	1 time a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user chooses how to sort the reviews (ascending, descending) (date published, numeric review grade) 2. The system displays the reviews in the sorted order according to the user's choice.
Alternative Flows:	RS.AC.1: Default sorting: <ol style="list-style-type: none"> 1. The system sorts the reviews in descending order based on the date they are published by default.
Exceptions:	RS.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	PR		
Use Case Name:	Publish a Review		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The users must be able to publish reviews about a certain course.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. A specific course must be selected. 4. The user should enter: <ol style="list-style-type: none"> a. Numeric review grade. b. Text feedback (optional). c. Date (auto-generated). d. Author's Name (optional).
Postconditions:	<ol style="list-style-type: none"> 1. The review record must be added in the database. 2. The review must be visible to the other users.
Priority:	Medium
Frequency of Use:	0-10 times per semester .
Flow of Events:	<ol style="list-style-type: none"> 1. The user selects a specific course. 2. The user enters the required data. 3. The user publishes the review. 4. The review is visible to the other users under the specific course page.
Alternative Flows:	-
Exceptions:	PR.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. PR.EX.1: Some required data is missing: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	AF		
Use Case Name:	Add Favorite		
Created By:	Mo Jiayun	Last Updated By:	Ang Jia Ze
Date Created:	23/10/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The users must be able to label certain courses as 'favorite'.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. The Courses page must be opened.
Postconditions:	<ol style="list-style-type: none"> 1. The system displays the user's favorite courses at the top of the list.
Priority:	Medium.
Frequency of Use:	0-5 times per semester.
Flow of Events:	<ol style="list-style-type: none"> 1. The user labels a specific course as favorite by clicking the star icon. 2. The system displays the user's favorite courses at the top of the list.
Alternative Flows:	AF.AF.1: unfavorite <ol style="list-style-type: none"> 1. The user clicks the star icons for courses that are already in the user's favorite course.. 2. The system removes the course from the user's favorite courses.
Exceptions:	AF.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	OF		
Use Case Name:	Open Forum Page		
Created By:	Kristiyan Cholakov	Last Updated By:	Ang Jia Ze
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to access the forum page and interact with the different topics' information if they are logged in to their accounts.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The system must show the user all topics in descending order, based on the posted date of the latest post in the topic. 2. The user must be able to enter any topic page.
Priority:	Medium
Frequency of Use:	0-10 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The system displays all the available topics. 2. The user can select to view any forum topics.
Alternative Flows:	-
Exceptions:	OF.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	PS		
Use Case Name:	Search for Post		
Created By:	Kristiyan Cholakov	Last Updated By:	Mo Jiayun
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to search for a post by its content or parts of it.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. The searched topic must exist.
Postconditions:	<ol style="list-style-type: none"> 1. The system displays posts containing the searched information to the user.
Priority:	Medium
Frequency of Use:	0-10 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user can search for a post by parts of its content. 2. The system displays posts containing the searched information to the user.
Alternative Flows:	-
Exceptions:	PS.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. PS.EX.2: No topic with the searched data in its name exists: <ol style="list-style-type: none"> 1. The system displays a message. 2. The system returns to step 1.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	SRTT		
Use Case Name:	Sort Topics		
Created By:	Kristiyan Cholakov	Last Updated By:	Mo Jiayun
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to sort all of the topics in order (ascending, descending) based on their name or number of posts in them.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The system must display the topics in the chosen manner.
Priority:	Low
Frequency of Use:	0-5 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user chooses the order (ascending, descending) and sorting criteria (number of posts, name) for the topics. 2. The topics are sorted in the chosen manner.
Alternative Flows:	-
Exceptions:	SRTT.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	OTP		
Use Case Name:	Open Specific Topic Page		
Created By:	Kristiyan Cholakov	Last Updated By:	Mo Jiayun
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The system must show all posts about the selected topic to the user.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. The user must select a specific topic.
Postconditions:	<ol style="list-style-type: none"> 1. The system must show all the posts within the selected topic.
Priority:	Medium
Frequency of Use:	10 times a day
Flow of Events:	<ol style="list-style-type: none"> 1. The user selects a topic. 2. The system displays all posts within the selected topic.
Alternative Flows:	-
Exceptions:	OTP.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	LP		
Use Case Name:	Like a Post		
Created By:	Kristiyan Cholakov	Last Updated By:	Mo Jiayun
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to like any post from a specific topic.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. A specific topic must be selected.
Postconditions:	<ol style="list-style-type: none"> 1. The system must allow the user to like any of the posts from a specific topic. 2. The likes of the post should be updated in the database. 3. The likes count should be updated for other users.
Priority:	Medium
Frequency of Use:	10 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user selects a topic. 2. The user goes through the posts . 3. They should be able to like any post.
Alternative Flows:	-
Exceptions:	LP.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	PC		
Use Case Name:	Comments on a Post		
Created By:	Kristiyan Cholakov	Last Updated By:	Mo Jiayun
Date Created:	06/09/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to comment on any post from a specific topic.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. A specific topic must be selected. 4. The comment must consist of the required data: <ol style="list-style-type: none"> a. Author's Name (user can select to be anonymous). b. Text Content. c. Date (auto-generated).
Postconditions:	<ol style="list-style-type: none"> 1. The system must allow the user to comment on any of the posts. 2. The comment record should be added to the database. 3. The comment should be visible to other users. 4. The system will display the latest comments on top of the discussion by default.
Priority:	Medium
Frequency of Use:	10 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user has selected a specific topic and a certain post in it. 2. The user enters all the required data for a comment record. 3. The user publishes the comment. 4. The comment is visible to other users.
Alternative Flows:	-
Exceptions:	PC.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. PC.EX.2: Not all the required data for a comment record is entered: <ol style="list-style-type: none"> 1. The system displays a message. 2. The system returns to step 1.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	UP		
Use Case Name:	Upload Post		
Created By:	Ang Jia Ze	Last Updated By:	Mo Jiayun
Date Created:	13/10/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to upload a post on any topic.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first. 3. A specific topic must be selected. 1. The comment must consist of the required data: <ol style="list-style-type: none"> a. Author's Name (user can select to be anonymous). b. Post Content. c. Date (auto-generated).
Postconditions:	<ol style="list-style-type: none"> 1. The system must allow the user to upload a post on any topic. 2. The post record should be inserted in the database. 3. The post should be visible to other users. 4. The system will display the latest post on top of the topic by default.
Priority:	Medium
Frequency of Use:	0-5 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user has selected a specific topic. 2. The user enters all the required data for a post record. 3. The user published the post. 4. The post is visible to other users.
Alternative Flows:	-
Exceptions:	UP.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. UP.EX.2: Not all the required data for a comment record is entered: <ol style="list-style-type: none"> 1. The system shows a message. 2. The system returns to step 1.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	OP		
Use Case Name:	Open Profile Page		
Created By:	Mo Jiayun	Last Updated By:	Ang Jia Ze
Date Created:	24/10/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to access the profile page if they are logged in to their accounts.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The system must display the profile page of the user, including profile picture, names, email, and options to log out, change account details, change password, or access the frequently asked questions.
Priority:	Medium
Frequency of Use:	0-5 times a day.
Flow of Events:	<ol style="list-style-type: none"> 1. The user opens the profile page. 2. The system displays the profile page of the user.
Alternative Flows:	-
Exceptions:	OP.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	CP		
Use Case Name:	Change Password		
Created By:	Mo Jiayun	Last Updated By:	Ang Jia Ze
Date Created:	24/10/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to change the password of their account if they are logged in to their accounts.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The system must update the new password into the database.
Priority:	Medium
Frequency of Use:	0-5 times per lifetime.
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the 'Change Password' button. 2. The user inputs an old password. 3. The user inputs a new password. 4. The system updates the new password into the database.
Alternative Flows:	-
Exceptions:	<p>CP.EX.1: No internet connection:</p> <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection. <p>CP.EX.2: The new password is the same as the old one:</p> <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system returns to step 1. <p>CP.EX.2: The old password is wrong:</p> <ol style="list-style-type: none"> 3. The system displays an error message. 4. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	UAD		
Use Case Name:	Update Account Details		
Created By:	Mo Jiayun	Last Updated By:	Ang Jia Ze
Date Created:	24/10/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to update their account details if they are logged in to their accounts.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The new account details must be updated into the database.
Priority:	Medium
Frequency of Use:	0-5 times per lifetime.
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the 'Update Account Details' button. 2. The user inputs new account details. 3. The system updates the new account details into the database.
Alternative Flows:	-
Exceptions:	UAD.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	OFAQ		
Use Case Name:	Open Frequently Asked Questions		
Created By:	Mo Jiayun	Last Updated By:	Ang Jia Ze
Date Created:	24/10/2022	Date Last Updated:	29/10/2022

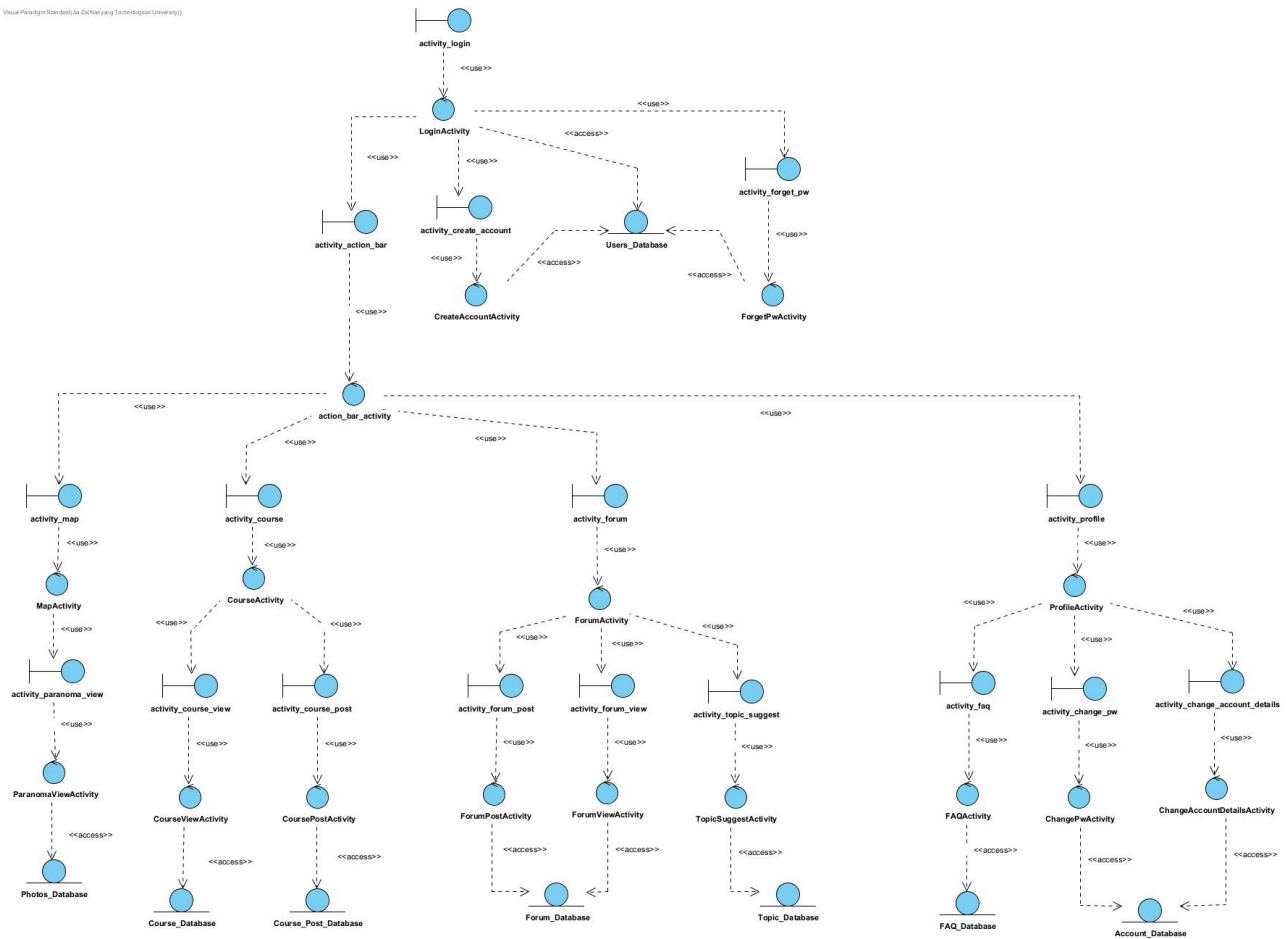
Actor:	Users
Description:	The user should be able to access the frequently asked questions page if they are logged in to their accounts.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The system must display a list of frequently asked questions and their corresponding answers.
Priority:	Low
Frequency of Use:	0-3 times a month.
Flow of Events:	<ol style="list-style-type: none"> 1. The user clicks the 'FAQs' button. 2. The system displays all the frequently asked questions and their corresponding answers.
Alternative Flows:	-
Exceptions:	OFAQ.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	LO		
Use Case Name:	Log Out		
Created By:	Mo Jiayun	Last Updated By:	Ang Jia Ze
Date Created:	24/10/2022	Date Last Updated:	29/10/2022

Actor:	Users
Description:	The user should be able to access the profile page if they are logged in to their accounts.
Preconditions:	<ol style="list-style-type: none"> 1. There must be an internet connection. 2. The user must log in into their account first.
Postconditions:	<ol style="list-style-type: none"> 1. The user must be logged out of their account.
Priority:	High
Frequency of Use:	0-3 times a month.
Flow of Events:	<ol style="list-style-type: none"> 1. The user selects the 'Log Out' option. 2. The system logs out the user from their account.
Alternative Flows:	-
Exceptions:	LO.EX.1: No internet connection: <ol style="list-style-type: none"> 1. The system displays an error message. 2. The system waits for an internet connection.
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

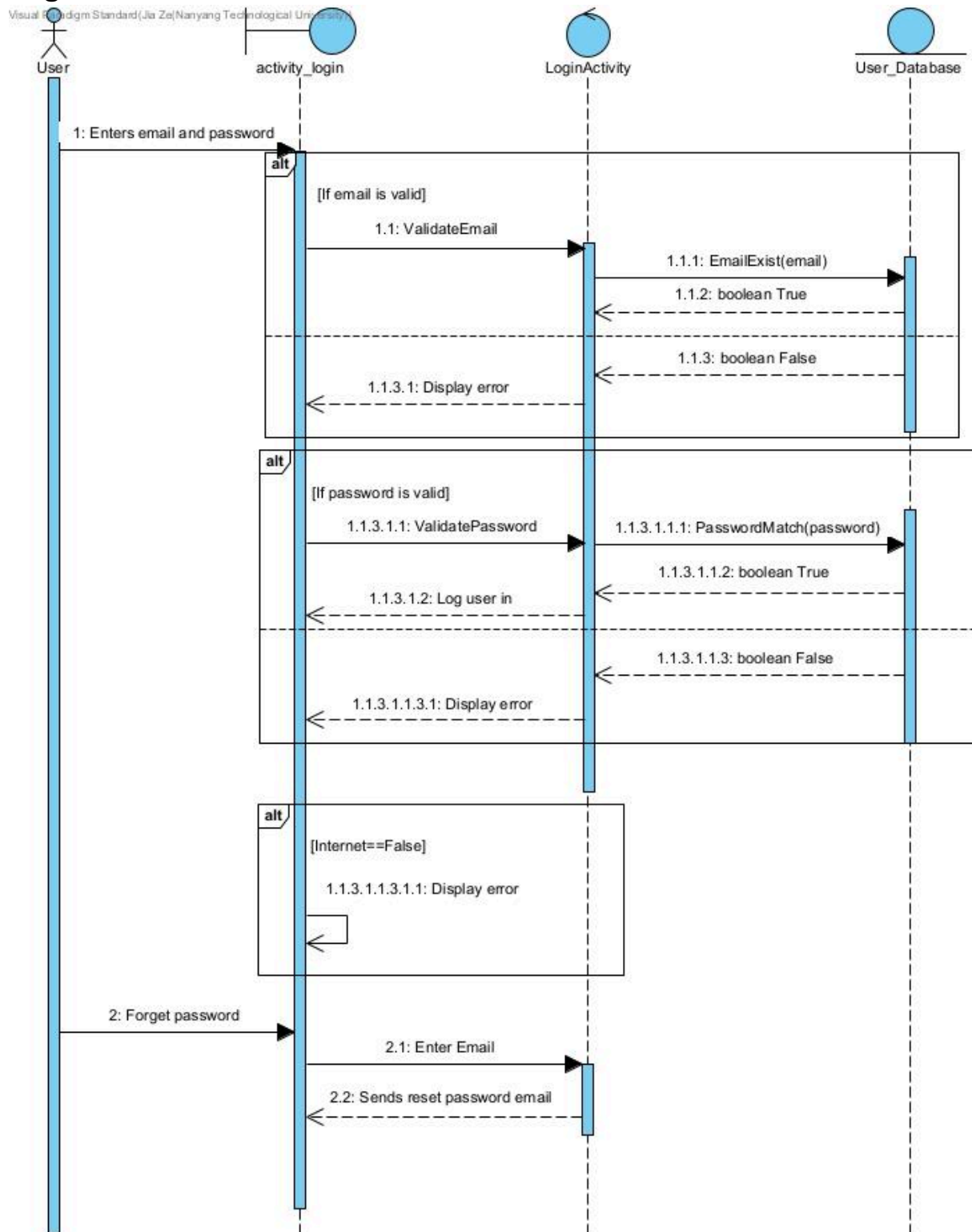
2.4 Class Diagram

Visual Paradigm Standard (Jia De Nan Jiang Technological University)



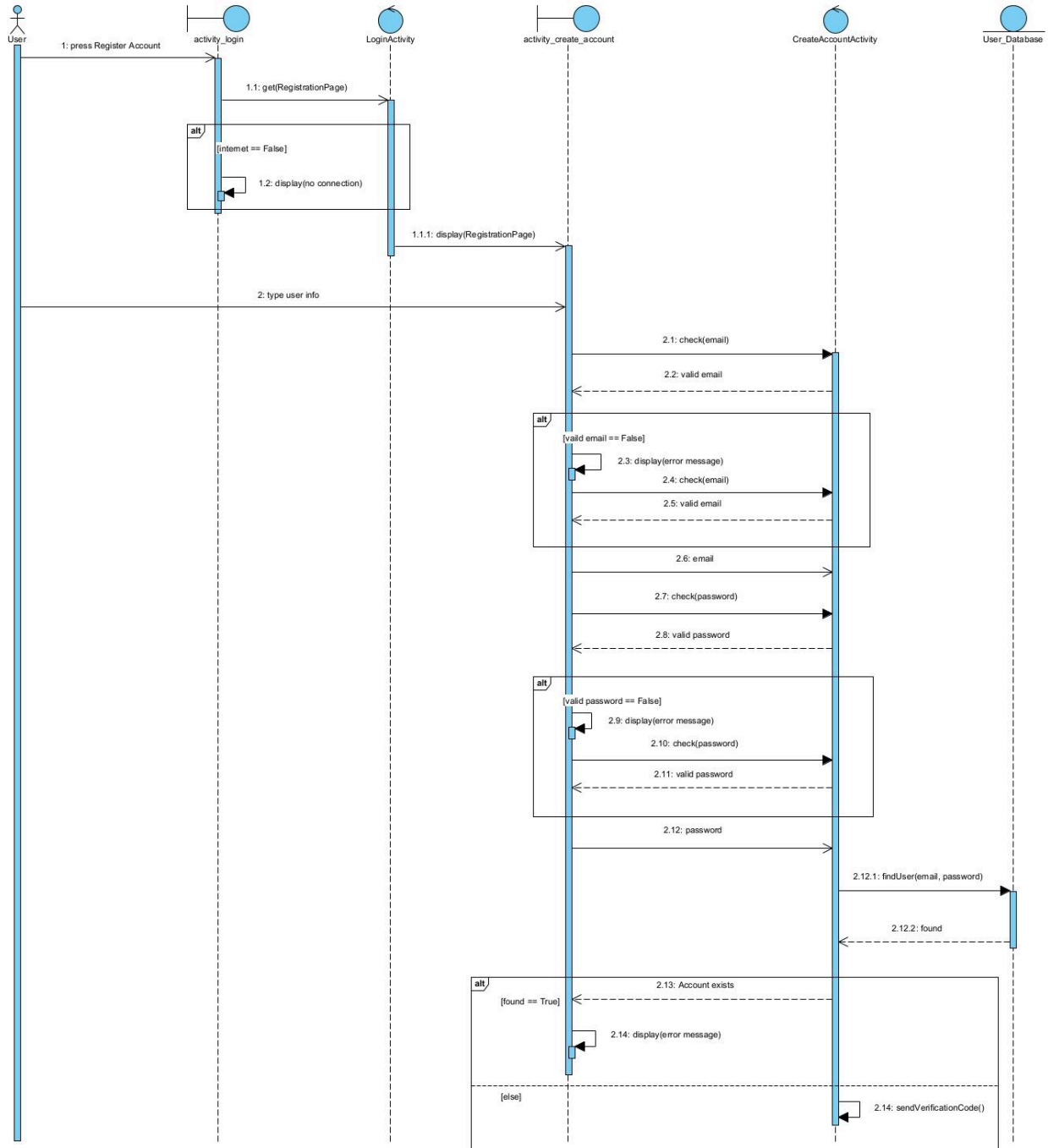
2.5 Sequence Diagram

Login

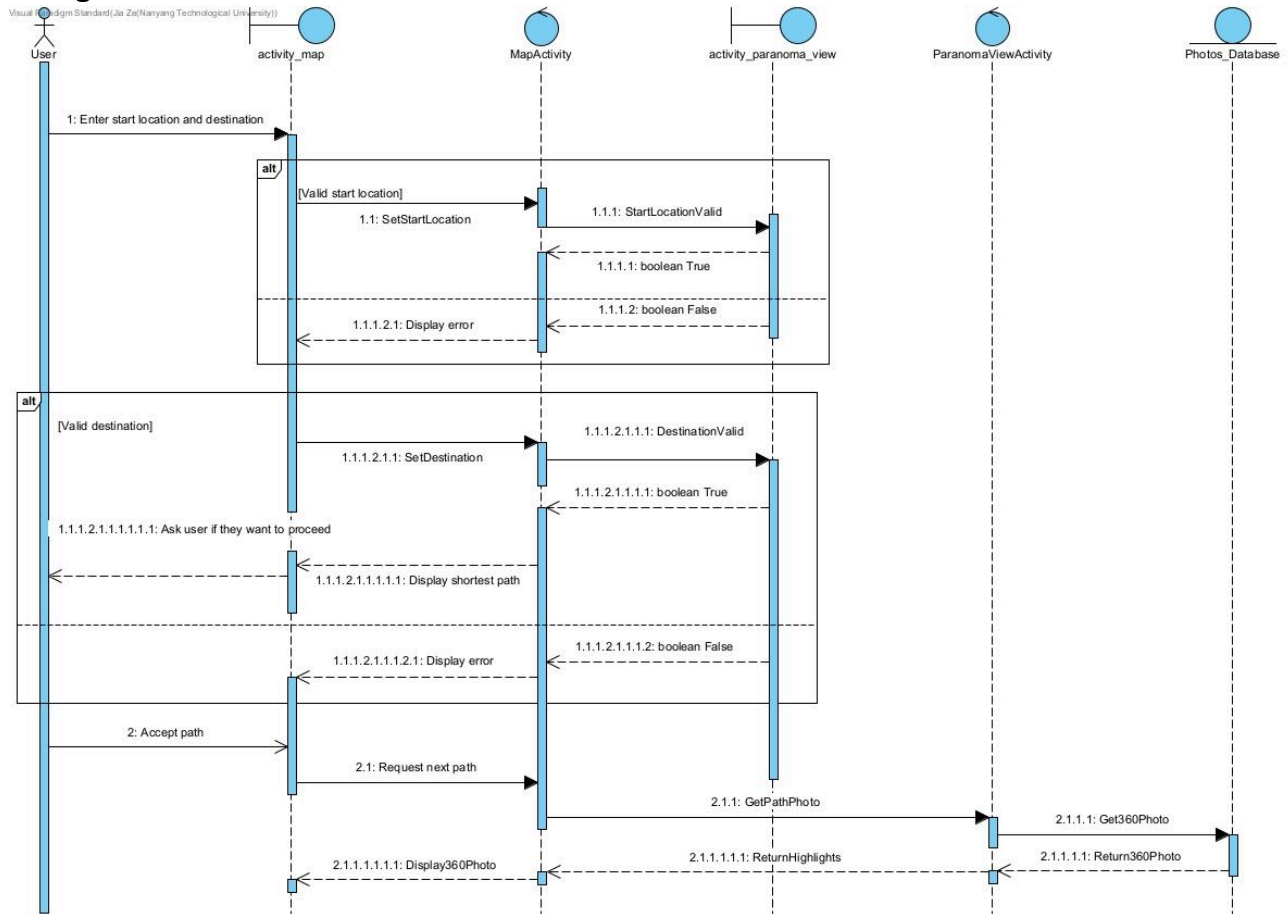


Register Account

© 2019 Peking University (Beijing)

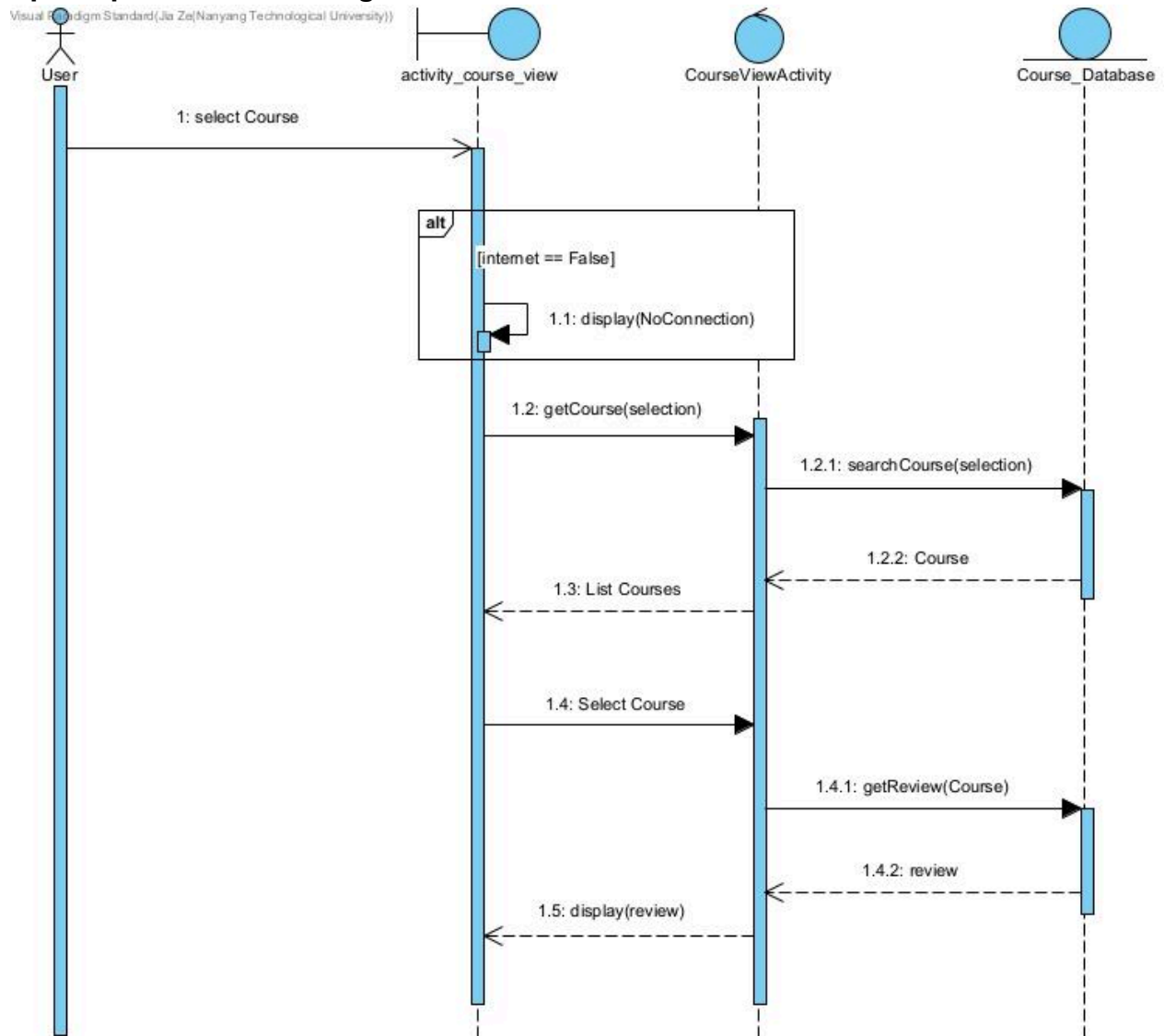


Navigation



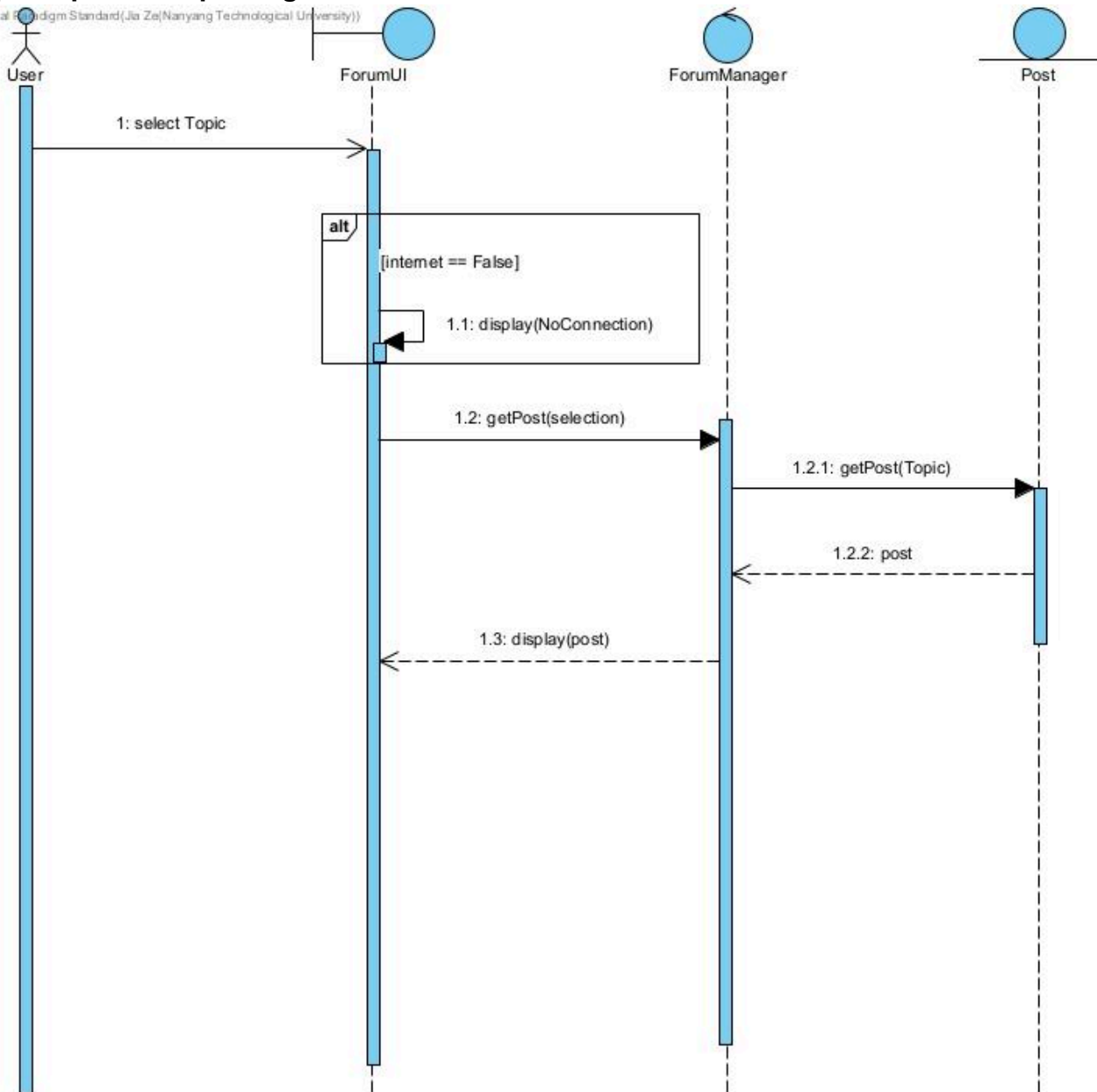
Open Specific Course Page

Visual Modeling Standard (Jia Ze(Nanyang Technological University))

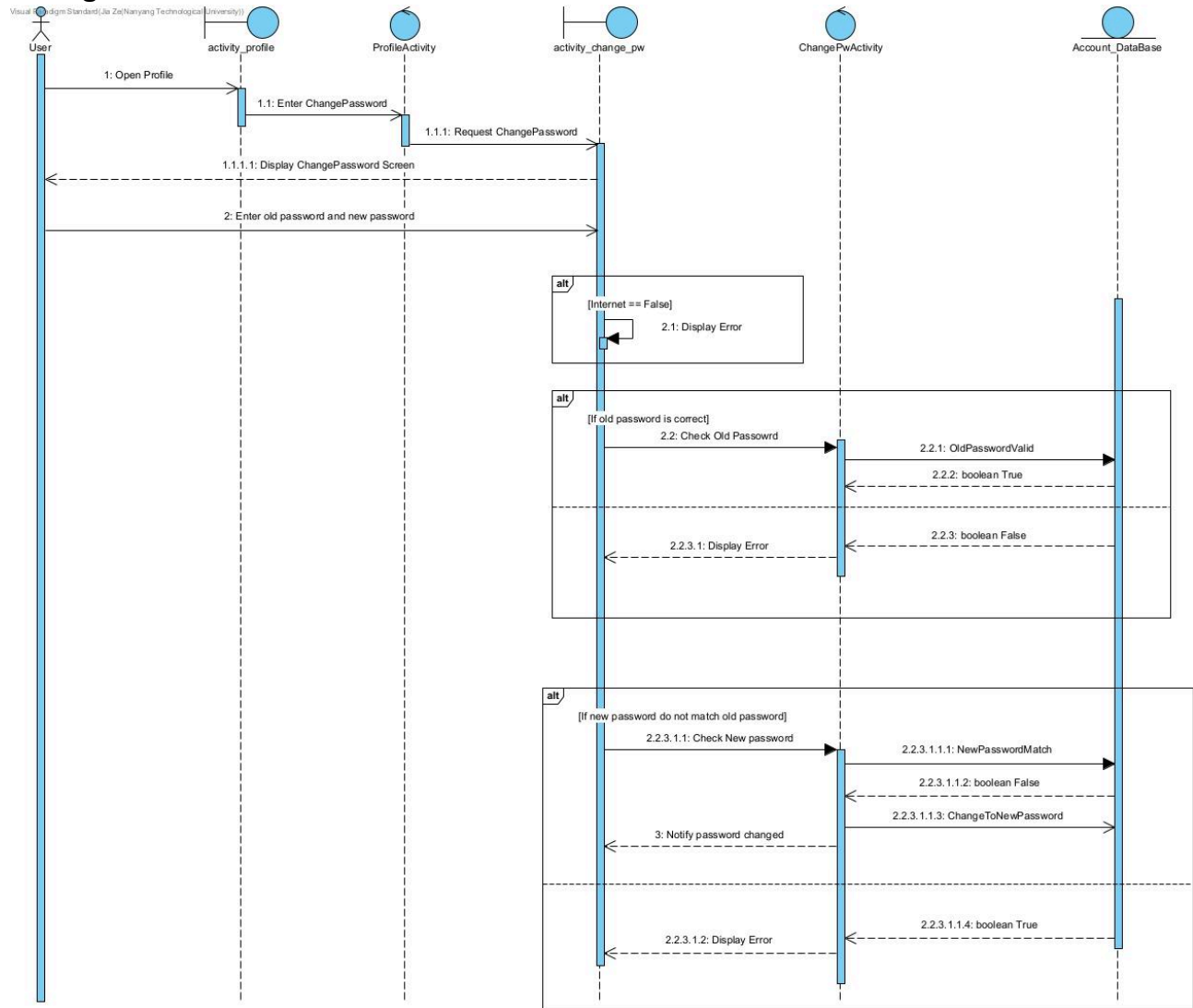


Open Specific Topic Page

Visual Paradigm Standard (Jia Ze (Nanyang Technological University))

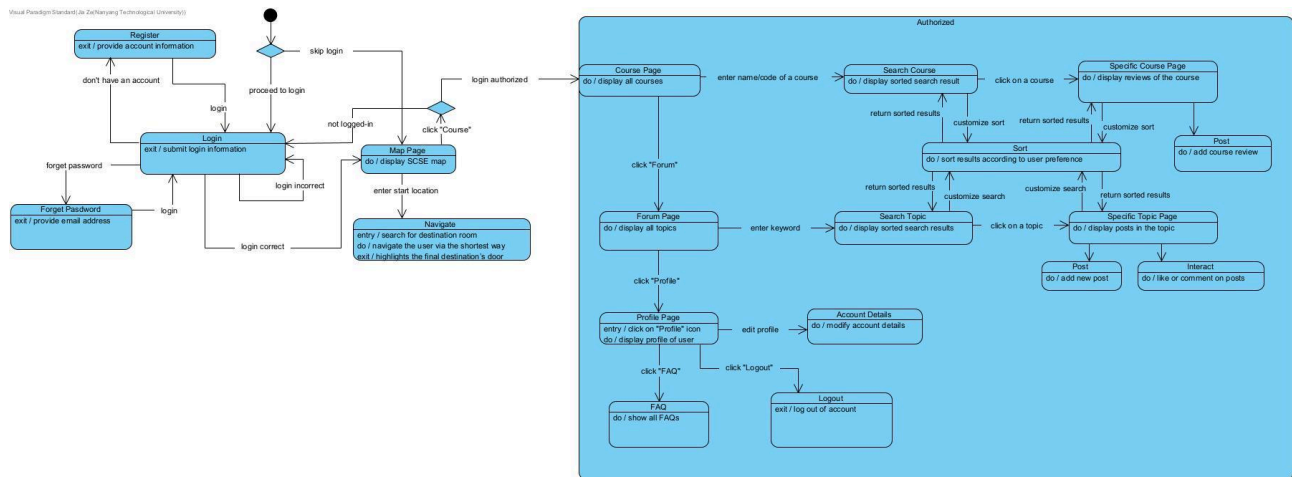


Change Password



2.6 Dialog Map

Visual Paradigm Standard (© Zhejiang Technology University)



3. Non-functional Requirements

3.1 Usability Requirements

1. The application must be thumb friendly.
 - 1.1. The pressable button area should be bigger than 0.5 cm in length and height.
 - 1.2. The spacing between buttons should be bigger than 0.5 cm in length and height.
2. The application must be consistent.
 - 2.1. The color palettes of the design must be the same across the different pages.
 - 2.2. The set of fonts used in the design must be the same across the different pages.
3. The application must allow easy reverse of action.
 - 3.1. Users must be able to label and un-label courses as favorite.
 - 3.2. Users must be able to like and unlike posts in the forums.

3.2 Performance Requirements

1. Upon opening the app, the login page should appear within 1 second.
2. The search of rooms in maps should be completed within 1 second.
3. The search for courses should be completed within 1 second.
4. The sorting of courses should be completed within 1 second.
5. The search for forum posts should be completed within 1 second.
6. The sorting of forum posts should be completed within 1 second.
7. The account creation process should be completed within 2 minutes.

3.3 Security Requirements

1. The database storing the account information must undergo a hash algorithm before these information are stored.
2. When users are logging in, the system must input the login details they entered into a hash algorithm and compare it with the database.

4. Interface Requirements

4.1 Users

SCSEBuddy is tailored towards the needs of past, current, and prospective SCSE students, alumni, and faculties. In addition, SCSEBuddy works for all users who would benefit from the following functions:

1. Map navigation around SCSE building.
2. View or post course reviews.
3. View, post, like, or comment on forum topic posts.

4.2 Software

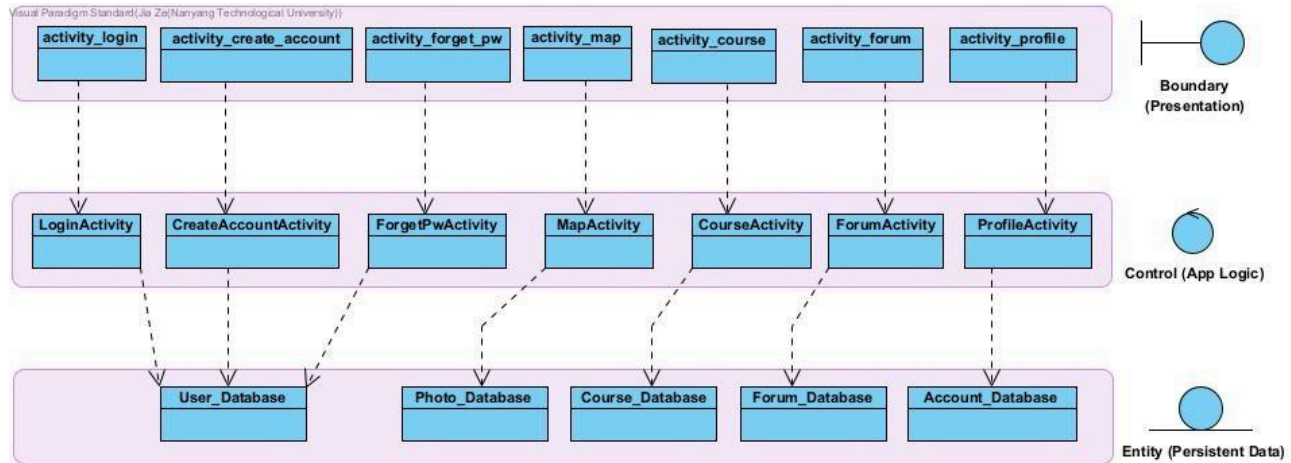
SCSEBuddy only works on devices running on android API level 29 and up.

4.3 Communication Interfaces

SCSEBuddy functions can only be accessed with connections to the internet.

5. Architecture Design

5.1 System Architecture Diagram



5.2 Design Pattern

5.2.1 Chain of Responsibility

Problem

When errors exist in the program, problems would occur if the system tries to proceed to further parts of the code that utilizes the errored part.

Solution

We adopted the design pattern of chain of responsibility, where requests are passed along a chain of handlers. Further functions will only be called after the results returned from the previous ones are valid. The use of this design pattern solves the problem because the program would not continue to the extended functionalities with an error already existing.

5.2.2 Mediator

Problem

Without a mediator, the application would become too complex as it consists of both the functionality of displaying the user interfaces, completing the frontend actions, while also querying the databases, implementing backend algorithms, including calculating the shortest path. Android phones' computation power might be limited for larger graphs, affecting the performance of the application.

Solution

The design pattern of Mediator is used to resolve this problem. The application will first access the server, followed by the server accessing the database. This way, the application can access the server only by http requests, calling functions using their respective names. Then, the server, which can only be accessed by admins managing the application, can be accessed by the database. The application itself only handles displaying user interfaces, and it does not need to know the backend skeleton, reducing chaotic dependencies between objects and ensuring the performance of the application on Android phones.

5.2.3 State**Problem**

A three layered architecture is used for our product, consisting of the application, the mediator function, and the database server. This means that problems might exist without efforts to synchronize the actions of the different parts. For example, one part performs an action not knowing that there is an error from the function that passes it its needed input.

Solution

We used the design pattern of State to resolve the issue. A http status code is passed in, signaling the different parts of the system. This way, their behavior may be altered based on the changes of states, allowing them to be aware of whether there is an error, and thus handle it in a proper way.

6. Data Dictionary

User	A person who downloads the app, agrees to the Terms and Conditions of Usage, creates an account and performs a successful login into the app at least once.
SCSE	School of Computer Science and Engineering. It is one of the six schools under the College of Engineering, Nanyang Technological University.
NTU	Nanyang Technological University.
SCSE Students	Matriculated undergraduates of NTU who are currently pursuing a single/double degree which is (jointly) offered by SCSE.
SCSE Professors	An academic staff with the rank of Associate Professor or Professor tenured by NTU. He/She teaches undergraduate courses offered by SCSE.
SCSE Staff	A non-academic staff member who is employed by NTU to work under SCSE.
SCSE Course	A unit of teaching that is offered by SCSE which lasts one academic semester. SCSE courses can be divided into 2 categories: compulsory and electives. SCSE courses consist of lectures, tutorials and labs.
Navigation/Map Page	A user interface which allows the user to view the structural layout of the SCSE building floor that they are currently on. Each map page consists of only one floor of the SCSE building.
360-Degree Photo	A type of panoramic photograph which is made by digitally stitching multiple photographs together to create a single photograph. When viewed through a dedicated application, this photograph offers a 360 degree view of the photographer's surroundings from a single point in space.
Pinned Location	A specific location on the map page which represents the search result of a user in the navigation map. A visual marker in the shape of a bubble will appear on the pinned location when the search is complete.
Course Review Page	The user interface which displays all course records in ascending or descending order, allows users to search for courses and allows users to post reviews or read reviews which are posted by other users.
Floor Number	A numbering system used to represent the level at which the User is currently located in the SCSE building. There are a total of 6 floors, starting from B4 (basement level 4), B3 (basement level 3), B2 (basement level 2), B1 (basement level 1), 1 (first floor) and 2 (second

	floor).
Stair Directions	A set of instructions given to the user when the system requires the user to travel through a flight of stairs to reach the target destination. The instructions are in the format of “Go up/down to X floor”, where X denotes the floor number of the target destination.
Arrow Directions	A visual marker used to represent the direction which has to be taken by the user to reach the user’s target destination.
Room Number	The official alphanumeric code given to a room/laboratory/venue inside the SCSE building. Eg: N4-01A-03.
Course Name	The official full name given to a course offered by NTU. Eg: Software Engineering.
Course Code	The official alphanumeric identifier assigned to a course offered by NTU. Eg: SC2006.
Review	A user-generated post which purpose is to provide information or feedback about a particular course to other users. A review must contain the course code, date of publication, author’s name (if available), text description and numeric grade of the relevant course.
Numeric Grade	An ordinal scale in the range of 1 to 5 which is used to indicate the satisfaction level of a user towards a particular course. ‘1’ indicates least satisfied while ‘5’ indicates most satisfied.
Topic	A short text used to describe the theme of the post that the user is writing. It also serves as a way to categorize user posts in the forum.
Forum Page	The user interface which allows users to discuss topics publicly. Users can upload posts, like posts, comment on posts.
Like	An app feature which allows users to indicate their liking of another user’s posts. Each post can only be liked once by the same user. All users are able to see the total number of likes on any post.
Account	Data item used to indicate that a particular user has been granted personalized access to use the application. An account stores a user’s name, user email and user password.
AU	Academic Units. Under NTU’s Academic Unit System, each course is assigned a certain number of academic units. The academic unit is a measure of the student’s work-load associated with both class attendance and preparation.

7. Testing

7.1 Black Box Testing

7.1 1 Register Account

Scenario	Expected Result	Actual Result
Register with valid information and valid verification code.	System registers the user.	System registers the user.
Register with valid information but invalid verification code.	System prompts the user that the verification code entered is wrong.	System prompts the user that the verification code entered is wrong.
Register with invalid information.	System prompts the user that information entered is wrong.	System prompts the user that information entered is wrong.
Register with missing information.	System prompts the user regarding missing information.	System prompts the user regarding missing information.

Emails Used to Register

Email	Expected Result	Actual Result
"email@email.com"	Accepted	Accepted
"email"	Rejected	Rejected
"email@email"	Rejected	Rejected

Password Used to Register

Scenario	Expected Result	Actual Result
Password is at least 8 characters long and have at least one of each of the following: digit, lowercase, uppercase, special character.	The password is accepted by the system.	The password is accepted by the system.
Password shorter than 8 characters long and have at least one of each of the following: digit, lowercase, uppercase, special character.	The password is rejected by the system.	The password is rejected by the system.
Password is at least 8 characters long but is missing one or more of the following: digit, lowercase, uppercase, special character.	The password is rejected by the system.	The password is rejected by the system.

Scenario	Expected Result	Actual Result
"Pass@123"	The password is accepted by the system.	The password is accepted by the system.
"Pass@12"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"
"Pass@@@@"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"

"Pass1123"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"
"11223344"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"
"asdfasdf"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"
"ASDFASDF"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"
"!!@@!!@@"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"

First Name	Last Name	Email	Password	Re-Enter Password	Expected Output	Actual Output
"First"	"Last"	"Email@gmail.com"	"Pass@123"	"Pass@123"	Show verification code page	Show verification code page
	"Last"	"Email@gmail.com"	"Pass@123"	"Pass@123"	"Empty First Name"	"Empty First Name"
"First"		"Email@gmail.com"	"Pass@123"	"Pass@123"	"Empty Last Name"	"Empty Last Name"
"First"	"Last"		"Pass@123"	"Pass@123"	"Email is not valid"	"Email is not valid"
"First"	"Last"	"Email"	"Pass@123"	"Pass@123"	"Email is not valid"	"Email is not valid"
"First"	"Last"	"UsedEmail@gmail.com"	"Pass@123"	"Pass@123"	"UsedEmail@gmail.com is used"	"UsedEmail@gmail.com is used"
"First"	"Last"	"Email@gmail.com"	"Pass@1233"	"Pass@123"	"Password does not match!"	"Password does not match!"
"First"	"Last"	"Email@gmail.com"	"Pass@123"	"Pass@1233"	"Password does not match!"	"Password does not match!"

Verification Code

Verification Code	Expected Output	Actual Output
Correct verification code	Signed up successfully.	Signed up successfully.
Wrong verification code	Wrong verification code.	Wrong verification code.
Wrong verification code length	Wrong verification code.	Wrong verification code.

7.1.2 Log In

Scenario	Expected Result	Actual Result
Log in with a valid email and password.	System logs in user.	System logs in user.
Log in with wrong information.	System prompts the user that information entered is wrong.	System prompts the user that information entered is wrong.
Log in without information.	System prompts the user that information entered is wrong.	System prompts the user that information entered is wrong.
User skips log in.	System only allows users to use the navigation.	System only allows users to use the navigation.

Email	Expected Result	Actual Result
"email@email.com"	Accept	Accept
"email"	Reject	Reject
"email@email"	Reject	Reject

Email	Password	Expected Output	Actual Output
"email@email.com"	"Pass@123"	System logs in user.	System logs in user.
"email"	"Pass@123"	"Email or Password is wrong."	"Email or Password is wrong."
"	"Pass@123"	"Email or Password is wrong."	"Email or Password is wrong."
"email@email.com"	"	"Email or Password is wrong."	"Email or Password is wrong."
"email@email.com"	"Pass@12"	"Email or Password is wrong."	"Email or Password is wrong."

7.1.3 Maps

Searching For A Room

Scenario	Expected Result	Actual Result
Start and destination locations exist.	Display guide for users.	Display guide for users.
Start location does not exist.	System prompts the user that information entered is wrong.	System prompts the user that information entered is wrong.
Destination location does not exist.	System prompts the user that information entered is wrong.	System prompts the user that information entered is wrong.
Start and destination locations do not exist.	System prompts the user that information entered is wrong.	System prompts the user that information entered is wrong.

Start Location	Destination Location	Expected Result	Actual Result
"SCSE Entry"	"Software Lab 1"	Display navigation for users to follow.	Display navigation for users to follow.
"Jurong point"	"Software Lab 1"	"Start or End Location does not exist."	"Start or End Location does not exist."
"Software Lab 1"	"Jurong point"	"Start or End Location does not exist."	"Start or End Location does not exist."
"Jurong point"	"NTU"	"Start or End Location does not exist."	"Start or End Location does not exist."

7.1.4 Course

Searching For A Course

Scenario	Expected Result	Actual Result
Search for a course that exists.	Display course for users.	Display course for users.
Search for part of a course code that exists.	Display all courses containing the searched part for users.	Display all courses containing the searched part for users.
Search for a course that does not exist.	Display error for users.	Display error for users.

Searched Course Codes	Expected Result	Actual Result
"CC0007"	Display search result containing CC0007 for users.	Display search result containing CC0007 for users.
"CC"	Display all search results containing CC for users.	Display all search results containing CC for users.
"CC0008"	"No search results"	"No search results"

Adding Course To Favorites

Scenario	Expected Result	Actual Result
Add to favorite	Move the course to the top of the list.	Move the course to the top of the list.
Remove from favorite	Remove the course from the top of the list.	Remove the course from the top of the list.

7.1.5 Forum

Search

Scenario	Expected Result	Actual Result
Search for a post that exists.	Display the post to users.	Display the post to users.
Search for part of a word that exists in forum posts.	Display all posts containing the searched part for users.	Display all posts containing the searched part for users.
Search for a post that does not exist in forum posts.	Display error for users.	Display error for users.

Searched Course Codes	Expected Result	Actual Result
"Internships"	Display posts containing the keyword "Internships" to users.	Display posts containing the keyword "Internships" to users.
"Inte"	Display posts with words containing the string "Inte" to users.	Display posts with words containing the string "Inte" to users.
"Intelships"	"No search results"	"No search results"

Creating A Post

Scenario	Expected Result	Actual Result
Post contains a title and descriptions.	Uploads user posts.	Uploads user posts.
Post contains a title but no descriptions.	Display error for users.	Display error for users.
Post contains no title but has descriptions.	Display error for users.	Display error for users.

Title	Descriptions	Expected Result	Actual Result
"Hello World!"	"I love SC2006"	Uploads user posts.	Uploads user posts.
"Hello World!"	"	"Post requires title and descriptions to be filled."	"Post requires title and descriptions to be filled."
"	"I love SC2006"	"Post requires title and descriptions to be filled."	"Post requires title and descriptions to be filled."

7.1.6 Profile

Change Password

Scenario	Expected Result	Actual Result
Change password to valid password.	Password has been updated. The system returns to the profile page.	Password has been updated. The system returns to the profile page.
Change password to invalid password.	The system displays an error message to the users.	The system displays an error message to the users.

New Password	Re-enter New Password	Expected Result	Actual Result
pass1234	pass1234	Password has been updated to pass1234. The system returns to the profile page.	Password has been updated to pass1234. The system returns to the profile page.
pass1234	pass1334	"New Password Not Same!"	"New Password Not Same!"
111	111	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"	"Password does not contain 1 digit or 1 lower or 1 special character. or It does not have 8 characters"

Change Profile Display Name

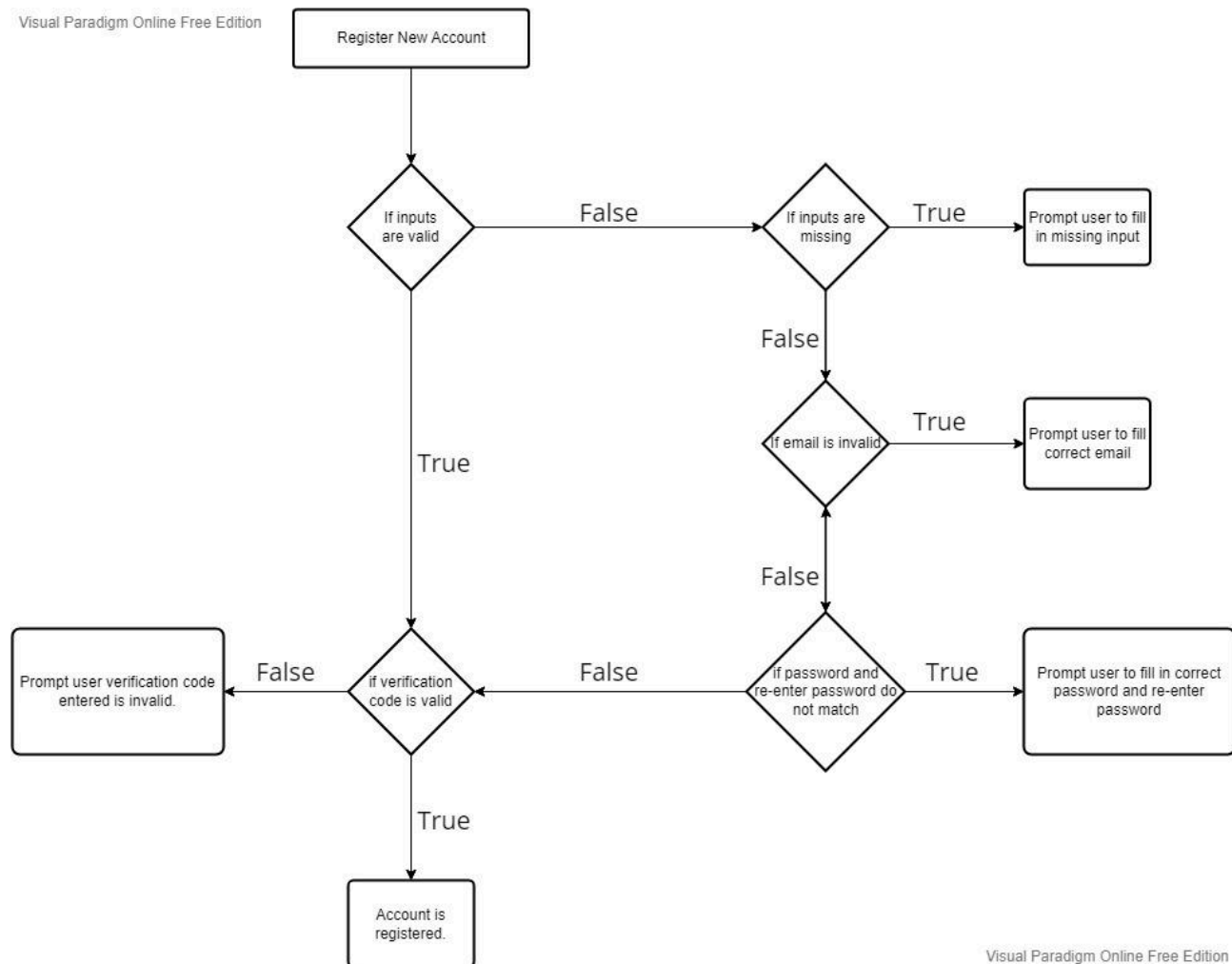
Scenario	Expected Result	Actual Result
Change profile name to a valid entry.	Profile name has been updated. The system returns to the profile page.	Profile name has been updated. The system returns to the profile page.

Old First Name	Old Last Name	New First Name	New Last Name	Expected Result	Actual Result
Harry	Potter	Happy		Profile name has been changed to Happy Potter.	Profile name has been changed to Happy Potter.
Harry	Potter			Profile name remains Harry Potter.	Profile name remains Harry Potter.
Harry	Potter	Happy	Potter	Profile name has been changed to Happy Potter.	Profile name has been changed to Happy Potter.
Harry	Poter		Potter	Profile name has been changed to Harry Potter.	Profile name has been changed to Harry Potter.

7.2 White Box Testing

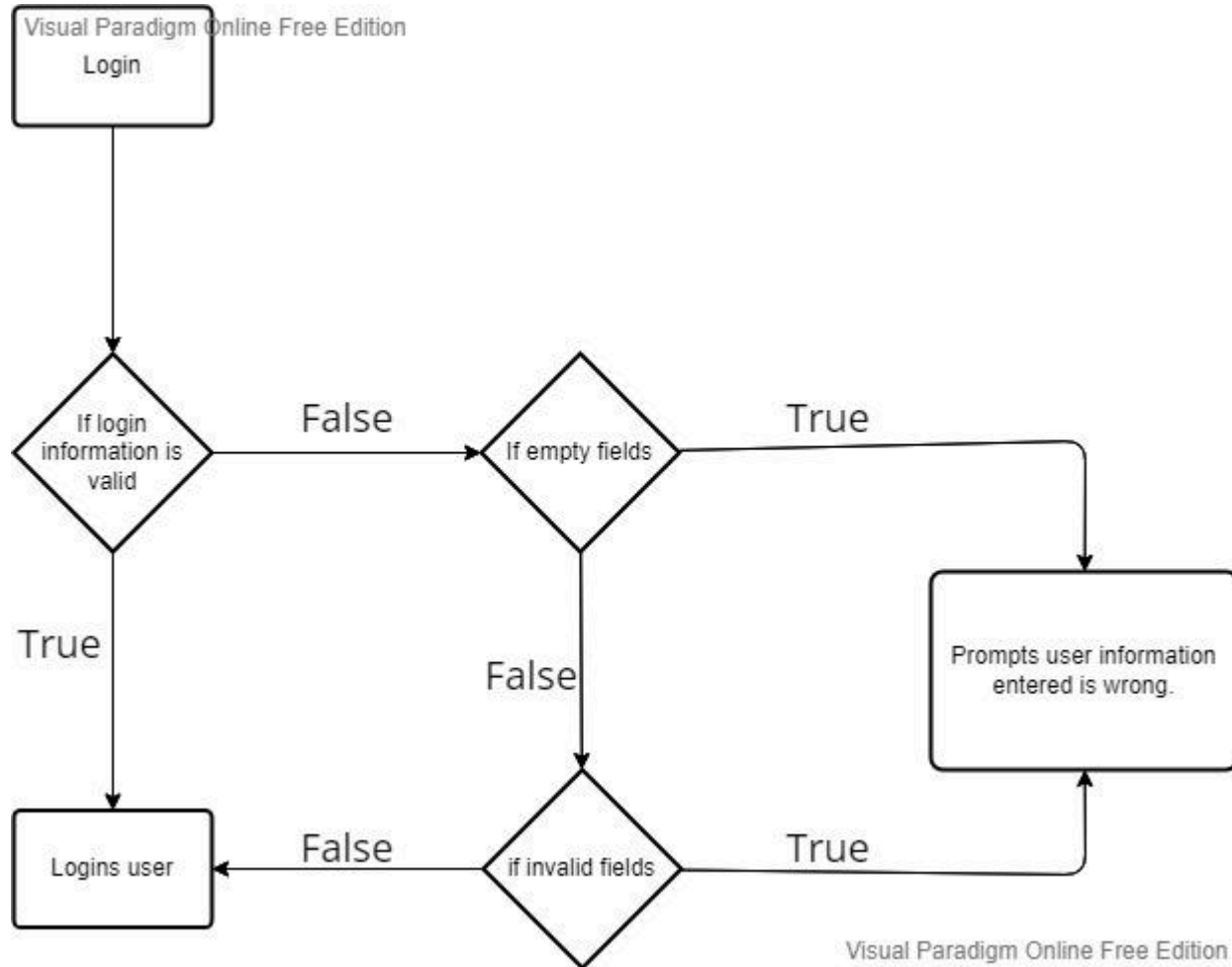
7.2.1 Register Account

Visual Paradigm Online Free Edition

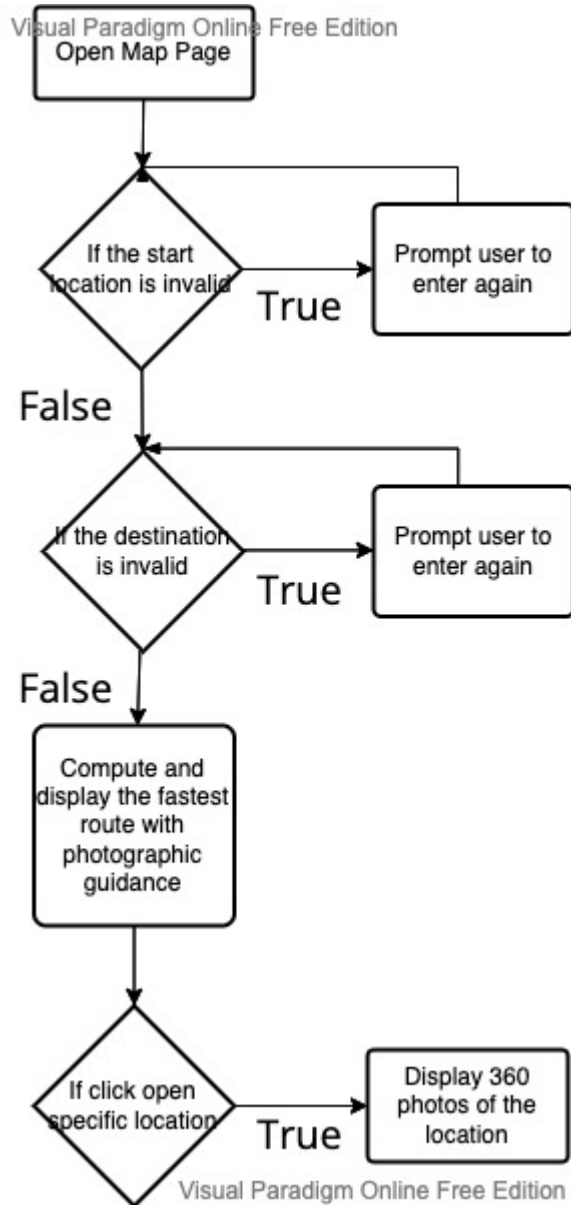


Visual Paradigm Online Free Edition

7.2.2 Log in

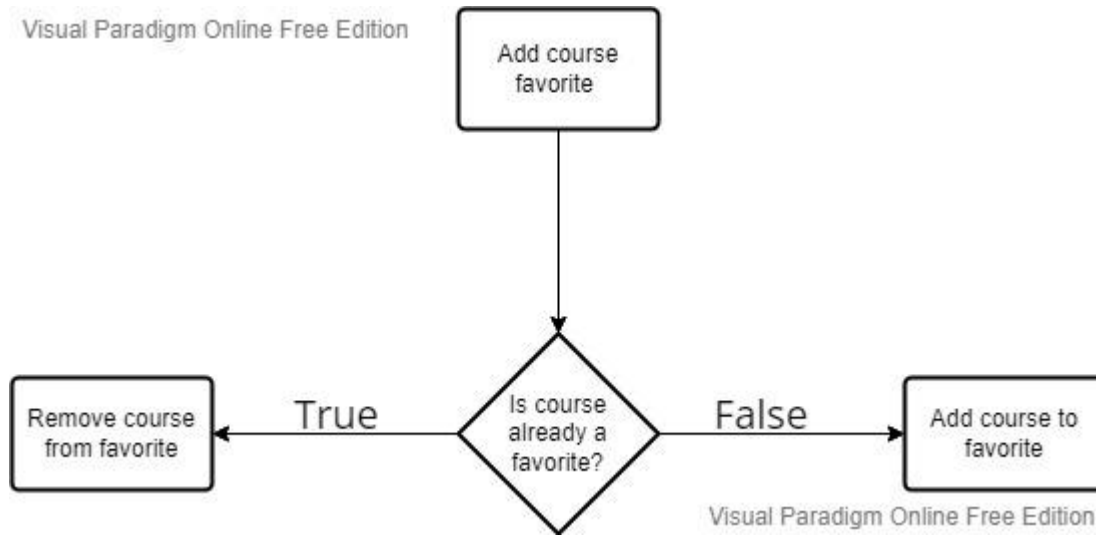


7.2.3 Map Navigation

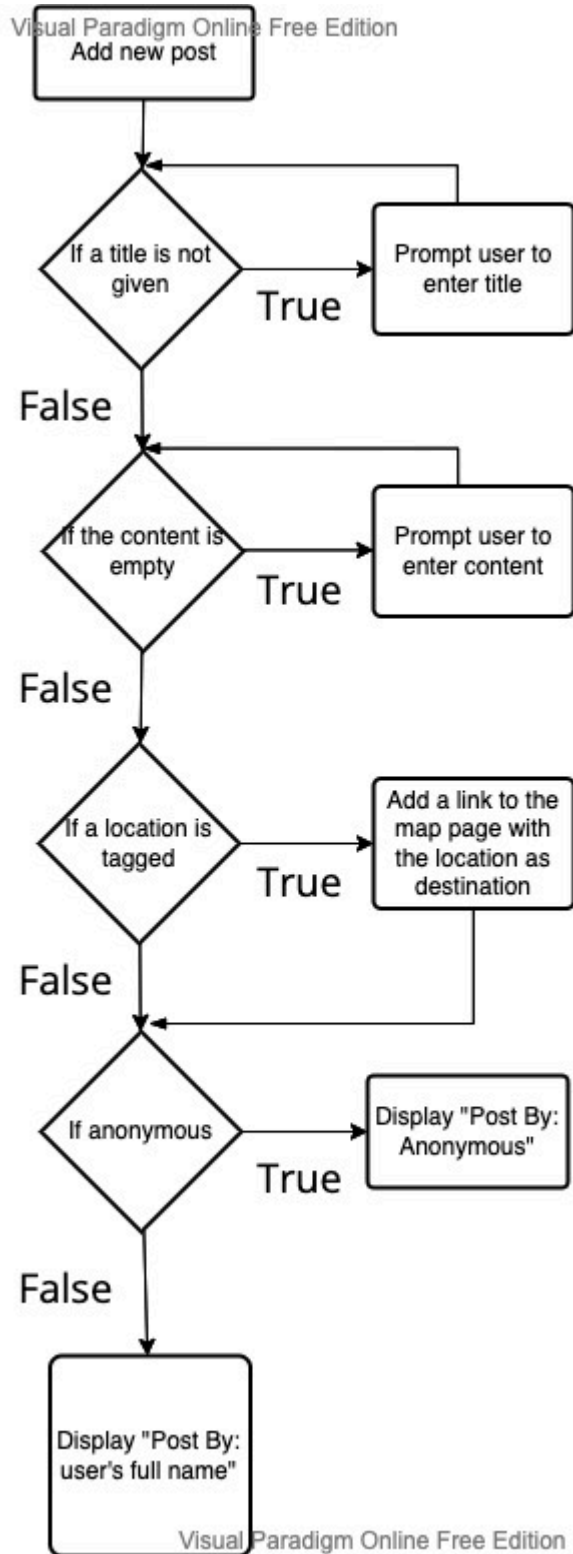


7.2.4 Course Favorite

Visual Paradigm Online Free Edition



7.2.5 Forum Post



Appendix A: Video Demonstration

For a demonstration of our application, you may view the videos at the following link:

Trailer video: <https://youtu.be/3LJqkY7j-Qs>

SCSEBuddy demonstration: <https://youtu.be/lf7BrzpTVg4>