#### Project Description

Our team will create an app that suggests things to do or places to visit in Singapore. We aim to develop a platform for people to rate & review places, and suggest places/activities they have enjoyed to their friends. Our app will also allow users to track places they have visited and wish to visit for their ease of reference. We hope to allow people to discover unique places/activities that they would otherwise have not known about, and help them decide where to go on their next outing.

#### Use Case Diagram

#### 

#### 

#### Use Case Description

| Use Case ID: | 1 | | |
| --- | --- | --- | --- |
| Use Case Name: | Verify Login Credentials | | |
| Created By: | Deionna Chee | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | System |
| --- | --- |
| Description: | Account authentication process through username and password. |
| Preconditions: | 1. User’s account must already exist in the database. 2. User’s device must be connected to WiFi/Mobile Data |
| Postconditions: | 1. The user is able to add items to their list of favourites. 2. The user is able to retrieve items from their list of favourites. 3. The user is able to remove items to their list of favourites. 4. The user is able to use the to-go list feature. 5. The user is able to add and remove items to the to-go list list. 6. The user is able to send recommendations to others (via email). |
| Priority: | High |
| Frequency of Use: | Rarely |
| Flow of Events: | 1. User tap login via username. 2. User enters their credentials in the login interface. 3. User’s select the login button. 4. System validates the account by checking the user’s credentials with the database. 5. System authenticates the user to login successfully. |
| Alternative Flows: | AF-S1: User tap login via Facebook/Google account.   1. Return to Step 4.   AF-S3: System detects empty username or password fields.   1. System displays an error message “Username or password fields cannot be empty.” 2. User filled up the required field(s) for username and password. 3. Return to Step 3.   AF-S4: User enters the wrong credentials   1. System displays an error message “Invalid username or password. Please re-enter.” 2. Return to Step 2. |
| Exceptions: | EX-AF-S1: User not logged in to their Facebook/Google account.   1. System will prompt the user to enter their credentials to login to their Facebook/Google account. |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | - Database can be referred to both System’s database and Facebook’s database. |
| Notes and Issues: | - |

| Use Case ID: | 2 | | |
| --- | --- | --- | --- |
| Use Case Name: | Register for a new account | | |
| Created By: | Deionna Chee | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | Registration for a new user account. |
| Preconditions: | 1. User’s account must not already exist in the database. 2. User’s device must be connected to WiFi/Mobile Data |
| Postconditions: | 1. A verification code will be sent to the user’s registered email address. |
| Priority: | Medium |
| Frequency of Use: | Rarely |
| Flow of Events: | 1. User enters a valid username, email address, first name, last name, password and confirms password fields. 2. User selects the register button. 3. System validates the password and confirms password fields are identical. 4. System will send verification code to the registered email address. |
| Alternative Flows: | AF-S1: System detects an invalid (existing) username.   1. System displays an error message “Username has been taken. Try another one.” 2. Return to Step 1.   AF-S1: System detects an existing email.   1. System displays error message “Email has been taken. Try another email or try logging in.” 2. Return to step 1 or redirect the user to login.   AF-S3: System detects mismatch between the password and confirm password.   1. System displays error message “Password and Confirm Password mismatch. Please re-enter.” 2. Return to Step 1. |
| Exceptions: | - |
| Includes: | 1. Validate the account availability. |
| Special Requirements: | - |
| Assumptions: | Database can be referred to both System’s database and Facebook’s database. |
| Notes and Issues: | - |

| Use Case ID: | 3 | | |
| --- | --- | --- | --- |
| Use Case Name: | Register via Facebook/Google account | | |
| Created By: | Deionna Chee | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | Registration for a new user account via the user's Facebook/Google account. |
| Preconditions: | 1. User’s account must not already exist in the database. 2. User’s device must be connected to WiFi/Mobile Data |
| Postconditions: | - |
| Priority: | Medium |
| Frequency of Use: | Rarely |
| Flow of Events: | 1. User selects “Login via Facebook”/”Login via Google” button. 2. User is already logged in to their Facebook/Google account. 3. System will validate the account availability with respect to the database. 4. User will be prompted to grant permission for System to access User’s Facebook/Google account. 5. User will grant the permission. 6. System will extract User data from User’s Facebook/Google account and save them in the database. 7. User will be able to login successfully using their Facebook/Google account. |
| Alternative Flows: | AF-S2: System detects user’s mobile is not logged in to the Facebook/Google account.   1. System will prompt User to enter their credentials to login to their Facebook/Google account. 2. User provide their credentials to login. 3. Return to Step 3. |
| Exceptions: | - |
| Includes: | 1. Validate the account availability. |
| Special Requirements: | - |
| Assumptions: | - Database can be referred to both System’s database and Facebook’s database. |
| Notes and Issues: | - |

| Use Case ID: | 4 | | |
| --- | --- | --- | --- |
| Use Case Name: | Validate account availability | | |
| Created By: | Lara Hor | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | System |
| --- | --- |
| Description: | System validates account availability by receiving the correct verification code, and creates the account in the database. |
| Preconditions: | 1. User’s device must be connected to Wifi/Mobile data. 2. Account cannot already exist in the database. |
| Postconditions: | 1. Account is successfully created in the system’s database. 2. User is able to login to the system using the created account. |
| Priority: | Medium |
| Frequency of Use: | Rarely |
| Flow of Events: | 1. User enters the correct verification code. 2. System checks that the verification code is accurate. 3. System creates the account in the database. |
| Alternative Flows: | AF-S1: System detects the wrong verification code.   1. System displays an error message “Wrong verification code.” 2. System prompts the user to select the option to send a new verification code. 3. User re-enters the new verification code. 4. Return to Step 1. |
| Exceptions: | EX1: Code not received by the user.   1. System has a “Resend Code” button that allows the user to click for the verification code to be sent again. |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: |  |

| Use Case ID: | 5 | | |
| --- | --- | --- | --- |
| Use Case Name: | Request for password change | | |
| Created By: | Ihsan | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | Allow User to reset or change their password |
| Preconditions: | 1. User’s device must be connected to WiFi/Mobile Data |
| Postconditions: | 1. User’s password has to be updated in the database. 2. User must be able to login with new password |
| Priority: | Medium |
| Frequency of Use: | Rarely |
| Flow of Events: | 1. User enters registered email and submits request 2. System checks if there exists an account with that email on the database 3. System sends OTP code to User’s email address 4. User enter OTP code and submits 5. System validates OTP code 6. User enters new password, enters it again to reconfirm and submits 7. System check if both typed passwords match 8. System saves users new password in the databse |
| Alternative Flows: | AF-21: Email address not in database   1. System displays “Account with email doesn’t exist” 2. Return to Step 1.   AF-S5: System detects an existing email.   1. System displays “OTP code is invalid” 2. Return to step 4   AF-S7: System detects mismatch between the password and confirm password.   1. System displays error message “Password and Confirm Password mismatch. Please re-enter.” 2. Return to Step 6. |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 6 | | |
| --- | --- | --- | --- |
| Use Case Name: | Scrolling recommended locations and retrieving location details | | |
| Created By: | Lara Hor | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | Scroll through the list of recommended locations and retrieve information about the selected location. |
| Preconditions: | 1. User’s device must be connected to Wifi/Mobile data. 2. User must be logged into the application for locations based on the user's interests. |
| Postconditions: | 1. Details of selected location will be displayed in the system |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. Pictures of recommended locations will be displayed on the home page. 2. User scrolls through the pictures to view the various recommended places. 3. User selects a location from the list of pictures shown to find out more information about the location. 4. System displays information about the selected location. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | Retrieve location information |
| Special Requirements: | Requires user to be logged in to show extra locations based on the user's indicated interests. |
| Assumptions: | - |
| Notes and Issues: | If user is not registered and logged in, system will only display current popular places that other users have favourited. |

| Use Case ID: | 7 | | |
| --- | --- | --- | --- |
| Use Case Name: | Search | | |
| Created By: | Ihsan | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | Search for places of interest/Activity spots using filters |
| Preconditions: | 1. User’s device must be connected to Wifi/Mobile data. |
| Postconditions: | Based on user’s search conditions and specified filters, a list of places will be displayed |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. User enters text to be queried into the search bar 2. System autocompletes/makes suggestions to users text query 3. User can tap submit without using filters 4. System displays list of places based on User’s input and specified filters/sort 5. By default, results are sorted by prominence if no filters/sort are selected |
| Alternative Flows: | AF-S3: User modified the search using filters   1. Available filters: Price, Distance radius, Ratings, Open or not 2. System will limit apply constraints with text entered by user 3. Return to step 4   AF-S3: User modified the search using “sort by”   1. Available sorting conditions: A-Z, Z-A, Low - High Price, Closest - Furthest Distance radius, High - Low Ratings 2. System will apply sorting algorithm to results 3. Return to step 4   AF-S3: User taps suggestions instead of submit   1. User is directed straight to the selected suggestion |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 8 | | |
| --- | --- | --- | --- |
| Use Case Name: | Add location to their favourites | | |
| Created By: | Deionna Chee | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | To add location to their list of favourite locations. |
| Preconditions: | 1. User’s device must be connected to WiFi/Mobile Data. 2. User must be logged in to their account. |
| Postconditions: | 1. Added location will be saved into the database. |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. User navigates to the location details page. 2. User clicks the favourite button to add location to their favourites. 3. System will save the user’s favourite location into the database. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Verify Login Credentials 2. Retrieve location information |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 9 | | |
| --- | --- | --- | --- |
| Use Case Name: | Add interest to their favourites | | |
| Created By: | Deionna Chee | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | To add location type to their list of interests. |
| Preconditions: | 1. User’s device must be connected to WiFi/Mobile Data. 2. User must be logged in to their account. |
| Postconditions: | 1. Added interest will be saved into the database. |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. User navigates to the location details page. 2. After user clicks the favourite button to add location to their favourites, the system prompts user to add the location’s type(s) to their list of interest. 3. User selects the location type(s) and clicks ‘Add’. 4. System will save the user’s interest(s) into the database. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Verify Login Credentials |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 10 | | |
| --- | --- | --- | --- |
| Use Case Name: | Retrieve their list of favourite items | | |
| Created By: | Deionna Chee | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | To retrieve their list of favourite items. |
| Preconditions: | 1. User’s device must be connected to WiFi/Mobile Data. 2. User must be logged in to their account. |
| Postconditions: | 1. A list of user’s favourites will be displayed on the Favourites page. |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. User navigates to the Favourites page. 2. User clicks on the category of favourites to view (location or interest). 3. System will retrieve the list of user’s favourites from the database based on the category selected. 4. System will display a list of user’s favourites on the Favourites page. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Verify Login Credentials |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 11 | | |
| --- | --- | --- | --- |
| Use Case Name: | Remove their favourite items | | |
| Created By: | Deionna Chee | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | To remove their favourite items. |
| Preconditions: | 1. User’s device must be connected to WiFi/Mobile Data. 2. User must be logged in to their account. |
| Postconditions: | 1. User’s selected item will be removed from the list of favourite items. |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. User navigate to Favourites page. 2. User clicks on the category of favourites to remove item from (location or interest). 3. User selects the item to be removed. 4. System will remove the selected item from the list of user’s favourite items and update the database. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Verify Login Credentials |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 12 | | |
| --- | --- | --- | --- |
| Use Case Name: | Adding to the user's to-go list. | | |
| Created By: | Magdeline | Last Updated By: | Deionna |
| Date Created: |  | Date Last Updated: | 20/2/2022 |

| Actor: | User |
| --- | --- |
| Description: | Add a place to the user’s to-go list. |
| Preconditions: | 1. 1.User must be logged into the application. 2. User’s device must be connected to Wifi/Mobile data. |
| Postconditions: | The changes the user made will be reflected in the to-go list. |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. User clicks the calendar logo. 2. System will prompt the user to enter the date. 3. User clicks the clock logo. 4. System will prompt the user to enter the time. 5. User clicks on ‘add to to-explore’. 6. System will add and save the item to the user’s to-explore list. |
| Alternative Flows: | AF-S5: User clicks on ‘invite someone’   1. System retrieves the list of contacts from the user account and prompts the user to select user(s) to invite. 2. System sends notification to the invitee(s) and adds the item to the invitee(s)’ inbox. 3. Return to step 6 |
| Exceptions: | - |
| Includes: | 1. Verify Login Credentials |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 13 | | |
| --- | --- | --- | --- |
| Use Case Name: | Respond to invitations | | |
| Created By: | Magdeline | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | Accept or reject invitations from other users and reflect in the explore list. |
| Preconditions: | 1. User’s device must be connected to Wifi/Mobile data. |
| Postconditions: | Invitation details are reflected in the user's explore list should the user accept the invitation. |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. User receives in-app notifications of invitation from other users. 2. User can click the “accept” or “reject” button. 3. If “accept” button is clicked, the system will add the respective location to the explore list of the invitee. 4. The system will send a notification to inform the inviter. 5. If “reject” button is clicked, the system removes the invitation from the explore invitation tab of the user. 6. The system will send a notification to inform the inviter. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 14 | | |
| --- | --- | --- | --- |
| Use Case Name: | Send personalized recommended location email notification | | |
| Created By: | Herman Foo | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | System |
| --- | --- |
| Description: | Sending personalized recommended location email notification |
| Preconditions: | 1. User’s email must be verified |
| Postconditions: | - |
| Priority: | High |
| Frequency of Use: | Once daily or every two days |
| Flow of Events: | 1. System will collate 3 recommended places for each user 2. System will group users batch by batch based on UID 3. A cron job will dispatch the batches to receive email notification of their 3 recommended places at a random time between 0900hrs to 2359hrs |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 15 | | |
| --- | --- | --- | --- |
| Use Case Name: | Retrieve location information | | |
| Created By: | Herman Foo | Last Updated By: |  |
| Date Created: |  | Date Last Updated: |  |

| Actor: | User |
| --- | --- |
| Description: | Retrieving location information |
| Preconditions: | 1. User’s device must be connected to Wifi/Mobile data 2. User must be logged in to their account |
| Postconditions: | 1. User will be able to view the location information |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. System will display a list of location based on the user’s filters, type or search keywords 2. User will select or search the location of interest 3. System will display the selected location’s information |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

| Use Case ID: | 16 (from Use case 12 previously) | | |
| --- | --- | --- | --- |
| Use Case Name: | Removing from the user's to-go list. | | |
| Created By: | Magdeline | Last Updated By: | Deionna |
| Date Created: |  | Date Last Updated: | 20/2/2022 |

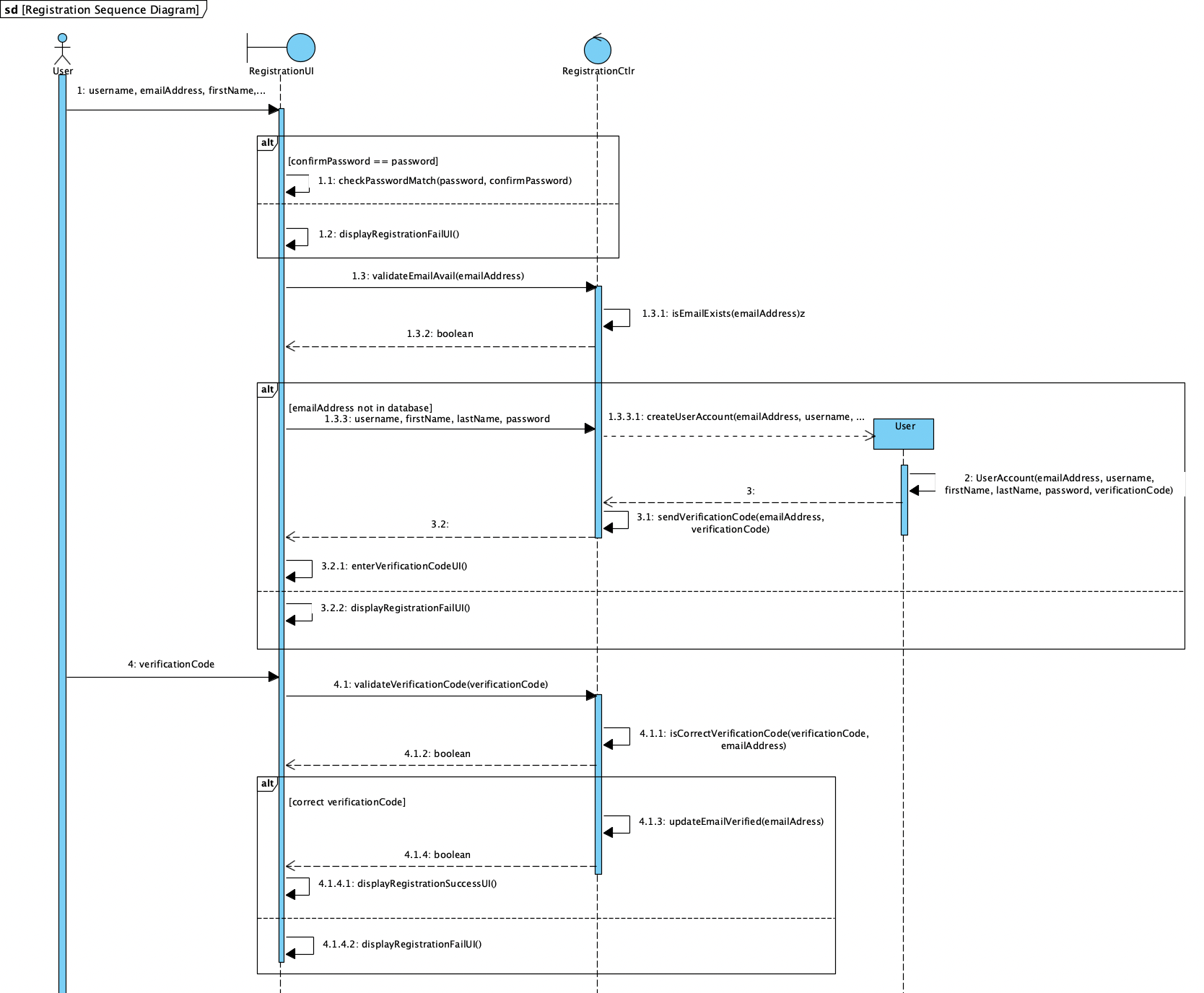
| Actor: | User |
| --- | --- |
| Description: | Remove a place from the user’s to-go list. |
| Preconditions: | 1. User must be logged into the application. 2. User’s device must be connected to Wifi/Mobile data. |
| Postconditions: | The changes the user made will be reflected in the to-go list. |
| Priority: | High |
| Frequency of Use: | Often |
| Flow of Events: | 1. In the to-explore page, the list of to-explore items are displayed. 2. User clicks on the ‘my status’ dropdown menu of the item the user wishes to remove. 3. User selects ‘unexplored’. 4. System removes the to-explore item from the user’s to-explore list. |
| Alternative Flows: |  |
| Exceptions: | - |
| Includes: | 1. Verify Login Credentials |
| Special Requirements: | - |
| Assumptions: | - |
| Notes and Issues: | - |

#### Class Diagram

#### 

#### Sequence Diagrams

##### Use case 2: Register for a new account



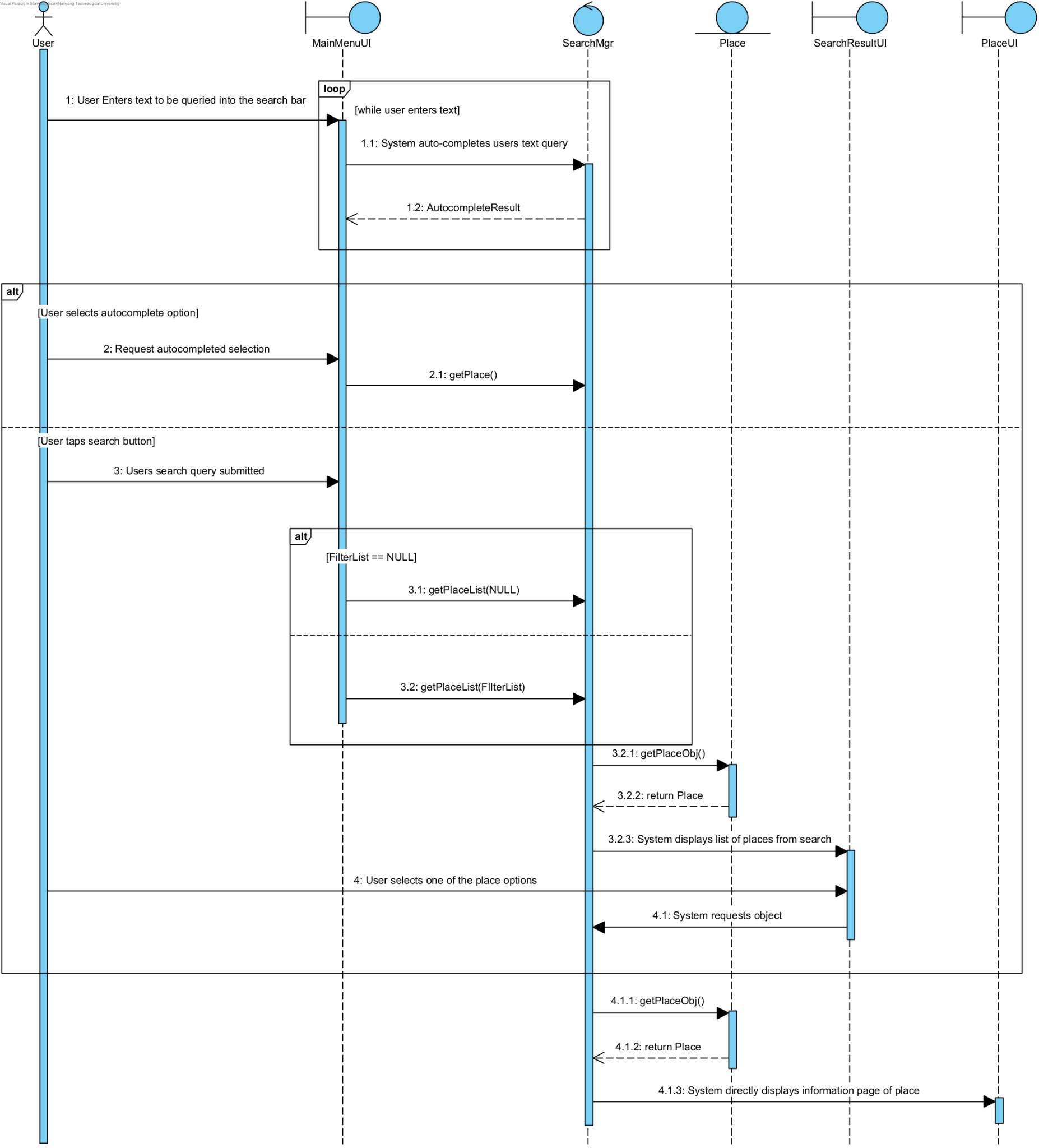
##### 

##### 

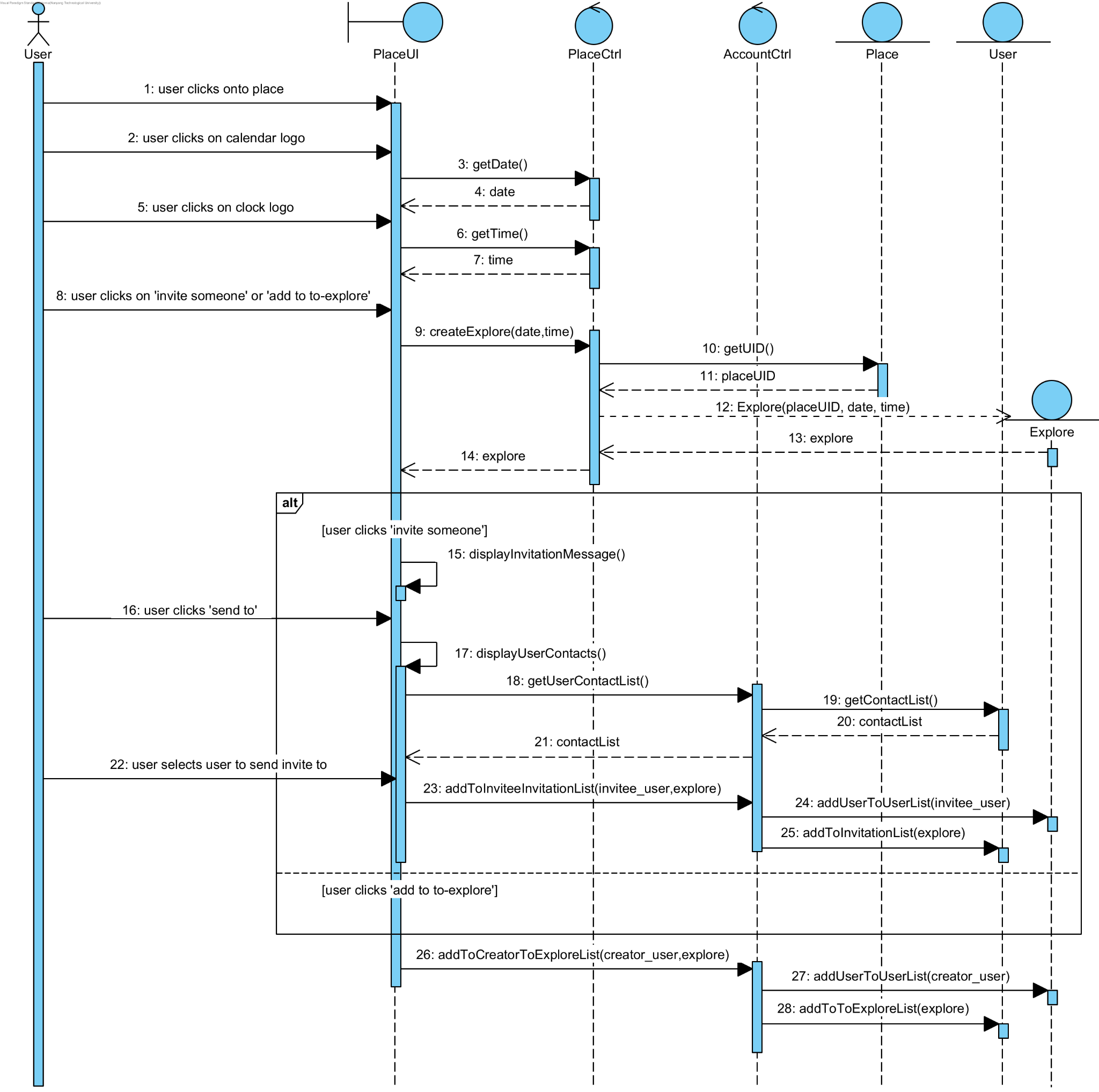
##### 

##### 

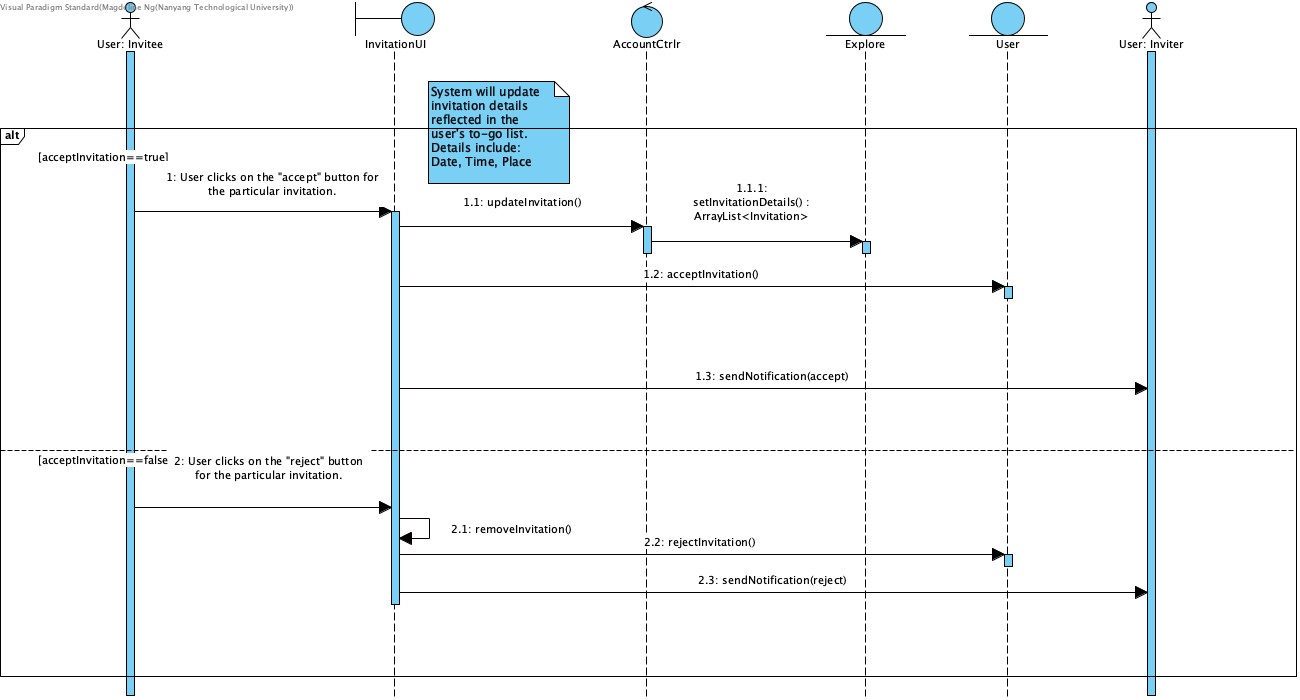
##### Use case 7: Search



##### Use case 12: Updating the user's to-go list.



##### Use case 13: Respond to invitations



#### State Machine Diagram (Dialog map)

