Software Requirements Specification

for

edUFind

Version 1.0 approved

Prepared by Team UPS-Elon

Nanyang Technological University

15 Feb 2022

Functional Requirements **3**

Non-Functional Requirements **5**

Data Dictionary **6**

Use Case Diagram **8**

Use Case Descriptions **9**

UI Mockup **25**

- Home Page 25
- Login Page 26
- Sign Up Procedures 27
- Recommendations 28
- Primary Schools 29
- Secondary School 30
- Junior College 31
- School details 32
- Shortlisted Schools 32
- Account Settings 34

Class Diagram **35**

Sequence Diagram **36**

- Registering User Account 36
- Request Password Change 37
- Shortlist Schools 37
- Remove Shortlisted Schools 38
- Retrieve Shortlisted Schools 38
- Create Comment 39
- Edit Comment 39
- Display Comment 40
- Like Comment 40
- Search School 41
- Select School 41

Dialog Map **42**

Functional Requirements

1. Home Page
    1.1. The system must be able to display the list of schools based on their corresponding education level offered, indicated by the school's main level code
    1.2. The system must provide be able to allow the user to use the search box
    1.3. The system must allow the user to be able to filter schools of their choosing based on certain filters
    1.4. The system must allow the user to login to their account
    1.5. The system must allow the user must to view their shortlisted schools
    1.6. The system must allow the user to register their account
2. Search
    2.1. The system must allow the user to find and filter schools of their choosing based on
        2.1.1. Co-Curricular Activities (CCA)
        2.1.2. Elective Programmes
        2.1.3. Location
        2.1.4. Mother Tongue Language (MTL) offered
        2.1.5. Subjects Offered
    2.2. The system must allow the user to use the search box to search for schools directly
3. Creating User Account
    3.1. The system must allow the user to create an account by providing the following information
        3.1.1. Username
        3.1.2. Email
        3.1.3. Password
    3.2. The system must validate that all required information is provided by the user
4. Login
    4.1. The system must allow the user to be able to login to their account using their email and password
    4.2. The system must be able to check that the email and password are correct
5. Manage Profile
    5.1. The system must allow the user to change their password to ensure account security
    5.2. The system must allow users to edit their preferences (9. Recommendations)
6. Shortlisting schools (For registered users)
    6.1. The system must allow the user to shortlist schools
    6.2. The system must allow the user to delete shortlisted schools
    6.3. The system must allow the user to view shortlisted schools
7. Comments (For registered users)
    7.1. The system must allow the user to create comments
    7.2. The system must allow the user to edit comments
    7.3. The system must allow the user to delete comments

7.4. The system must allow the user to like comments

8. Displaying School Specific Data
    8.1. The system must be able to display the following information about multiple schools based on the filters selected by the user
        8.1.1. Name
        8.1.2. Address
    8.2. The system must be able to display the following information about the school selected by the user
        8.2.1. School Name
        8.2.2. School Website
        8.2.3. Address
        8.2.4. MRT Description
        8.2.5. Bus Description
        8.2.6. Type Code
        8.2.7. Nature Code (Co-Ed, All boys, All Girls)
        8.2.8. MTL Offered
        8.2.9. Subjects Offered
        8.2.10. Elective Programmes
        8.2.11. Support for Special Educational Needs
9. Recommendation
    9.1. System will enquire registered user for his/her preferences on first time log-in
        9.1.1. Region
        9.1.2. Mother Tongue
        9.1.3. Gender
        9.1.4. CCA Categories
    9.2. System must display schools relevant to user’s selected preferences

10. Interfacing with data.gov.sg API

Non-Functional Requirements

1. Usability: Each tab will be clearly described so that the users can understand easily.
2. Reliability: System should be able to support up to 50 pax.
3. Performance: The system will be able to respond to user interactions within 2 seconds.
4. Supportability: Documentation and coding comments are consistent and traceable (with version control). The git update will be well commented and described appropriately.

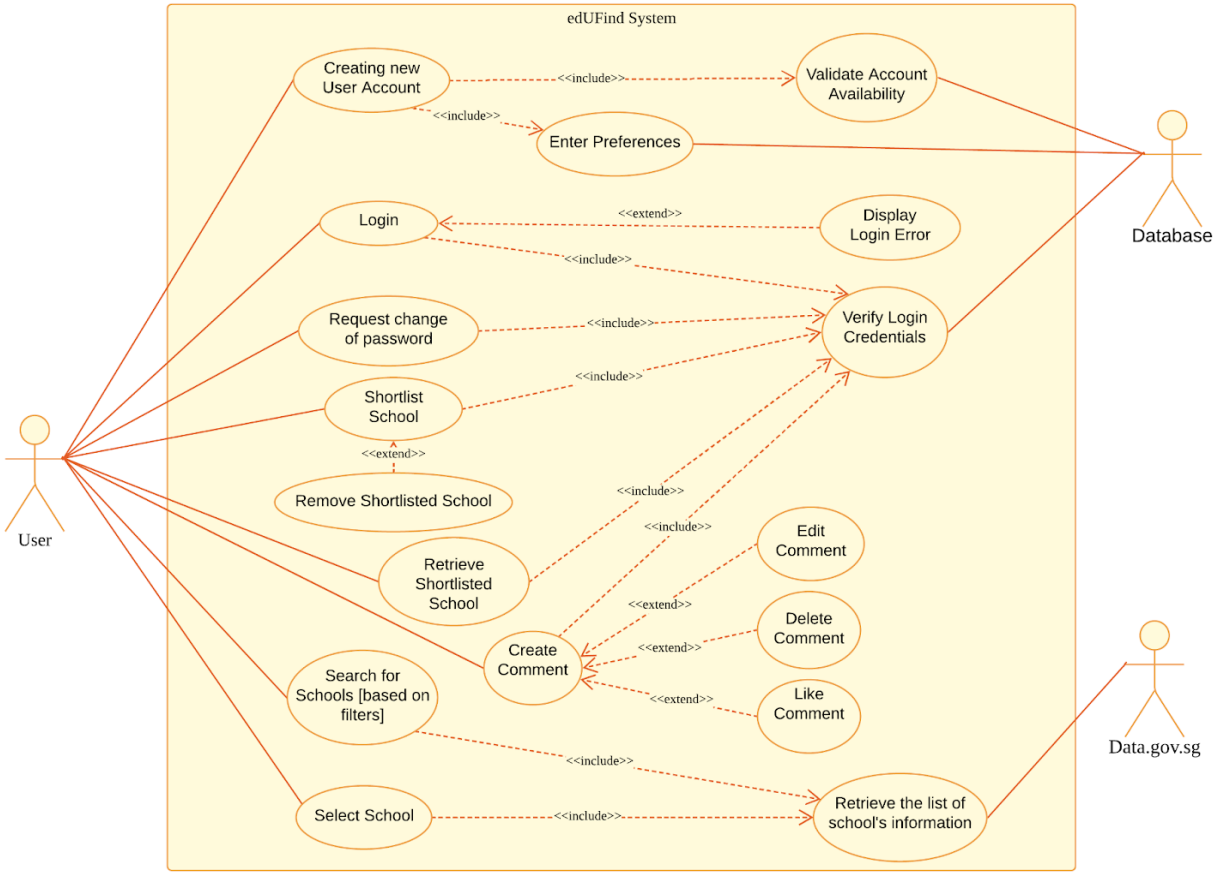
Data Dictionary

Term	Definition
User	A user is a person who is using the web application to find schools based on their interest and location in Singapore
System	A system refers to the web application
Search	Search is a feature that allows user to find schools based on school's name or certain keywords
Filter	Filter is a feature that the user can use to sort schools based on the school's characteristics (E.g., CCA, Subjects offered, etc.)
Shortlist	Shortlist is a feature that enables registered users to mark/save their preferred school. Users are also able to delete schools from their list or retrieve their list of shortlisted schools
Registration	Registration is a feature for users to create a new account (provided their email is valid). This enables users to use other features of the website, such as shortlisting and commenting
Registered User	A user that has created an account in the system and is logged into that account
Comment	A written remark by a user to denote their views on a particular school. The author of the comment can delete or modify the comment while other registered users can 'like' comments.
Nature Code	Refers to the gender that the school allows to be enrolled. Values include "Co-Ed", "Boys only" or "Female only"
Type Code	Refers to the type of school. Values include "Government School", "Government Aided School", "Independent School", "Specialized Independent School" and "Specialized School"
Elective Program	Types of learning programmes that the school's adopt, such as Chinese Language Elective Program (CLEP), Art Elective Program (AEP), etc
Education Level	The education requirement required to enter a school (Primary, Secondary and Junior College)
Main Level Code	Refers to education level reflected in the database. Values include "Primary", "Secondary", "Junior College", "Mixed Levels" and "Centralised Institute"
Co-Curricular Activities (CCA)	A non-academic activity that all students must undertake as part of their education

	There are 4 main categories of CCAs ● Clubs and societies ● Physical sports ● Uniformed groups ● Visual and performing arts
Mother Tongue Language (MTL)	A mother tongue language is a second language (the first being English) that a student will learn in school (such as Chinese, Tamil and Malay)
Subject	The various knowledge studied or taught in a school that are currently offered to the enrolling student cohort
Special Education Needs	Types of special needs the schools can support (E.g., dyslexia, ADHD, mild Autism Spectrum Disorder, mild hearing loss, visual impairment and physical impairment)

Use Case Diagram

Refer to PDF File for the clear image



Use Case Descriptions

Use Case ID:	1		
Use Case Name:	Creating New User Account		
Created By:	Lim Kai Sheng	Last Updated By:	Andre Lim
Date Created:	28th January 2022	Date Last Updated:	10th February 2022
Actor	User		
Description:	User creates a new account		
Precondition:	1. User account must not exist in the database 2. User must have an internet connection		
Postconditions:	1. User account must be updated in database 2. User can now log in using the created account		
Priority:	Medium		
Frequency of Use:	0 - 1 time per lifetime		
Flow of Events:	1. User keys in a valid username, email, password and confirm password in the respective fields 2. User clicks the “create account” button 3. System verifies the password and confirm password fields are identical		
Alternative Flows:	AF-S3: System detects discrepancy between the password and confirm password 1. System displays error message 2. Return to Step 1		
Exceptions:	-		
Includes:	1. Validate Account Availability 2. Enter Preferences		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	2		
Use Case Name:	Validate Account Availability		
Created By:	Andre Lim	Last Updated By:	
Date Created:	28th January 2022	Date Last Updated:	
Actor	System		
Description:	Only allow user to register an account if the email is valid		
Precondition:	1. User account must not exist in the database 2. User must have an internet connection		
Postconditions:	1. User account must be updated in database 2. User can now log in using the created account		
Priority:	Medium		
Frequency of Use:	1 - 3 times per lifetime		
Flow of Events:	1. System will validate the account availability in the database 2. System will add the user account in the database		
Alternative Flows:	AF-S1: System detects the user account has already exists in the database 1. System will display error message “Account with corresponding email already exist” 2. User enters a new email 3. User clicks the “create account” button to re-attempt registration again. 4. Return to Step 1		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	3		
Use Case Name:	Login		
Created By:	Andrew Ng	Last Updated By:	
Date Created:	28th January 2022	Date Last Updated:	
Actor	User		
Description:	User logs into his/her created account using the registered email and password		
Precondition:	1. User must have a registered account in the database 2. User must have an internet connection		
Postconditions:	1. User has access to his/her account 2. User can use additional features of the website such as Shortlisting Schools, Commenting etc		
Priority:	Medium		
Frequency of Use:	1 - 3 times per lifetime		
Flow of Events:	1. User enters their email, and password in the login field 2. User clicks on the “login” button 3. System authenticates the account with the database		
Alternative Flows:	AF-S2: User left the required field empty 1. System will display an error message 2. User fills up the required field(s) 3. Return to Step 2 AF-S3: User enters an invalid email or password 1. System will display an error message 2. User re-enters the respective field(s) 3. Return to Step 2		
Exceptions:	-		
Includes:	1. Verify Login Credentials		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	4		
Use Case Name:	Display Login Error		
Created By:	Lim Kai Sheng	Last Updated By:	
Date Created:	28th January 2022	Date Last Updated:	
Actor	System		
Description:	Displays “Invalid Email or Password entered” to the user		
Precondition:	1. User enters an invalid email or password when logging in 2. User must have an internet connection		
Postconditions:	1. User is prompted to re-enter a valid email and password		
Priority:	Medium		
Frequency of Use:	1 - 3 times per lifetime		
Flow of Events:	1. User enters an invalid email and/or password 2. Error message is displayed to the user 3. User is prompted to re-enter a valid email and password		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	5		
Use Case Name:	Verify Login Credentials		
Created By:	Andrew Ng	Last Updated By	
Date Created:	28th January 2022	Date Last Updated:	
Actor	System		
Description:	Authenticate account information via email and password		
Precondition:	1. User account must already exist in the database 2. User must have an internet connection		
Postconditions:	1. User can shortlist their shortlisted schools 2. User can retrieve their list of shortlisted schools 3. User can remove their shortlisted schools from the list		
Priority:	Medium		
Frequency of Use:	1 - 3 times per lifetime		
Flow of Events:	1. User enters the respective login fields 2. User clicks on the “login” button 3. System verifies the user’s credentials against the database. 4. User to login successfully		
Alternative Flows:	AF-S2: User left the required field empty 1. System will display an error message 2. User fills up the required field(s) 3. Return to Step 2 AF-S3: User enters an invalid email or password 1. System will display an error message 2. User re-enters the respective field(s) 3. Return to Step 2		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	6		
Use Case Name:	Request change of password		
Created By:	Andre Lim		
Date Created:	28th January 2022		
Actor	User		
Description:	User requests for a change of password		
Precondition:	1. User must be logged into his/her account 2. User must have an internet connection		
Postconditions:	1. Password must be updated in database		
Priority:	Medium		
Frequency of Use:	1 - 3 times per year		
Flow of Events:	1. User logs into his/her account 2. User clicks on his/her account 3. User selects the “change password” option 4. System prompts user to key in the old password, new password and confirm new password fields 5. System validates the old password 6. System validates the new and confirm password are identical 7. User’s account password is updated to the new password		
Alternative Flows:	AF-S5: User enters an invalid old password 1. System will display an error message 2. User re-enters all three field(s) 3. Return to Step 5 AF-S6: User enters a mismatch of new passwords 1. System will display an error message 2. User re-enters the respective field(s) 3. Return to Step 5		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	7		
Use Case Name:	Shortlist School		
Created By:	Lim Kai Sheng	Last Updated By	Ling Yin
Date Created:	28th January 2022	Date Last Updated:	10th February 2022
Actor	User		
Description:	User can add schools they are interested in to their list of shortlisted schools		
Precondition:	1. User account must already exist in the database 2. User must be logged into his/her account 3. User must have an internet connection		
Postconditions:	1. User's preferred schools will be saved in the database		
Priority:	High		
Frequency of Use:	0 - 10 times per day		
Flow of Events:	1. User searches for schools based on preference 2. User clicks on the "My Shortlist" button to bookmark their shortlisted schools. 3. System will save the user's shortlisted schools into the database.		
Alternative Flows:	-		
Exceptions:	-		
Includes:	1. Verify login credentials		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	8		
Use Case Name:	Remove Shortlisted School		
Created By:	Andre Lim	Last Updated By	Andre Lim
Date Created:	28th January 2022	Date Last Updated:	10th February 2022
Actor	User		
Description:	Shortlisted schools can be removed by the user		
Precondition:	1. User account must already exist in the database 2. User must be logged into his/her account 3. User must have an internet connection		
Postconditions:	1. User's selected school will be removed from his/her shortlisted list		
Priority:	High		
Frequency of Use:	0 - 5 times per day		
Flow of Events:	1. User clicks on the "My Shortlist" button to view their shortlisted schools 2. User selects the shortlisted schools that he/she wants to remove 3. User selects "Delete" 4. System will drop the user's shortlisted schools from the database		
Alternative Flows:	-		
Exceptions:	-		
Includes:	1. Verify login credentials		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	9		
Use Case Name:	Retrieve Shortlisted School		
Created By:	Andrew Ng	Last Updated By	Lim Kai Sheng
Date Created:	28th January 2022	Date Last Updated:	10th February 2022
Actor	User		
Description:	Displays list of shortlisted schools		
Precondition:	1. User account must already exist in the database 2. User must be logged into his/her account 3. User must have an internet connection		
Postconditions:	1. A list of User's shortlisted schools will be displayed		
Priority:	High		
Frequency of Use:	0 - 5 times a day		
Flow of Events:	1. User selects "My Shortlist" 2. List of shortlisted schools for the user is fetched from the database 3. System displays the retrieved shortlisted school(s) to the User		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	10		
Use Case Name:	Create Comment		
Created By:	Lim Kai Sheng	Last Updated By	
Date Created:	28th January 2022	Date Last Updated:	
Actor	User		
Description:	User is able to post their comments about schools		
Precondition:	1. User account must already exist in the database 2. User must be logged into his/her account 3. User must have an internet connection		
Postconditions:	1. Comments are displayed in the comment section of the respective school		
Priority:	Medium		
Frequency of Use:	0 - 5 times per year		
Flow of Events:	1. User types in his/her comments in comment box 2. User clicks “post” button 3. Comment is now publicly displayed in the comment section of the school		
Alternative Flows:	-		
Exceptions:	-		
Includes:	1. Verify Login Credentials		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	11		
Use Case Name:	Edit Comment		
Created By:	Andre Lim	Last Updated By	
Date Created:	28th January 2022	Date Last Updated:	
Actor	User		
Description:	User can edit his/her posted comment		
Precondition:	1. User account must already exist in the database 2. User must be logged into his/her account 3. User must have already posted a comment 4. User must have an internet connection		
Postconditions:	1. User's comment is edited and displayed in the comment section of the respective school		
Priority:	Medium		
Frequency of Use:	0 - 4 times per year		
Flow of Events:	1. User navigates to his comments 2. User clicks on "edit" button to start editing his post 3. User modifies his comment 4. User clicks on "confirm" button to modify his original comments		
Alternative Flows:	AF-S2: User decides not to edit his post 1. User clicks on "cancel button" 2. Comment is not modified		
Exceptions:	-		
Includes:	1. Verify Login Credentials		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	12		
Use Case Name:	Delete Comment		
Created By:	Andrew Ng	Last Updated By	
Date Created:	28th January 2022	Date Last Updated:	
Actor	User		
Description:	User can delete his/her posted comment		
Precondition:	1. User account must already exist in the database 2. User must be logged into his/her account 3. User must have already posted a comment 4. User must have an internet connection		
Postconditions:	1. User's comment is removed from the comment section of the respective school		
Priority:	Medium		
Frequency of Use:	0 - 4 times per year		
Flow of Events:	1. User navigates to his comments 2. User clicks on "delete" button 3. User clicks "confirm" button to delete his/her comment 4. Comment is removed		
Alternative Flows:	AF-S3: User decides not to delete his/her comment 1. User clicks on the "cancel" button 2. Comment is not deleted		
Exceptions:	-		
Includes:	1. Verify Login Credentials		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	13		
Use Case Name:	Like Comment		
Created By:	Lim Kai Sheng	Last Updated By	
Date Created:	28th January 2022	Date Last Updated:	
Actor	User		
Description:	User can like other posted comment		
Precondition:	1. User account must already exist in the database 2. User must be logged into his/her account 3. User must have an internet connection		
Postconditions:	1. “Liked” should be reflected on the intended comment		
Priority:	Medium		
Frequency of Use:	0 - 4 times per year		
Flow of Events:	1. User clicks on the “like” button 2. The system will update the comment’s “Liked” counts		
Alternative Flows:			
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	User cannot un-like the comments he/she has made.		
Notes and Issues:	-		

Use Case ID:	14		
Use Case Name:	Search for Schools [based on filters]		
Created By:	Andre Lim	Last Updated By	
Date Created:	28th January 2022	Date Last Updated:	
Actor	User		
Description:	Filter schools based on user's preference		
Precondition:	1. User must have an internet connection		
Postconditions:	1. A list of schools will be displayed under the search bar based on the user's search query and filters		
Priority:	High		
Frequency of Use:	0-40 times per day		
Flow of Events:	1. User enters his query in the Search bar 2. System will provide suggestions based on User's query 3. User clicks on the "Search" button without adding any filters 4. System displays list of schools based on User's query		
Alternative Flows:	1. User selects the pre-defined filter from the drop-down 2. System will filter the schools based in user's input criteria 3. System will display the filtered schools		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

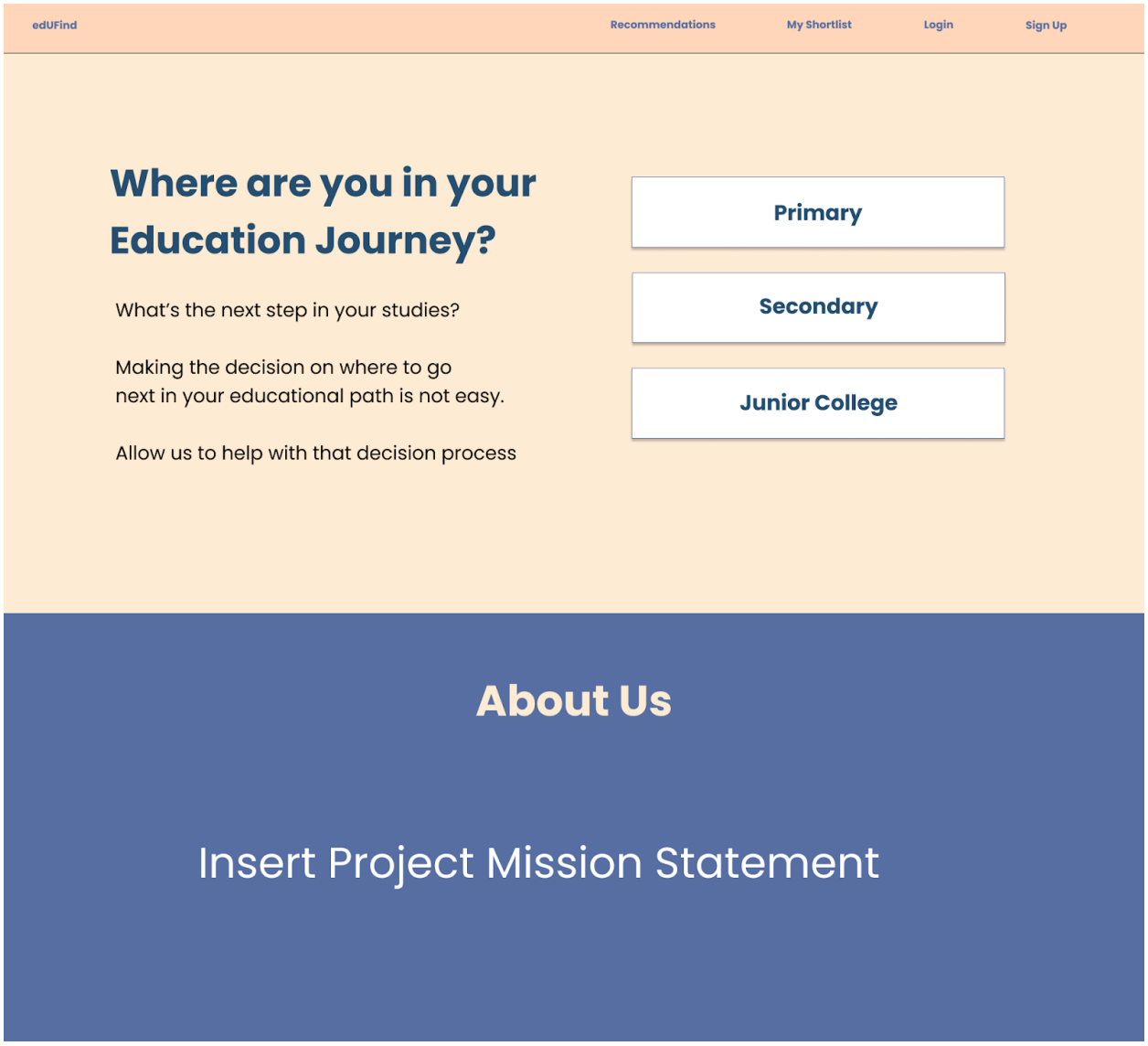
Use Case ID:	15		
Use Case Name:	Select School		
Created By:	Andrew Ng	Last Updated By	
Date Created:	28th January 2022	Date Last Updated:	
Actor	User		
Description:	User selects a school and its information is displayed		
Precondition:	1. User must have an internet connection		
Postconditions:	1. Selected school's information is displayed		
Priority:	High		
Frequency of Use:	0 - 20 times per day		
Flow of Events:	1. System displays a list of all schools 2. User selects a school 3. System displays the selected school's details		
Alternative Flows:	AF-S1: Filters have been applied 1. System displays a list of schools meeting the criteria of the filers 2. Return to Step 2		
Exceptions:	-		
Includes:	1. Retrieve the list of school's information		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

Use Case ID:	16		
Use Case Name:	Enter Preferences		
Created By:	Andrew Ng	Last Updated By	
Date Created:	10th February 2022	Date Last Updated:	
Actor	User		
Description:	System recommends a list of schools to the user based on their preferences		
Precondition:	1. User account must exist in the database 2. User must be logged into his/her account 3. User must have an internet connection		
Postconditions:	1. Display a list of schools based on user's preferences		
Priority:	Medium		
Frequency of Use:	1-3 time per account creation		
Flow of Events:	1. User selects Gender, Region, Level of Education, Mother Tongue and CCA Category 2. System displays a list of schools as recommendations		
Alternative Flows:	-		
Exceptions:	-		
Includes:	-		
Special Requirements:	-		
Assumptions:	-		
Notes and Issues:	-		

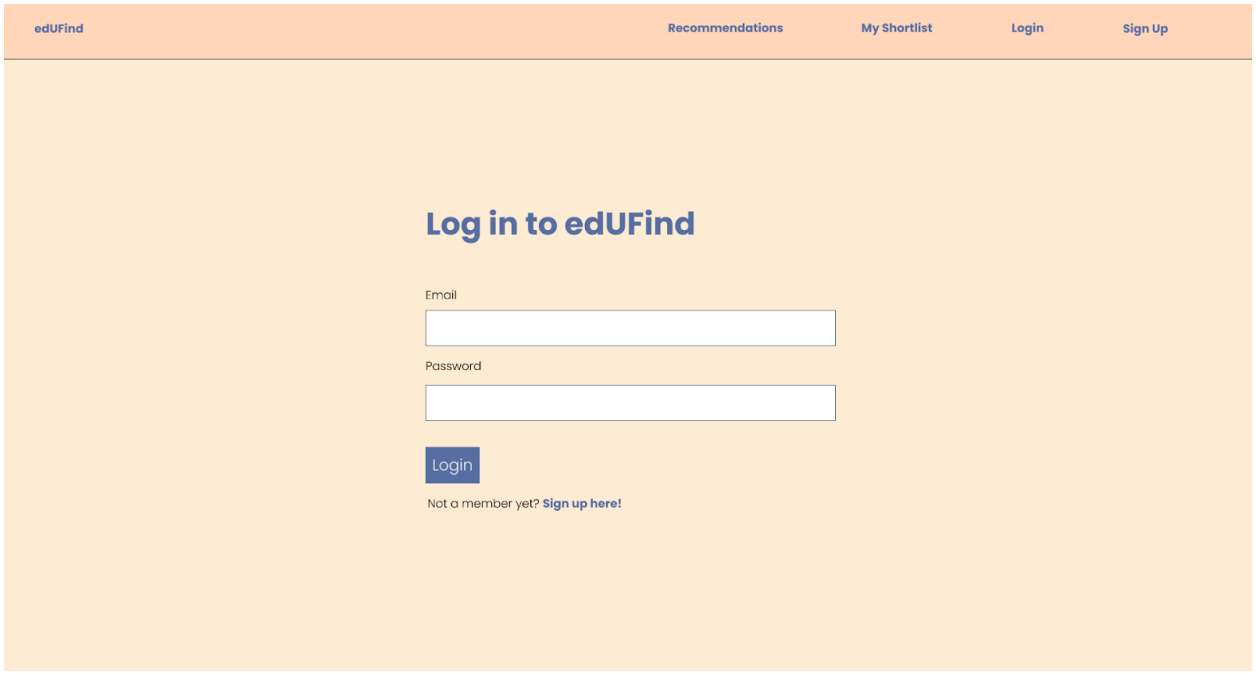
UI Mockup

Refer to UI Mockups v1.1.pdf F File for clearer images

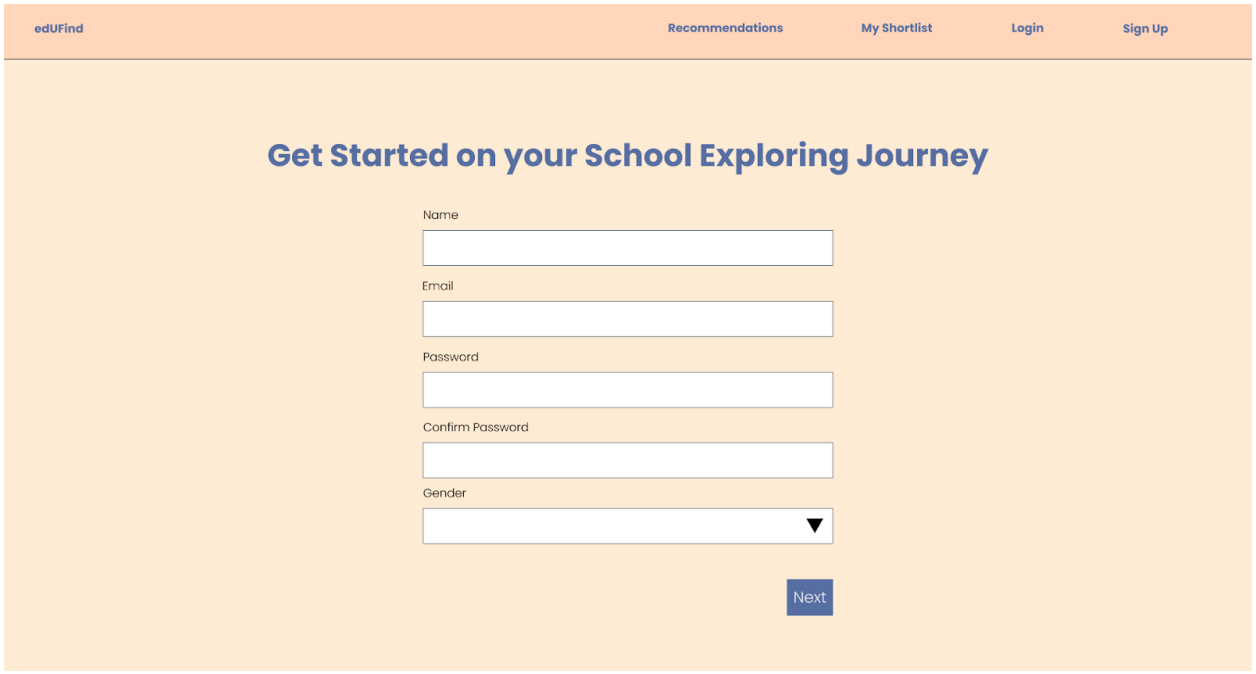
Home Page

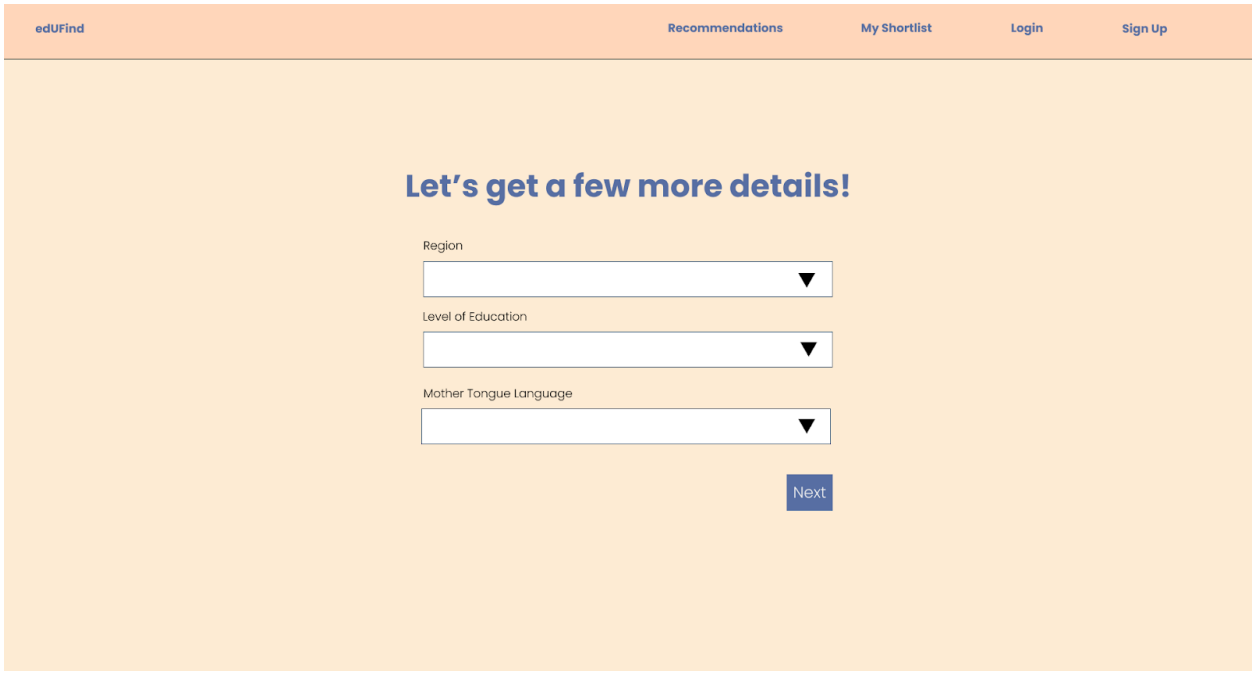


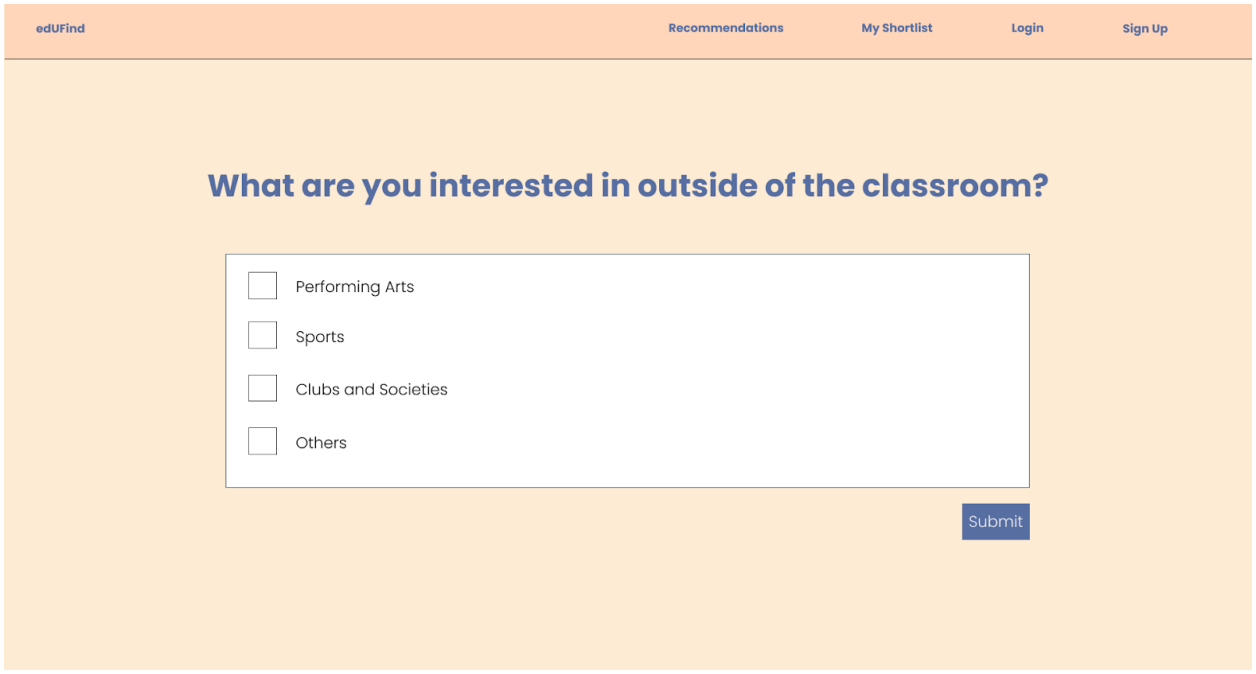
Login Page


edUFind
Recommendations
My Shortlist
Login
Sign Up
Log in to edUFind
Email
Password
Login
Not a member yet? Sign up here!

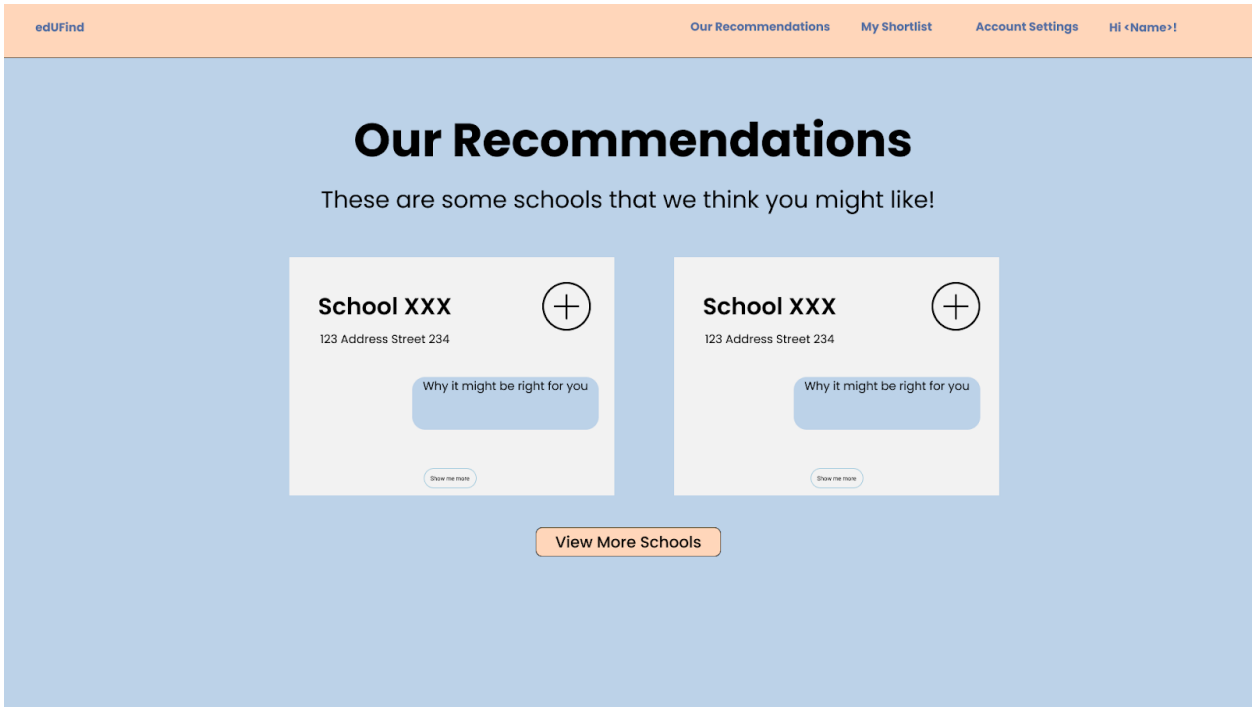
Sign Up Procedures



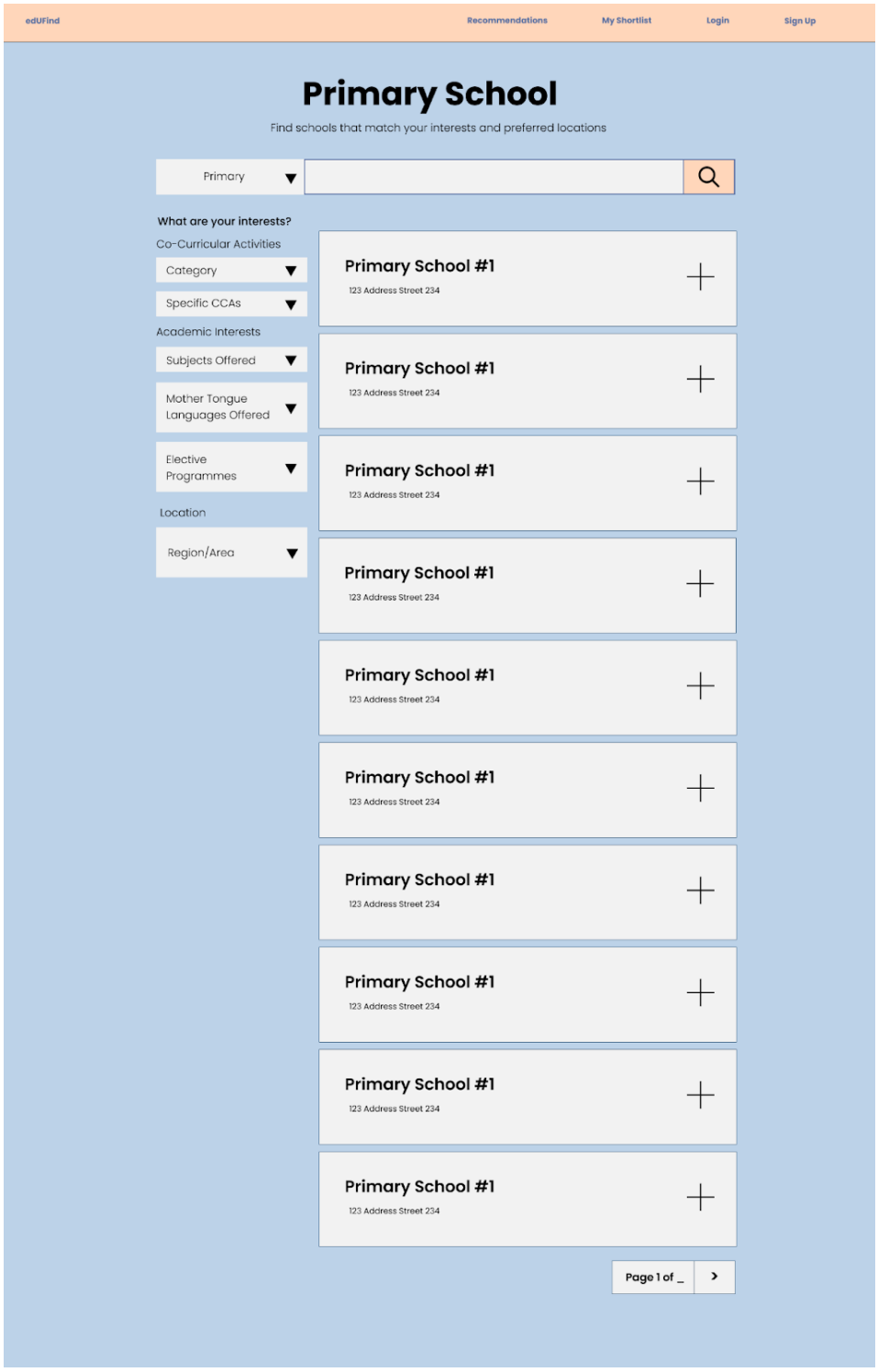




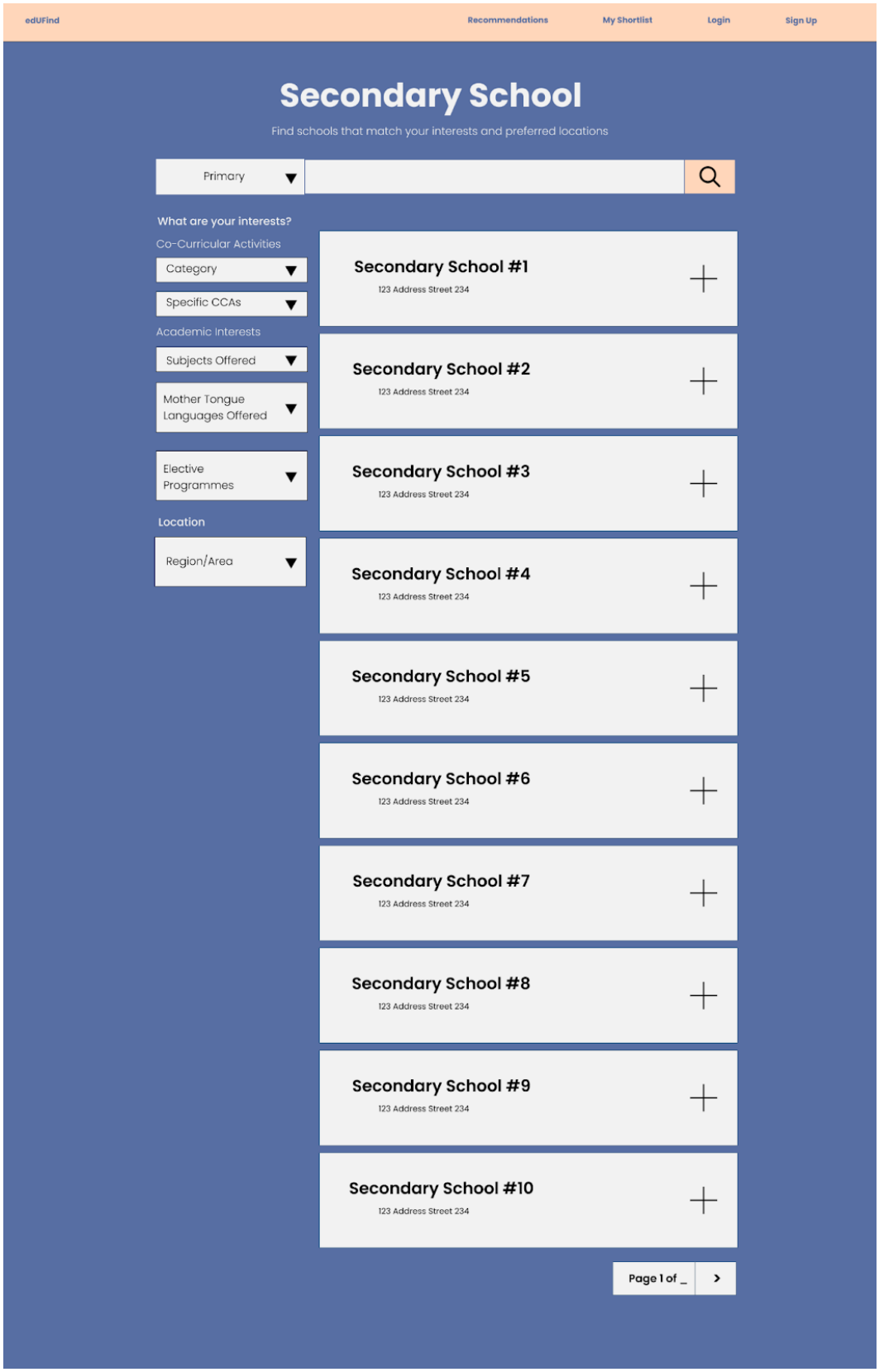
Recommendations



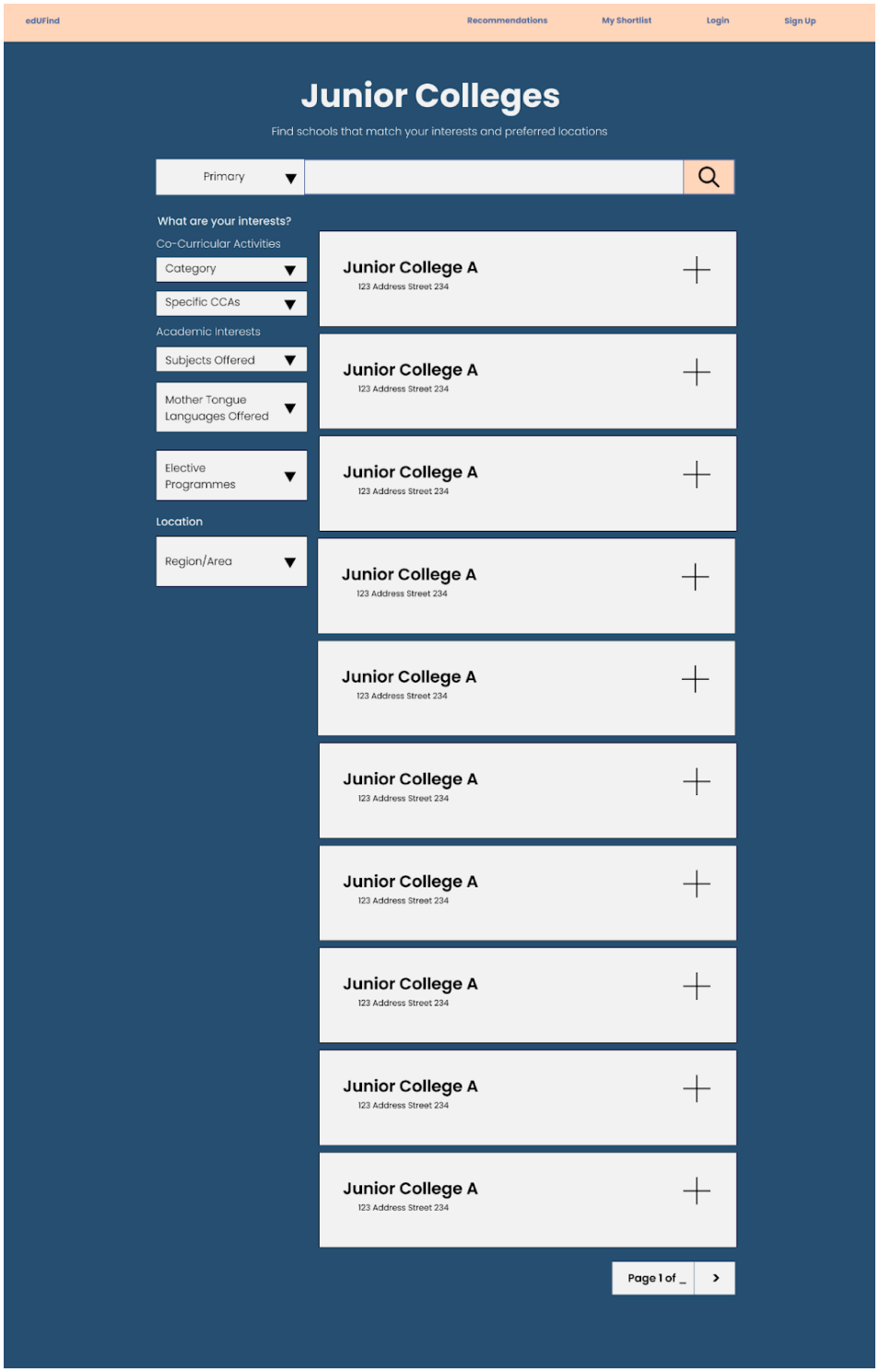
Primary Schools



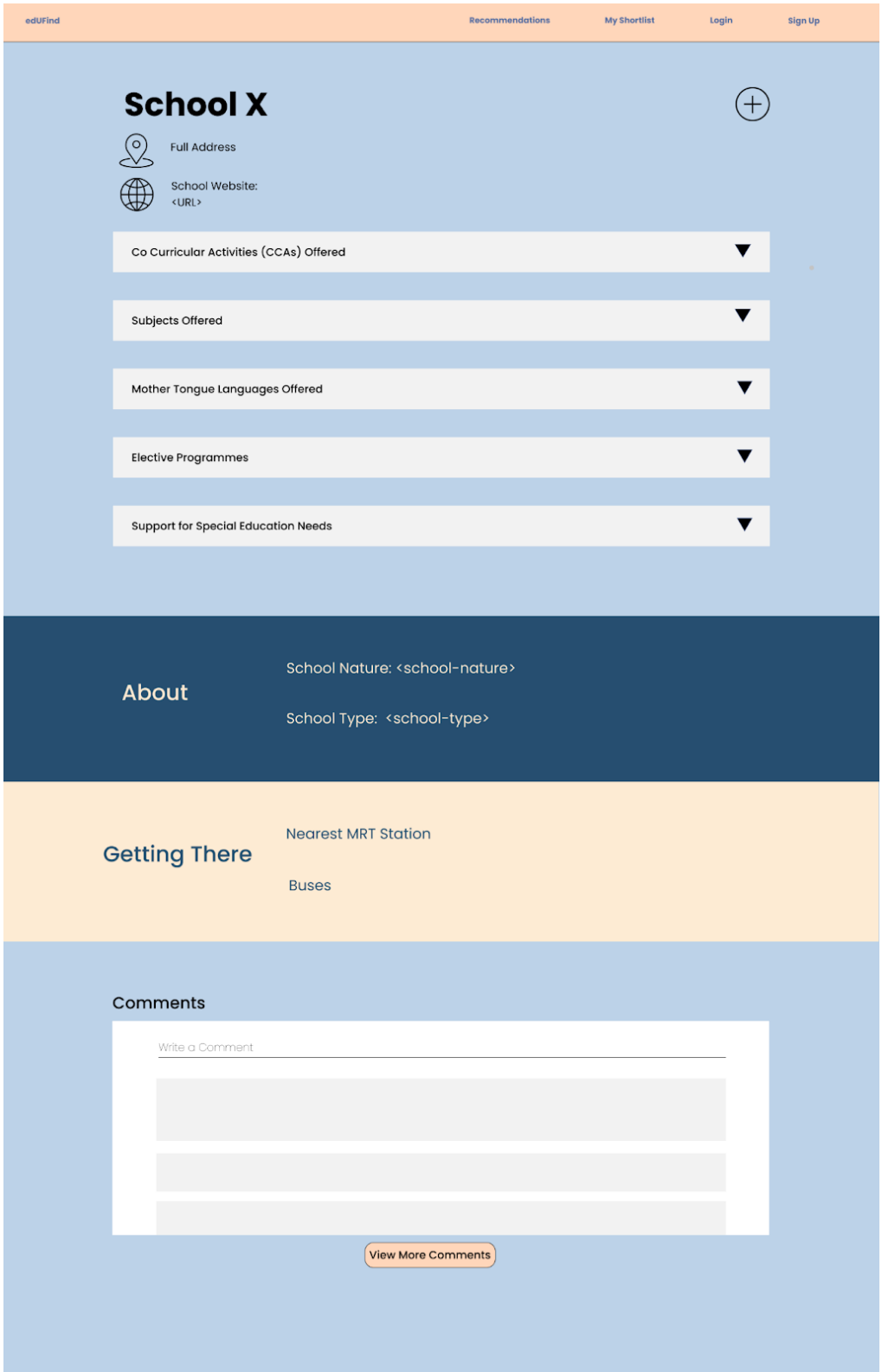
Secondary School



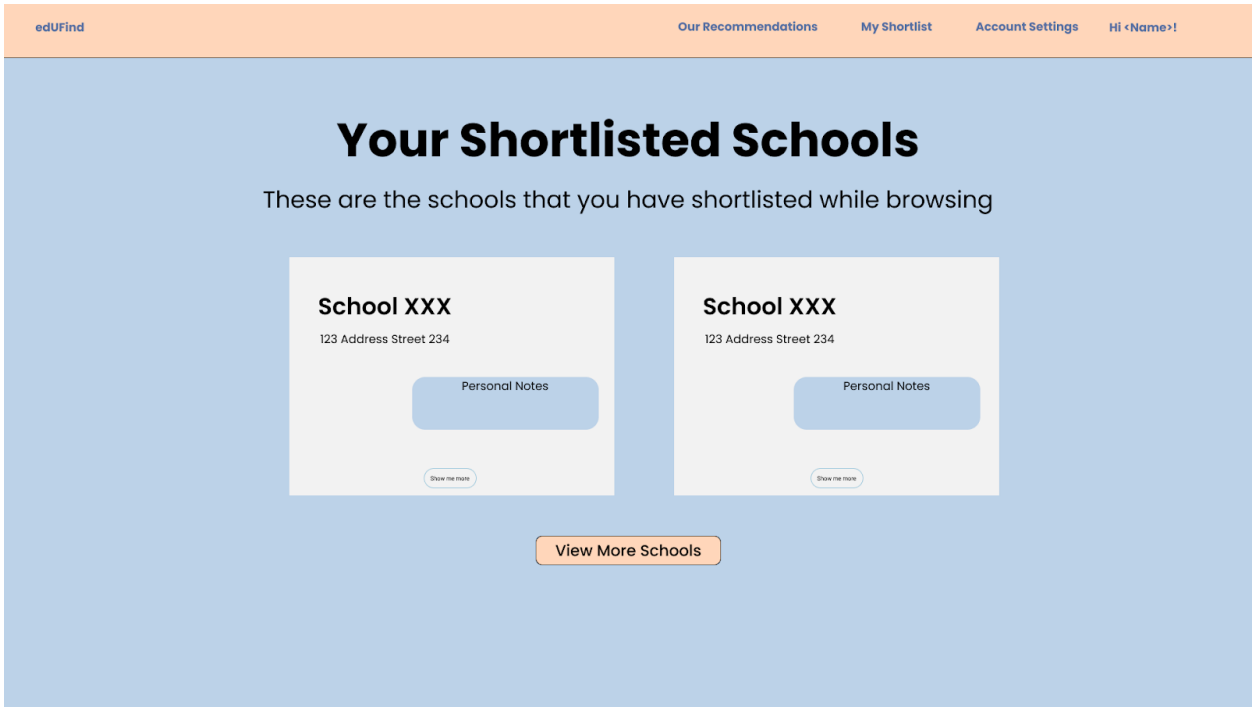
Junior College



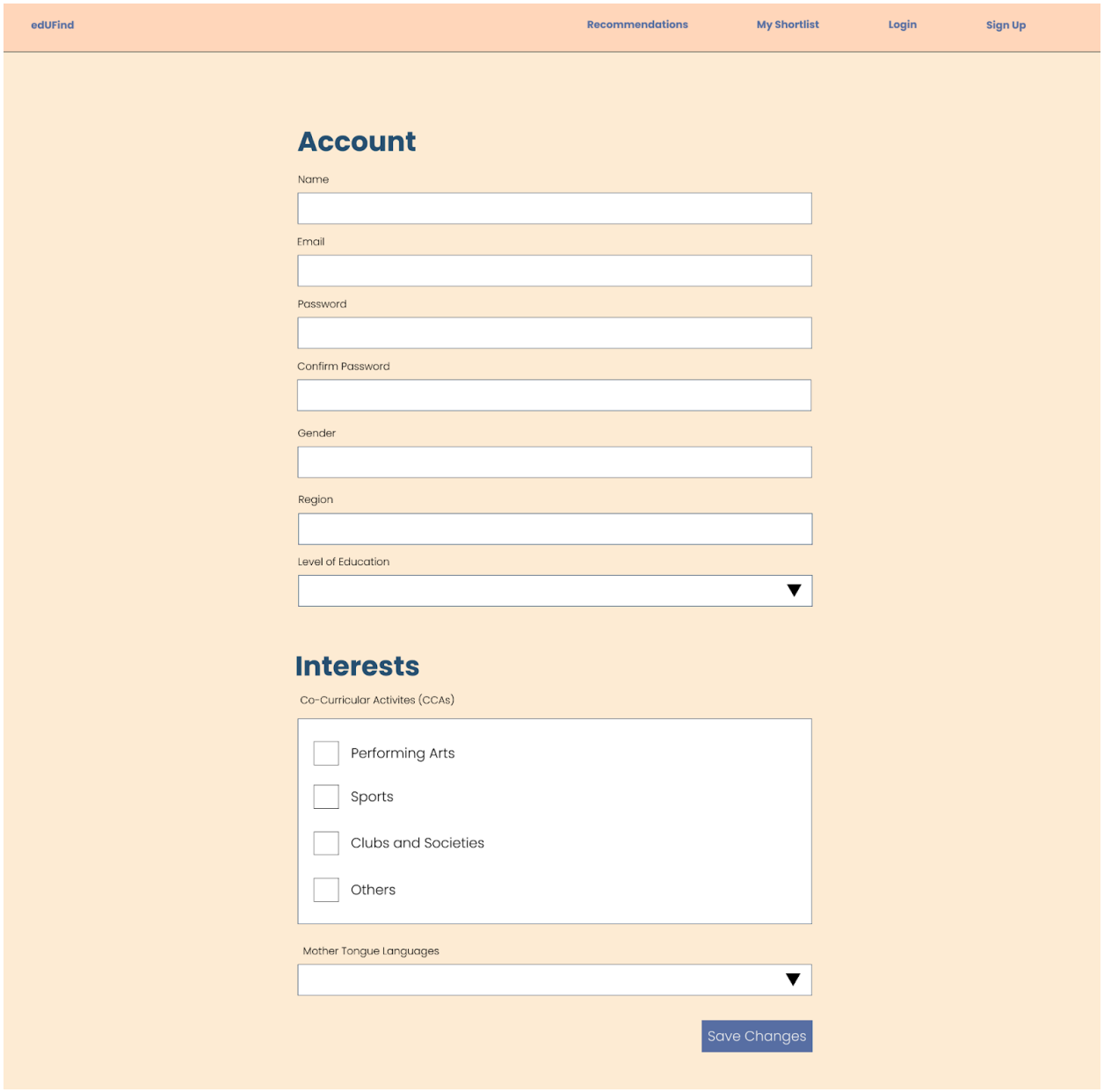
School details



Shortlisted Schools

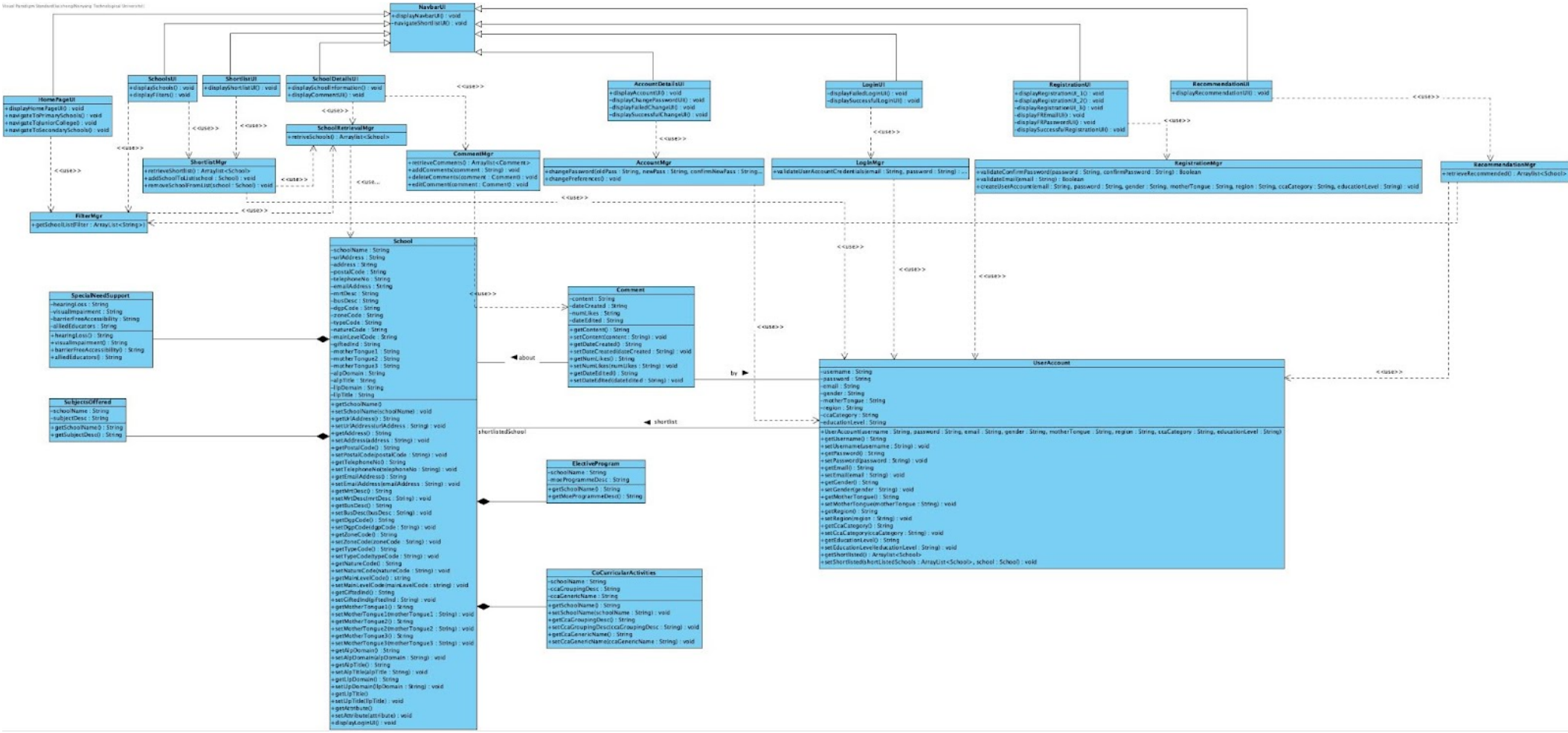


Account Settings



Class Diagram

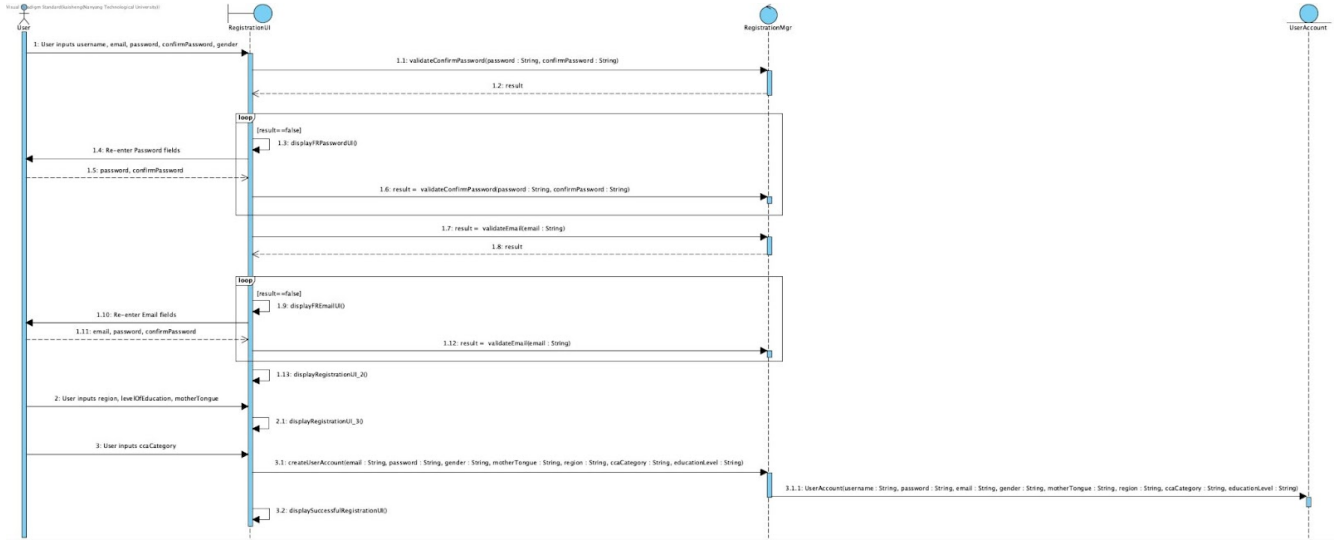
Refer to Class Diagram for the JPG File



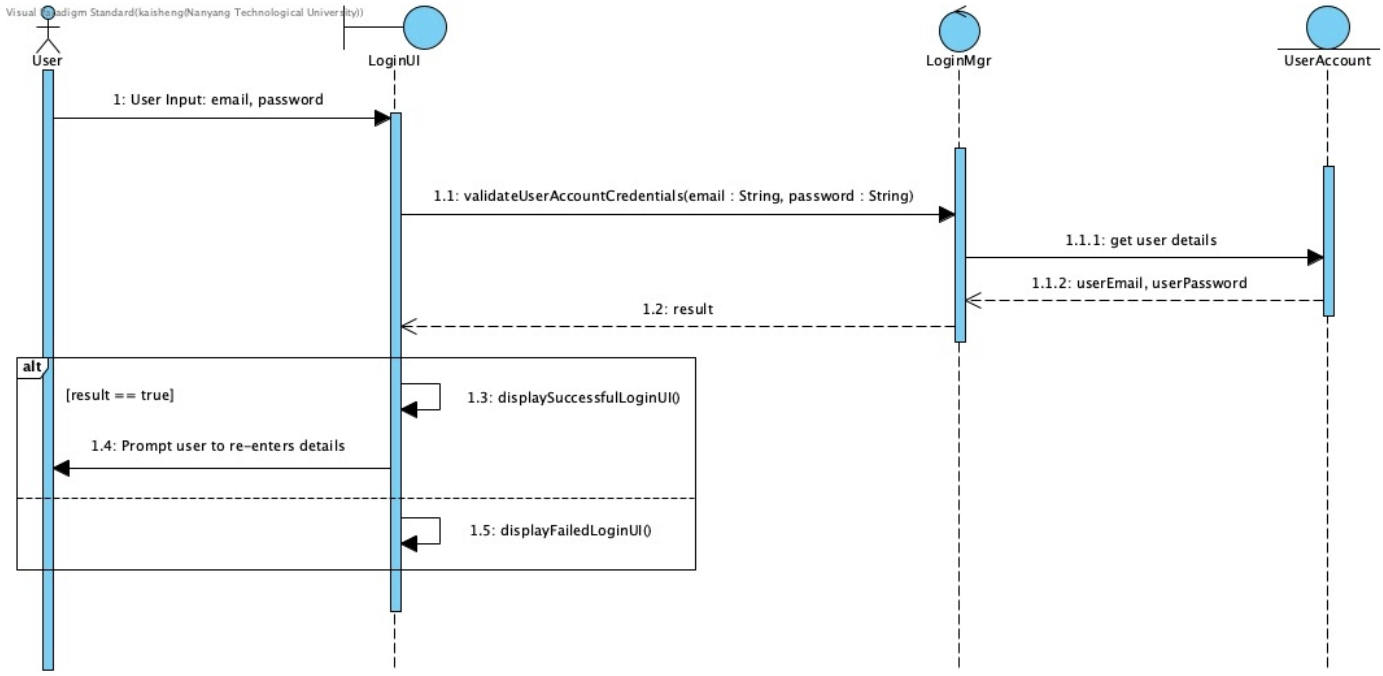
Sequence Diagram

Refer to Sequence Diagram folder for the JPG Files

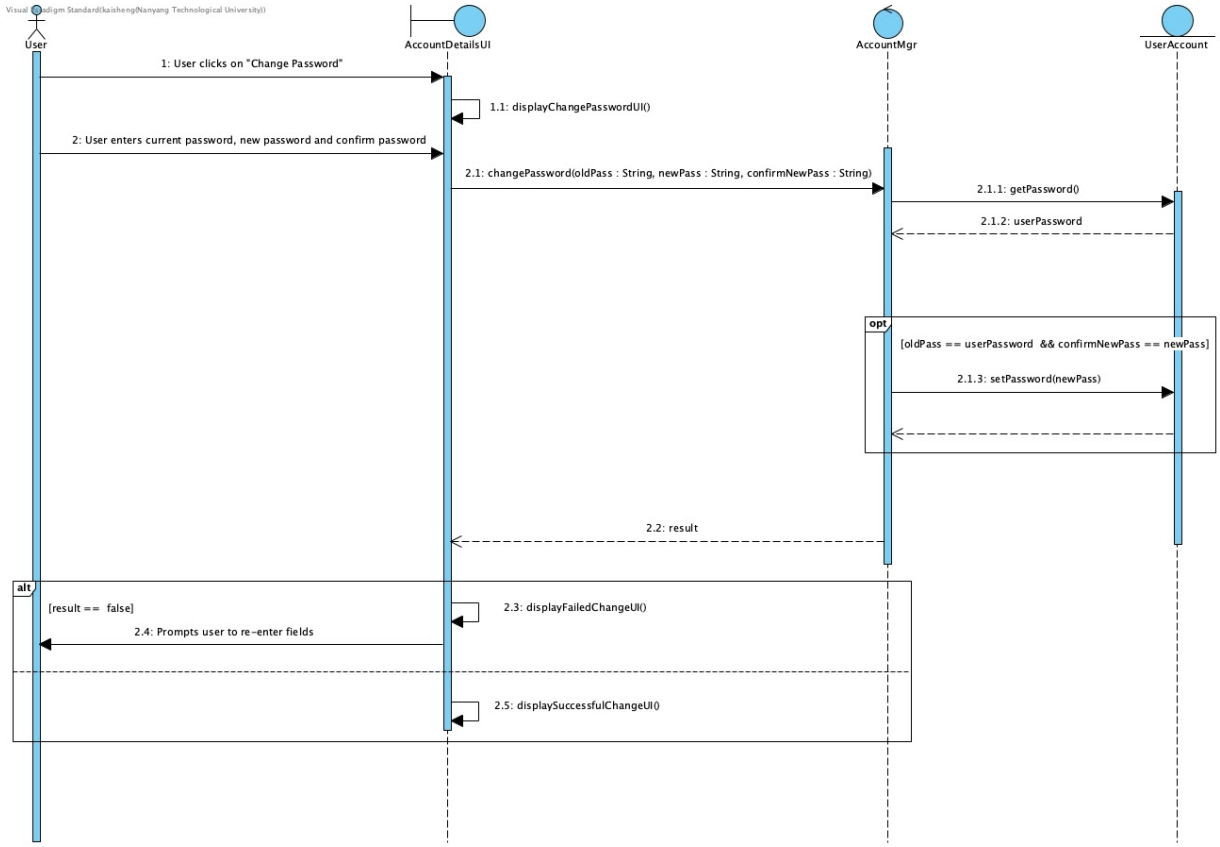
Registering User Account



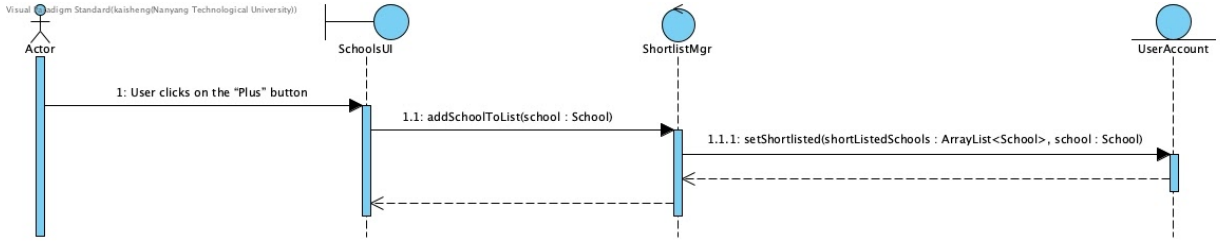
Login



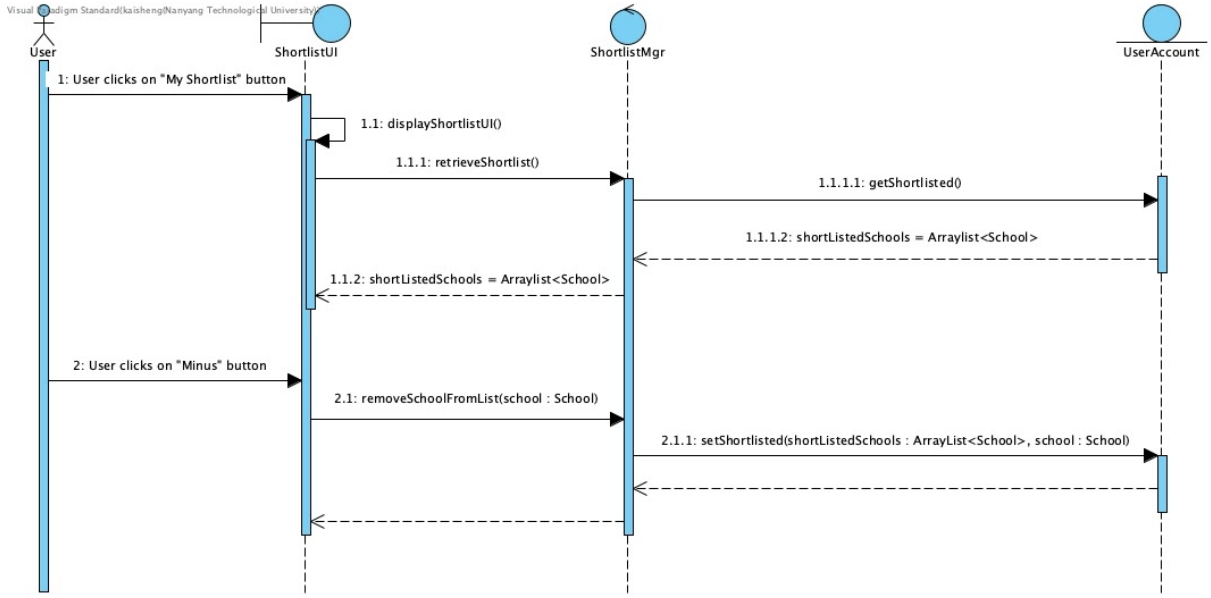
Request Password Change



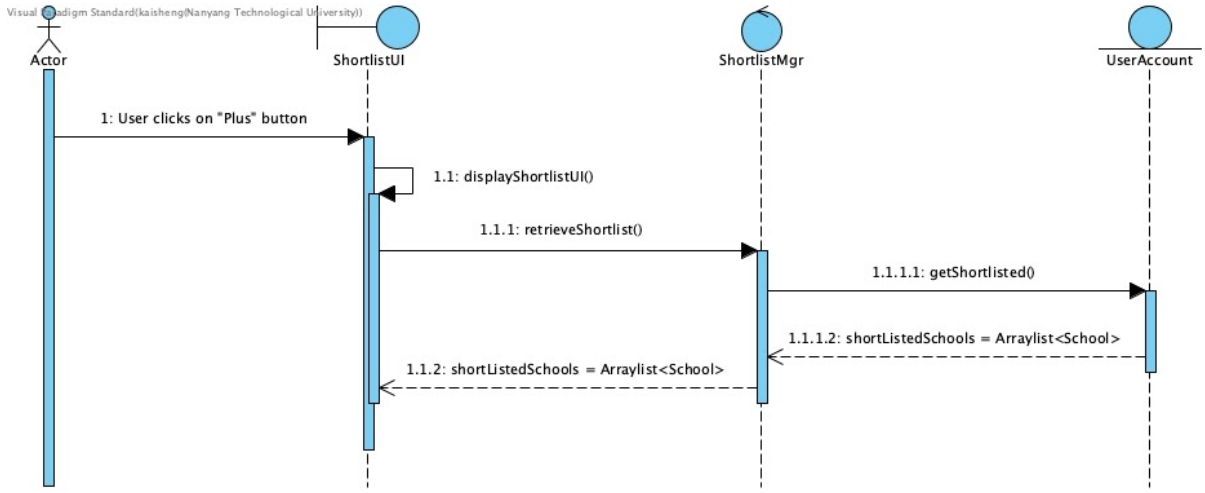
Shortlist Schools



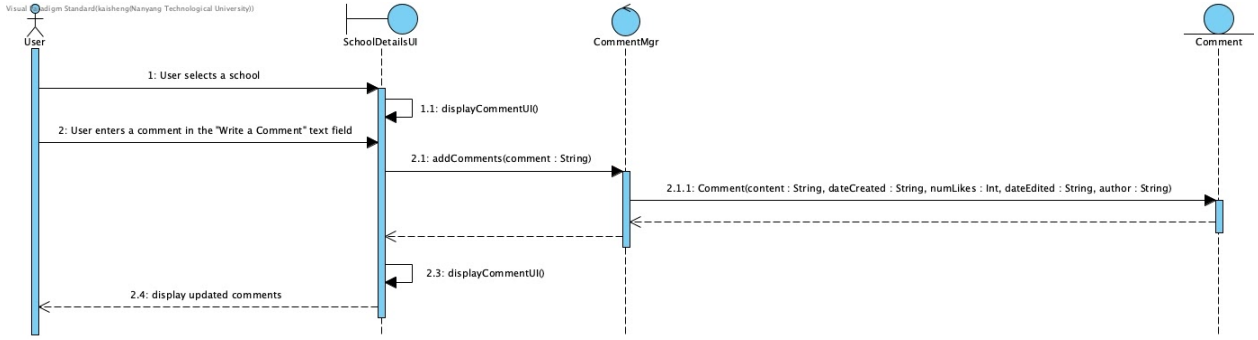
Remove Shortlisted Schools



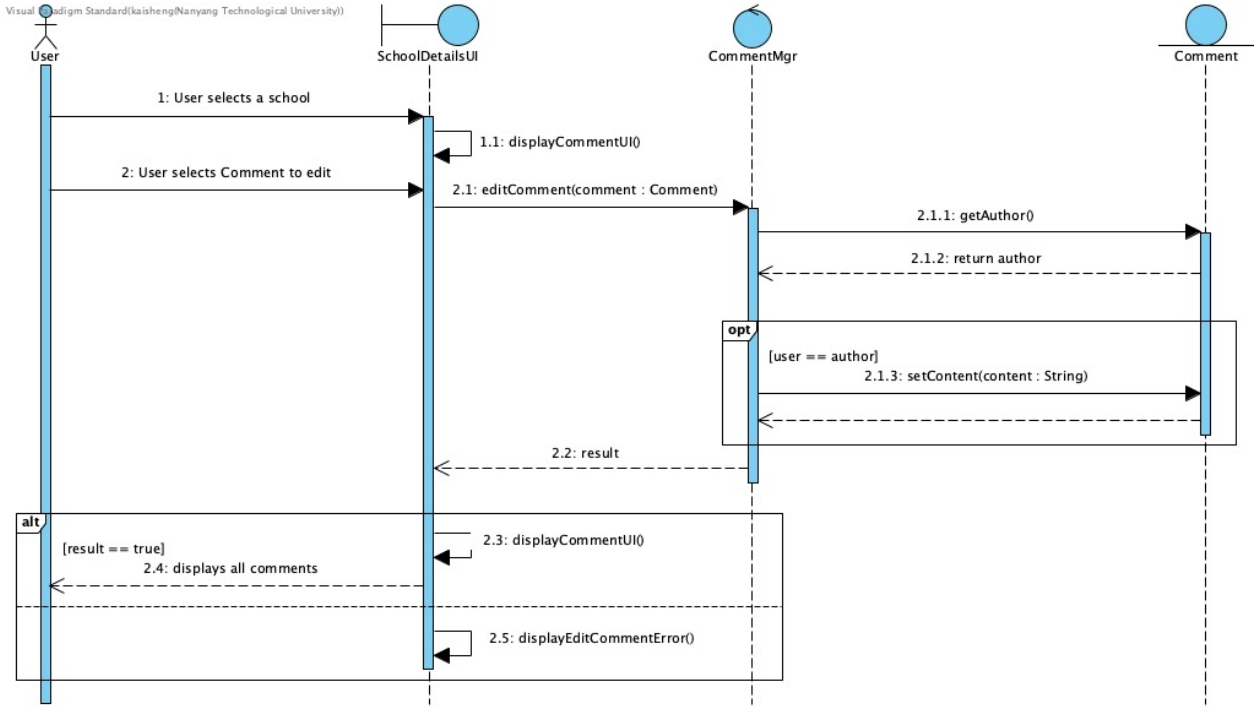
Retrieve Shortlisted Schools



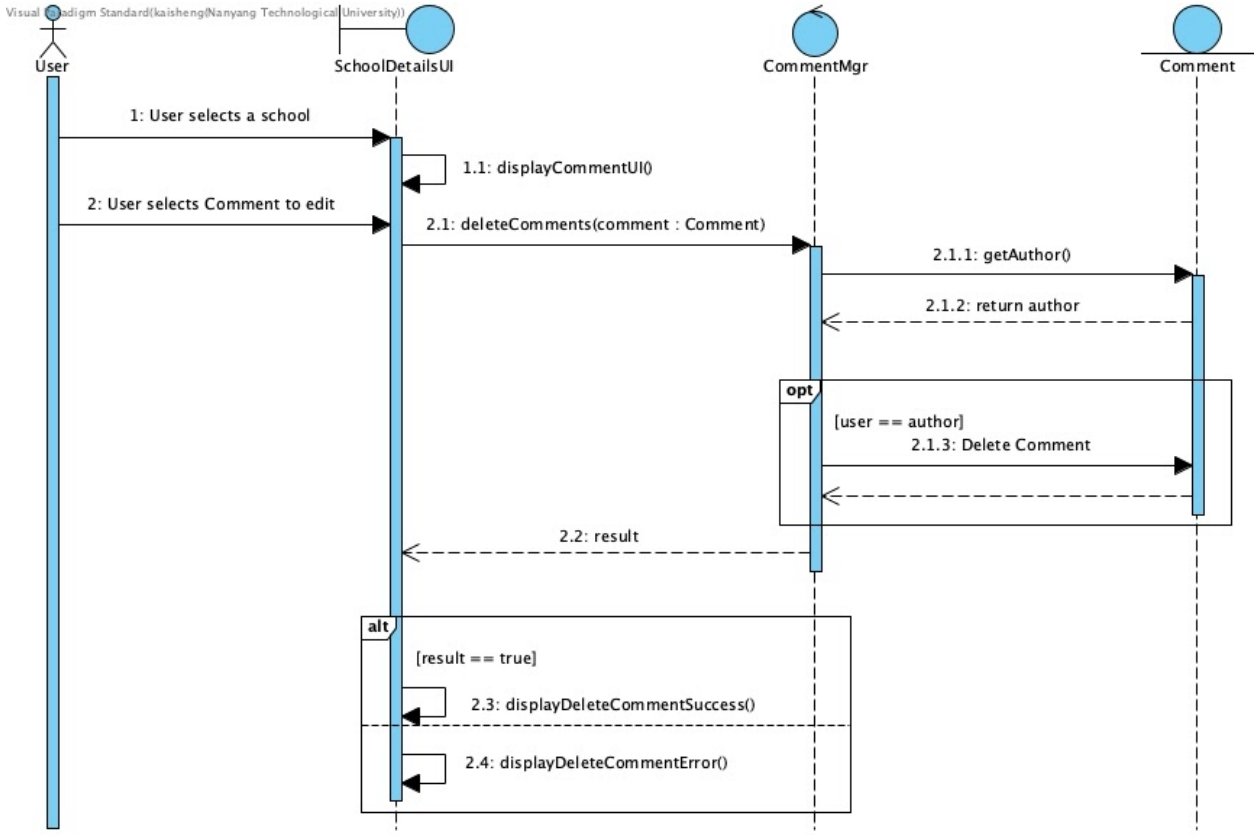
Create Comment



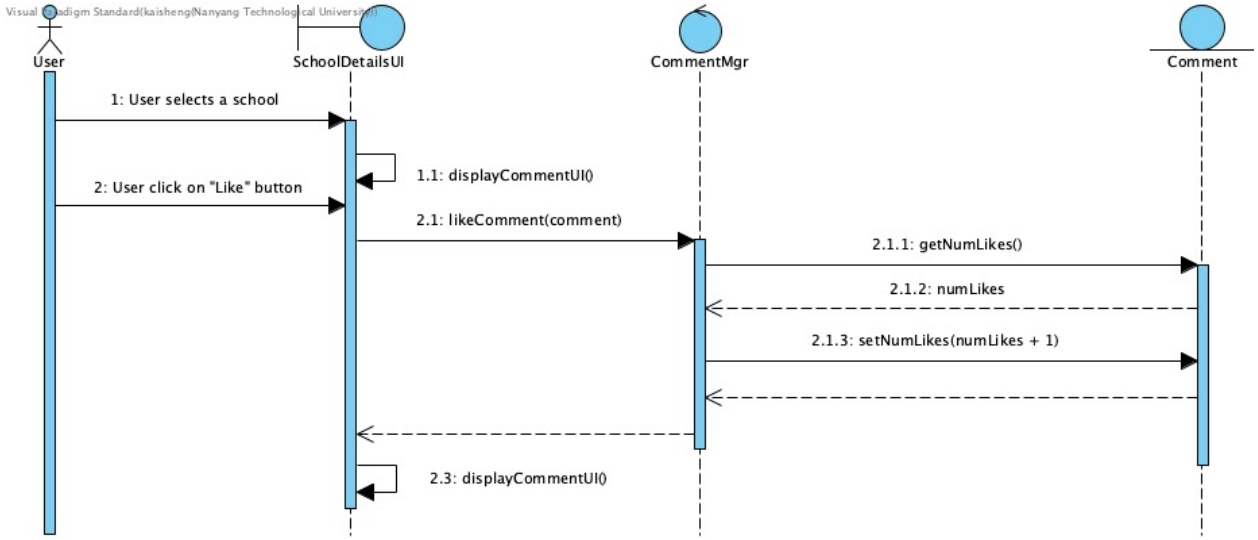
Edit Comment



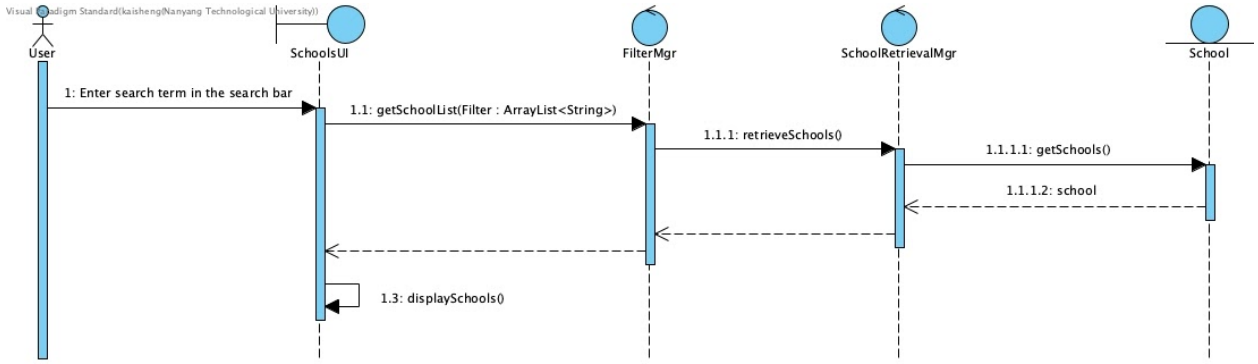
Display Comment



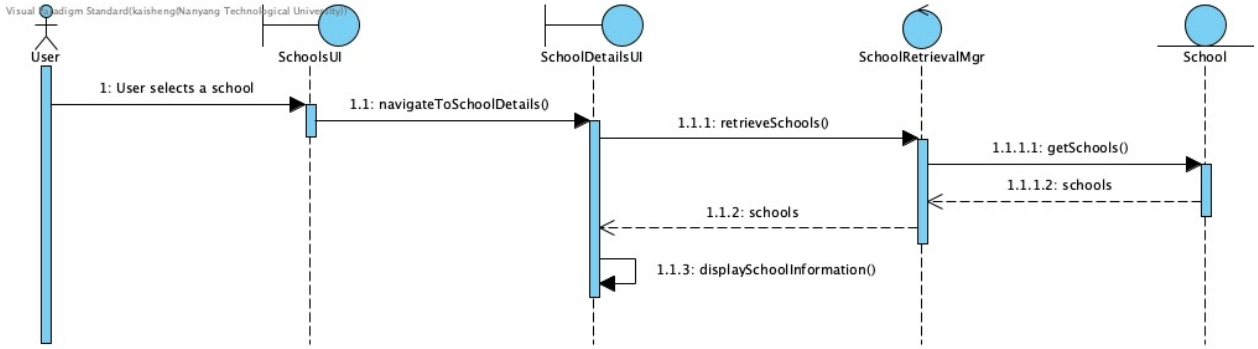
Like Comment

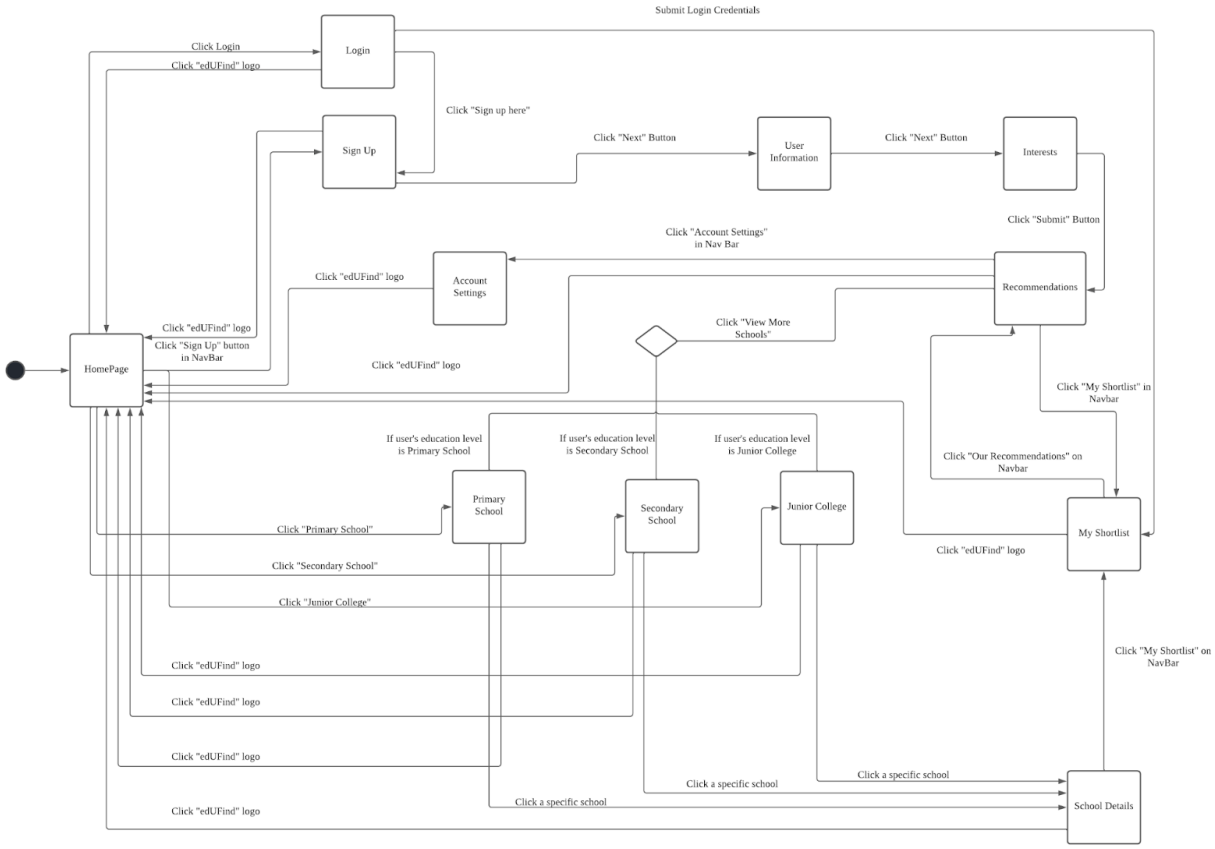


Search School



Select School



Dialog Map

Refer to Dialog Map.png for clearer image