

**CZ2006 Software Engineering**

**Lab 2 Deliverables**

**Lab Group SS6**

**Group ifandonlyif**

**Chua Kok Liang**

**Grace Mok Jie Qi**

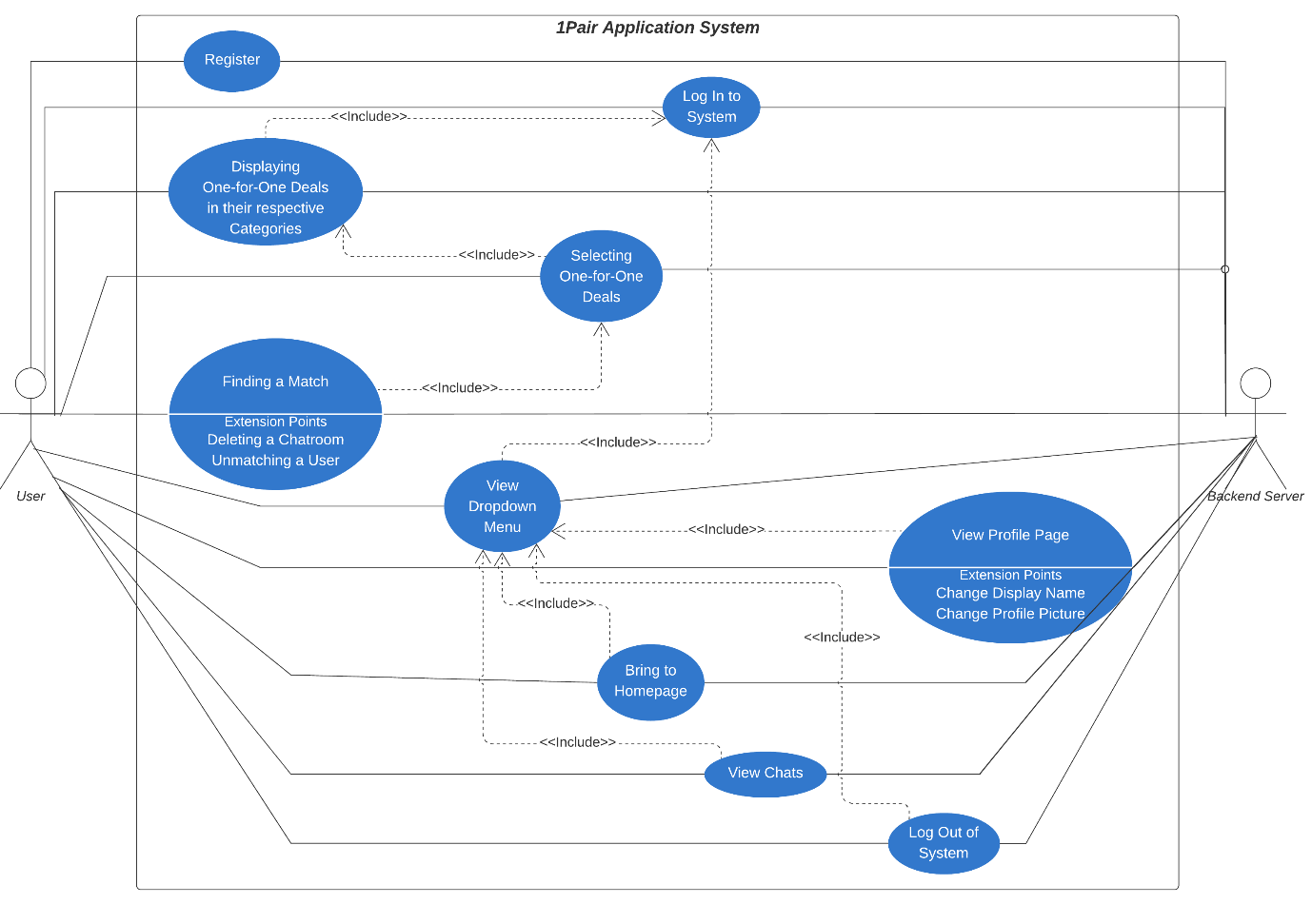
**Jozua Heng Yi Jie**

**Nicholas Yeo Ming Jie**

**Royce Ang Jia Jie**

**Shahrin Chua Zong Da**

**Use Case Diagram**

****

**Use Case Descriptions**

Use Case 1: Register

Use Case 2: Log In to System

Use Case 3: Displaying One-for-One Deals in their respective Categories

Use Case 4: Selecting One-for-One Deals

Use Case 5: Finding a Match

Use Case 6: Deleting a Chatroom

Use Case 7: Unmatching a User

Use Case 8: View Dropdown Menu

Use Case 9: View Profile Page

Use Case 10: Change Display Name

Use Case 11: Change Profile Picture

Use Case 12: Bring to Homepage

Use Case 13: View Chats

User Case 14: Log Out of System

Use Case 1: Register

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | User registers as a new user in the system. |
| Entry-Conditions: | 1. User has the system open and running. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. System displays the Homepage. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. System displays the Landing Page. 2. User clicks the “Register” button. 3. User enters their preferred Display Name. 4. User enters their Email. 5. User enters their preferred Password which is at least 6 characters long. 6. System will display a popup with the text “Registering User, This may take a while”. 7. The user’s account information is sent to the backend server and saved there. 8. The user is automatically logged into the system. 9. System displays the Homepage. |
| Alternative Flows: | AF-S3: If the entered Email already exists in the server   1. The system will display a temporary message that says “Invalid Email/Password”.   AF-S4: If the Password is not at least 6 characters long   1. The system will display a temporary message that says “Invalid Email/Password”. |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 2: Log In to System

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | User Logs In to the System |
| Entry-Conditions: | 1. User has the system open and running. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. System displays the Homepage. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. System displays the Landing Page. 2. User clicks the “Log In” button. 3. User enters their Email. 4. User enters their Password. 5. System will display a popup with the text “Logging In User, This may take a while”. 6. System displays the Homepage. |
| Alternative Flows: | AF-S2: If the entered Email does not exist in the server   1. The system will display a temporary message that says “Invalid Email/Password”.   AF-S3: If the Password associated with the Email is invalid   1. The system will display a temporary message that says “Invalid Email/Password”. |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 3: Displaying One-for-One Deals in their respective Categories

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | System displays all valid one-for-one deals for each specific category in the system. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. All valid one-for-one deals for the selected category are fully displayed. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. User logs in. 2. User selects the category he wants to view in the main page. 3. The system queries the backend server for the name, date, vendor, location, category, and terms and conditions of deals of all valid one-for-one deals for the deals belonging to the selected category. 4. The system displays the image and the name of all valid deals in the selected category in a ListView. |
| Alternative Flows: | AF-S4: If there are no valid one-for-one deals in the selected category   1. The system will display a blank page with a pop-up saying “Sorry there are no deals currently.”. |
| Exceptions: | - |
| Includes: | 1. Use Case 2 |
| Special Requirements: | - |
| Assumptions: | 1. The order of displayed deals is in the order of how they are being retrieved. 2. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 4: Selecting One-for-One Deals

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | User browses through a filtered list of displayed one-for-one deals based on the category selected and selects a desired deal. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. 3. Deals must be displayed in the system. |
| Exit-Conditions: | 1. User placed in waitlist of selected deal on the server. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. Displaying One-for-One Deals in their respective Categories (Use Case 3) has been invoked. 2. User selects the preferred one-for-one deal based on the displayed deals. 3. System will display name, date, vendor, location, category, and terms and conditions of the selected deal. 4. User clicks on the “Find a match” button. 5. System will display a popup with a scrollable list of all MRT stations, with each location having their own checkbox. 6. User selects up to 5 preferred locations by clicking each location’s checkbox. 7. System acknowledges the user’s selections and sends it to the backend server. 8. System displays a popup displaying the text “Finding you a match! This may take a while” to inform the user that he has been placed on the waitlist. |
| Alternative Flows: | AF-S6: If there are no preferred locations   1. The user exits the location selection popup. 2. The system returns to Step 3.   AF-S8: If the user presses the “Cancel” button   1. The system informs the backend server the cancellation of the matching. 2. The popup closes and the system returns to Step 3. |
| Exceptions: | - |
| Includes: | 1. Use Case 3 |
| Special Requirements: | - |
| Assumptions: | 1. The user knows which outlets the deal is available at. 2. The user knows which MRT stations are near the outlets. 3. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 5: Finding a Match

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | The system matches two users who have selected the same deal and selected at least one identical location. |
| Entry-Conditions: | 1. The waitlist on the server must contain at least 1 user. 2. A new user is placed on the waitlist. |
| Exit-Conditions: | 1. User has been successfully matched with another user from the waitlist. 2. Both users that are matched are removed from the waitlist. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. Selecting One-for-One Deals (Use Case 4) has been invoked. 2. System receives the id of both users, the deal id and the location names that both users have in common from the server. 3. System notifies both users that they have been successfully matched through a popup. 4. The first user clicks on the “Chat” button in the popup to prompt the creation of the chatroom and enters the chatroom. 5. The second user clicks on the “Chat” button in the popup to enter the same chatroom. |
| Alternative Flows: | AF-S4: If there are no users in the filtered waitlist to be matched.   1. System will continue to display the popup in Use Case 4, Step 8 until user is successfully matched. |
| Exceptions: | - |
| Includes: | 1. Use Case 4 |
| Special Requirements: | - |
| Assumptions: | 1. The user does not press the “Cancel” button (from Use Case 4 AF-S8) between Steps 1-6. 2. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 6: Deleting a Chatroom

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | Deletes one user’s side of the chatroom. |
| Entry-Conditions: | 1. Two users have successfully matched and have entered the same chatroom. |
| Exit-Conditions: | 1. When the user clicks on the “Delete Chat” button. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. Finding a Match (Use Case 5) has been invoked. 2. Users plan their meet-up in the chatroom. 3. User clicks on the “Delete Chat” button. 4. System closes the chatroom. 5. System notifies server of the deletion. 6. The chatroom will no longer be available for the user in the system. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use case 7: Unmatching a User

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | Allows a user to unmatch with a matched user. |
| Entry-Conditions: | 1. Two users have successfully matched and have entered the same chatroom. |
| Exit-Conditions: | 1. When the user clicks on the “Unmatch” button. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. Finding a Match (Use Case 5) has been invoked. 2. Users plan their meet-up in the chatroom. 3. User clicks on the “Unmatch” button. 4. System closes the chatroom for both users. 5. System notifies server of the deletion of the chatroom. 6. The chatroom will no longer be available for both users in the system. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 8: View Dropdown Menu

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | System displays the Dropdown Menu. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. System displays the Dropdown Menu. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. User clicks the Dropdown Menu icon. 2. System displays the Dropdown Menu with the options “Profile”, “Homepage”, “My Deals”, “Chat” and “Log Out”. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Use Case 2 |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 9: View Profile Page

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | System displays user’s profile page. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. System displays the Profile Page. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. View Dropdown Menu (Use Case 8) has been invoked. 2. User clicks on the “View Profile Page” button. 3. System displays the Profile Page, which contains the display picture and username of the user. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Use Case 8 |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 10: Change Display Name

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | User changes their Display Name. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. User has changed their Display Name. 2. System displays the Profile Page. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. View Profile Page (Use Case 9) has been invoked. 2. User clicks on the “Change Display Name” button. 3. System displays a page for the User to enter their new display name. 4. User enters their new display name and clicks the “Save” button. 5. System will display a popup with the text “Updating Display Name, This may take a while”. 6. System sends the updated Display Name to the server. 7. System displays the Profile Page again with the newly updated Display Name. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 11: Change Profile Picture

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | User changes their Profile Picture. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. User has changed their Profile Picture. 2. System displays the Profile Page. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. View Profile Page (Use Case 9) has been invoked. 2. User clicks on the “Change Profile Picture” button. 3. System displays the User’s photo gallery. 4. User selects their preferred photo and clicks the “Crop” button. 5. System will display a popup with the text “Uploading Image… This may take a while”. 6. System sends the updated Profile Picture to the server. 7. System displays the Profile Page again with the newly updated Profile Picture. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | - |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 12: Bring to Homepage

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | System displays user’s profile page. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. System displays the Homepage. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. View Dropdown Menu (Use Case 8) has been invoked. 2. User clicks on the “Homepage” button. 3. System displays the Homepage. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Use Case 8 |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 13: View Chats

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | System displays all chatrooms the User is in |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. System displays the specific chatroom the User wants to view. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. View Dropdown Menu (Use Case 8) has been invoked. 2. User clicks on the “Chat” button. 3. System displays a page with the Display Pictures of users who have matched with the user and the names of the users if the chatroom is not deleted. 4. User clicks on preferred chat to view the dialogue inside the chatroom. |
| Alternative Flows: |  |
| Exceptions: | - |
| Includes: | 1. Use Case 8 |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

Use Case 14: Log Out of System

|  |  |
| --- | --- |
| Actor: | System, User, Backend Server |
| Description: | System Logs the User out of the System. |
| Entry-Conditions: | 1. User must be logged in. 2. System must be connected to the backend server. |
| Exit-Conditions: | 1. System Logs the User out of the System. |
| Priority: | - |
| Frequency of Use: | - |
| Flow of Events: | 1. View Dropdown Menu (Use Case 8) has been invoked. 2. User clicks on the “Log Out” button. 3. System displays the Landing Page. |
| Alternative Flows: | - |
| Exceptions: | - |
| Includes: | 1. Use Case 8 |
| Special Requirements: | - |
| Assumptions: | 1. System must have access to an Internet Connection. |
| Notes and Issues: | - |

**Class Diagram of Entity Classes**

A close up of text on a white background

Description automatically generated

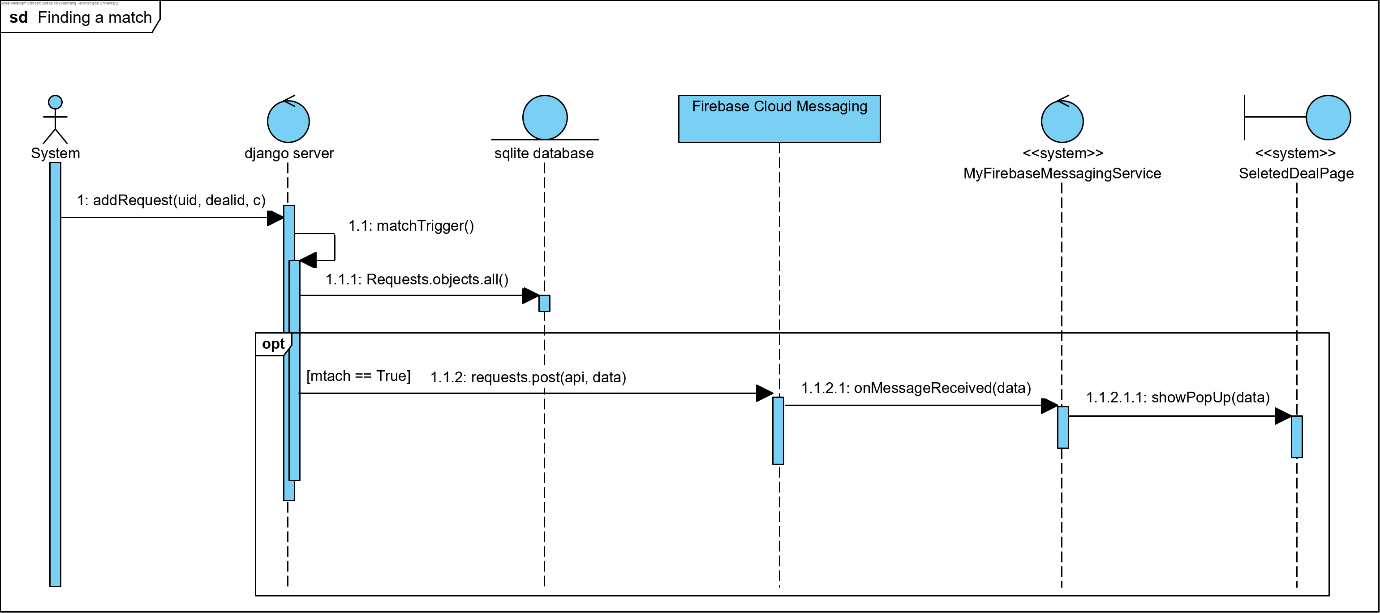
\*no relational arrows are added as entity classes are not directly linked to each other but linked to boundary and control classes

**Identification of Classes**

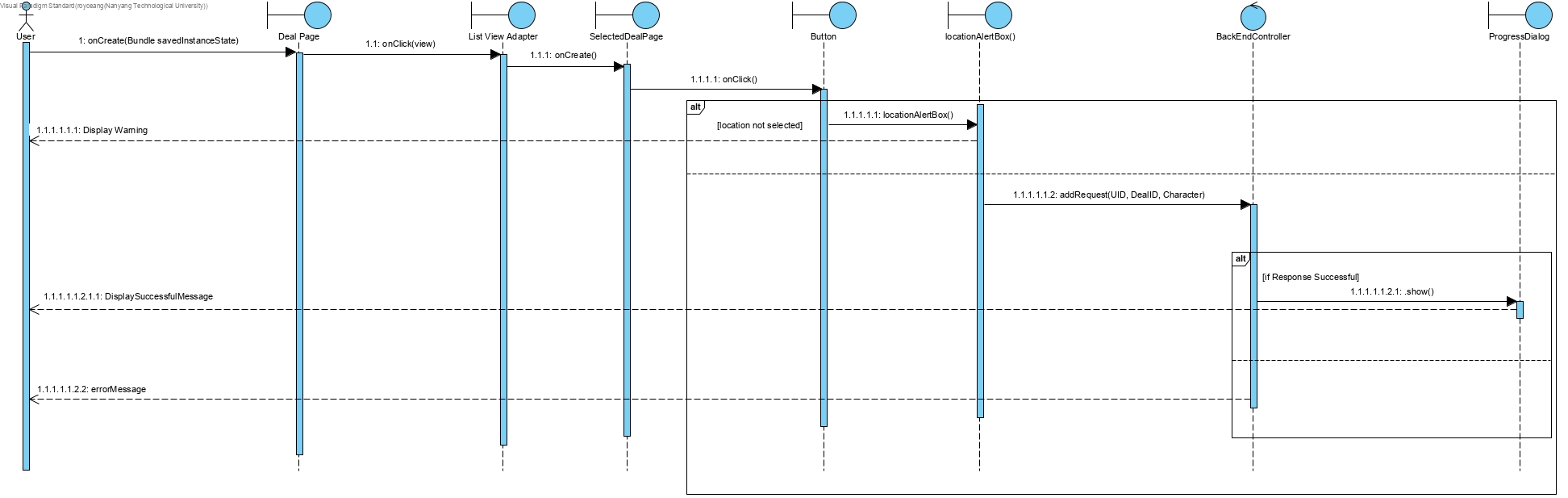
|  |  |
| --- | --- |
| Boundary Classes | 1. AllUsers.java 2. ChangeNameActivity.java 3. ChatActivity.java 4. EntertainmentDealsPage.java 5. FoodDealsPage.java 6. MainActivity.java 7. MatchedPersons.java 8. OtherDealsPage.java 9. ProfileActivity.java 10. RegisterActivity.java 11. RetailDealsPage.java 12. SelectedDealPage.java 13. StartActivity.java |
| Control Classes | 1. BackEndController.java 2. ListViewAdapter.java 3. ListViewAdapter2.java 4. MessageAdapter.java 5. MessageViewHolder.java 6. MyFirebaseMessagingService.java 7. NotificationHelper.java |
| Entity Classes | 1. ChatUser.java 2. Deal.java 3. Location.java 4. Messages.java 5. UserToken.java 6. Users.java |

**Sequence Diagrams of some Use Cases**

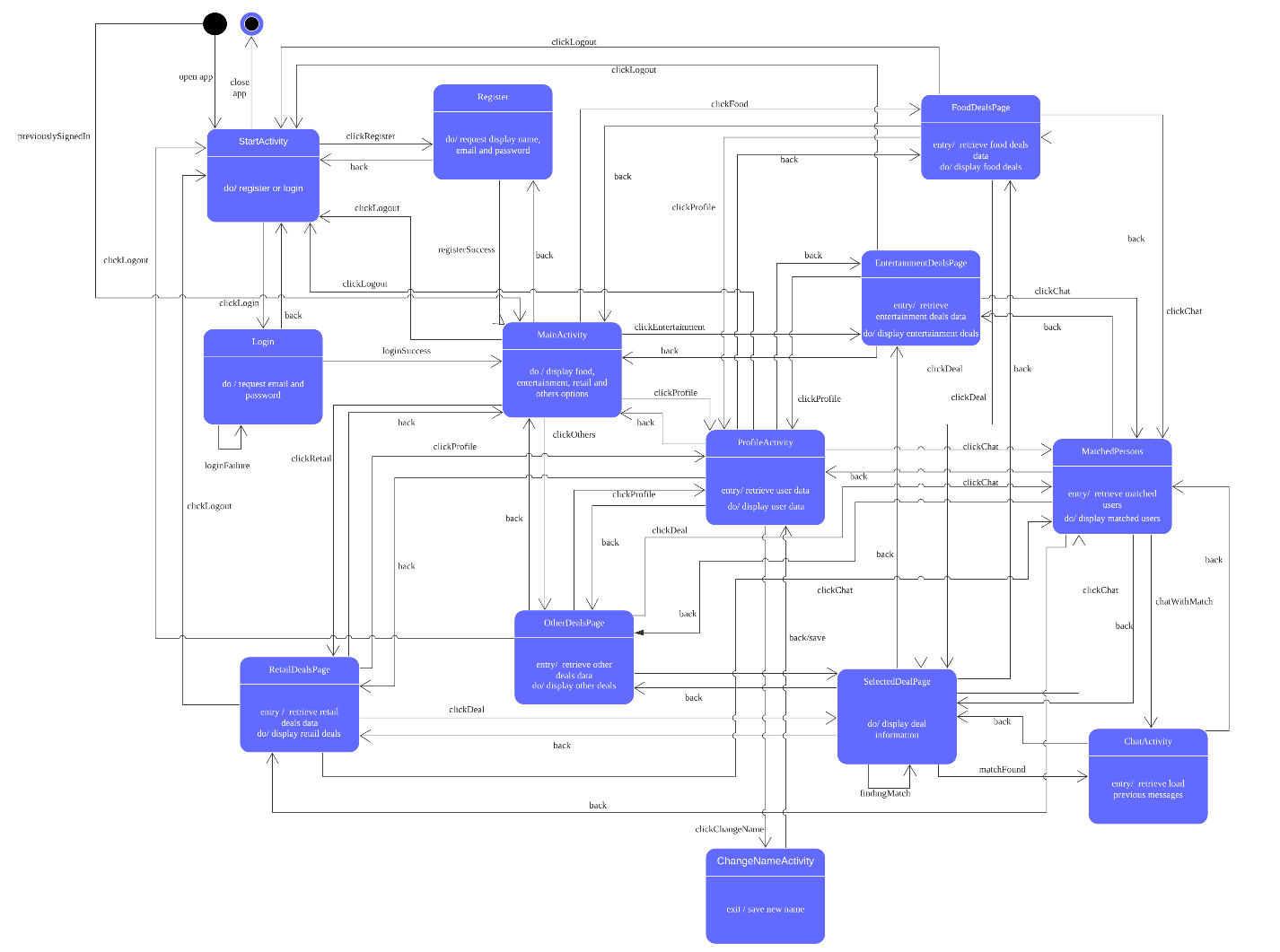
* + - 1. Finding a match



1. Selecting one-for-one Deals for entertainment/food/retail/others



**Initial dialog map**

****