College of Computing and Data Science
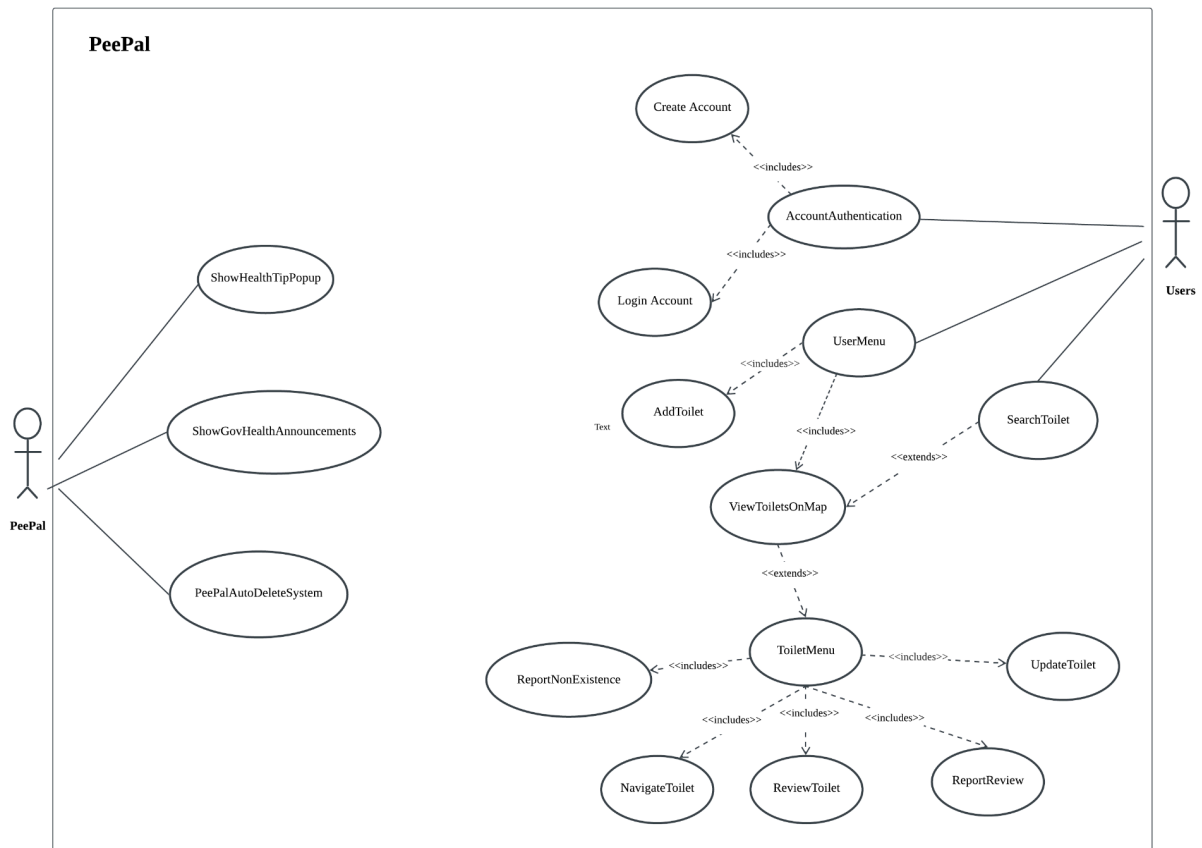
SC2006 - Software Engineering
Lab 2 Deliverables

FDAB Team 1

Name	Matriculation No.
Adam Soh Shi Jie	U2320112B
Joyce Lee Jia Xuan	U2320463F
Soong Jun Shen	U2340460H
Tan Kai Hooi	U2420694C
Liew Jia Wei	U2320233G

1. Complete Use Case Diagram

2. Use Case Description

1. For Functional Requirement #1 (PeePal)

Use Case ID:	#1-1		
Use Case Name:	ShowGovHealthAnnouncments		
Created By:	Jun Shen	Last Updated By:	Jun Shen
Date Created:	25/02/2025	Date Last Updated:	25/02/2025

Actor:	PeePal
Descriptions:	PeePal will automatically prompt the user with government health announcements tips.
Preconditions:	User successfully navigated to a toilet.
Postconditions:	A pop-up dialog containing a government health announcement is displayed on the user's screen within the app.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. The user navigates to a nearby toilet. 2. PeePal waits for a few random seconds. 3. PeePal shows a popup dialog containing a recent government health announcement. 4. The user closes the popup by tapping the close button.
Alternative Flows:	None
Exceptions	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Use Case ID:	#1-2		
Use Case Name:	ShowHealthTipPopup		
Created By:	Jun Shen	Last Updated By:	Jun Shen
Date Created:	25/02/2025	Date Last Updated:	25/02/2025

Actor:	PeePal
Descriptions:	PeePal will automatically prompt the user with health tips.
Preconditions:	User successfully navigated to a toilet.
Postconditions:	A pop-up dialog containing a health tip is displayed on the user's screen within the app.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	5. The user navigates to a nearby toilet. 6. PeePal waits for a few random seconds. 7. PeePal shows a popup dialog containing a short health tip like "Wash your hands". 8. The user closes the popup by tapping the close button.
Alternative Flows:	None
Exceptions	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Use Case ID:	#1-3		
Use Case Name:	PeePalAutoDeleteSystem		
Created By:	Joyce Lee	Last Updated By:	Joyce Lee
Date Created:	09/02/2025	Date Last Updated:	25/02/2025

Actor:	PeePal
Descriptions:	The system automatically removes flagged reviews and non-existent toilets when the number of reports exceeds a predefined threshold.
Preconditions:	None.
Postconditions:	Flagged reviews and non-existent toilets are deleted.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. If the amount of reports for a toilet exceeds a certain threshold, the system will auto delete the non existent toilet. 2. If the amount of report for a review exceeds a certain threshold, the system will auto delete the non existent review.
Alternative Flows:	None
Exceptions	1. If the system is offline or experiencing technical issues, the auto-deletion process may not execute properly.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

2. For Functional Requirement #2 (User Authentication)

Use Case ID:	#2-1
Use Case Name:	AccountAuthentication

Created By:	Joyce Lee	**Last Updated By:**	Adam Soh
Date Created:	11/02/2025	**Date Last Updated:**	25/02/2025

Actor:	Users
Descriptions:	Users must authenticate their accounts on PeePal by either logging in if they are existing users or creating a new account if they are first-time users.
Preconditions:	1. If users choose to logout, users must be logged in to Peepal.
Postconditions:	1. User account is created. 2. User is logged in.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. Users enter PeePal. 2. Users are prompted with an account authentication menu. 3. If users choose CreateAccount, the system invokes the **CreateAccount** use case.
Alternative Flows:	AF-S3: 1. If users choose LoginAccount, the system invokes the **LoginAccount** use case.
Exceptions	1. If the user is not found, the system will display an error message.
Includes:	1. LoginAccount 2. CreateAccount
Special Requirements:	None

Assumptions:	None
Notes and Issues:	None

Use Case ID:	#2-2		
Use Case Name:	CreateAccount		
Created By:	Tan Kai Hooi	Last Updated By:	Tan Kai Hooi
Date Created:	9/2/2025	Date Last Updated:	25/2/2025

Actor:	Users
Descriptions:	Allows a visitor to create a new account in the system.
Preconditions:	1. The user must click on the 'Create Account' button.
Postconditions:	1. A new user account is created and stored in the database. 2. The user receives a confirmation email.
Priority:	High
Frequency of Use:	Medium

Flow of Events:	1. User opens the application.
	2. User clicks on the "Create Account" button.
	3. User enters required details (e.g., name, email, username, password, contact number).
	4. System validates the input fields.
	5. If all details are correct:
	   a. System stores the user's information in the database.
	   b. System sends an account verification email.
	   c. System confirms account creation and redirects the user to the login page.

Alternative Flows:	AF-S5:
	1. Invalid Input:
	   - If required fields are missing, the system prompts the user to complete them.
	   - If the password is weak, the system suggests a stronger password.
	2. Username/Email Already Exists:
	   - The system notifies the user and asks them to enter a different username/email.

	3. Verification Required: - If email verification is enabled, the user must verify their email before logging in.
Exceptions	- If the database is down, account creation fails, and an error message is displayed. - If the email service is down, the confirmation email may not be sent immediately.
Includes:	- InputValidation - PasswordEncryption
Special Requirements:	None
Assumptions:	- The visitor does not already have an account. - The system has sufficient storage capacity for new user data.
Notes and Issues:	None

Use Case ID:	#2-3		
Use Case Name:	LoginAccount		
Created By:	Tan Kai Hooi	**Last Updated By:**	Tan Kai Hooi
Date Created:	9/2/2025	**Date Last Updated:**	25/2/2025

Actor:	User
Descriptions:	Allows a user to login to an existing account.
Preconditions:	1. Users must click on the login button.
Postconditions:	1. User logs in and is brought to the home page. 2. User menu appears.
Priority:	High
Frequency of Use:	Low
Flow of Events:	1. User opens the application. 2. User clicks on the login button. 3. User enters a username and password.

	4. System validates credentials against the database. 5. If credentials are correct, the user is authenticated and redirected to Home Screen.

Alternative Flows:	AF-S3:
	    1. Forgot Password:
	        - User clicks on "Forgot Password."
	        - System prompts the user to reset password via email verification.
	
	AF-S5:
	    1. Invalid Credentials:
	        - If the username or password is incorrect:
	        - The system displays an error message.
	        - Users are prompted to re-enter credentials.

Exceptions	- If the database is down, the login process fails, and an error message is displayed. - If there are multiple failed login attempts, the account may be locked for security reasons.
Includes:	None
Special Requirements:	None
Assumptions:	- Users have already registered and have valid credentials.
Notes and Issues:	None

3. For Functional Requirement #3 (Toilet Features)

Use Case ID:	#3-1		
Use Case Name:	ToiletMenu		
Created By:	Adam Soh	Last Updated By:	Adam Soh
Date Created:	9/2/2025	Date Last Updated:	9/2/2025

Actor:	User
Descriptions:	Allows users to choose options provided in the toilet menu.
Preconditions:	1. The user must be logged in and authenticated. 2. Users must select one toilet.

Postconditions:	- User will be brought to the toilet menu interface, where they are able to choose from a variety of different features.
Priority:	Low
Frequency of Use:	Medium
Flow of Events:	1. Users will be able to view the information details of the selected toilet. 2. Users can perform the actions provided in the menu.
Alternative Flows:	AF-S3: 1. If User selects NAVIGATE, the system will invoke the **NavigateToilet** use case. 2. If User selects REVIEW, the system will invoke the **ReviewToilet** use case. 3. If User selects REPORT REVIEW, the system will invoke the **ReportReview** use case. 4. If User selects UPDATE, the system will invoke the **UpdateToilet** use case. 5. If User selects REPORT TOILET, the system will invoke the **ReportNonExistence** use case.
Exceptions:	None.
Includes:	1. NavigateToilet 2. ReviewToilet 3. ReportReview 4. UpdateToilet 5. ReportNonExistence
Special Requirements:	None.
Assumptions:	None.
Notes and Issues:	None

Use Case ID:	#3-2		
Use Case Name:	NavigateToilet		
Created By:	Tan Kai Hooi	**Last Updated By:**	Tan Kai Hooi
Date Created:	9/2/2025	**Date Last Updated:**	25/2/2025

Actor:	User
Descriptions:	Allows users to navigate to the chosen toilet.
Preconditions:	1. The User must be logged in and authenticated. 2. The User must have their location service turned on. 3. The User must be on the Toilet menu.
Postconditions:	Successfully return the optimal route to the chosen toilet.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The user clicks on the 'Navigate' button. 2. System gets the user's current location. 3. System calculates the optimal route from the user's location to the chosen toilet location. 4. System displays the navigation instructions on the ToiletMapUI.
Alternative Flows:	None.
Exceptions:	- Location service is not turned on.
Includes:	None
Special Requirements:	None
Assumptions:	- Toilet is in Singapore. - The user is able to travel there by foot.
Notes and Issues:	None

Use Case ID:	#3-3		
Use Case Name:	ReviewToilet		
Created By:	Tan Kai Hooi	**Last Updated By:**	Tan Kai Hooi
Date Created:	9/2/2025	**Date Last Updated:**	25/2/2025

Actor:	User
Descriptions:	Allows users to review the chosen toilet.

Preconditions:	1. The User must be logged in and authenticated.
Postconditions:	A review of the toilet is published by the user and will be able to be viewed on the toilet menu.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. User clicks on the 'Review' button on the toilet menu, leading them to the review menu page. 2. User writes a review and gives a rating out of 5. 3. User posts the review. 4. Posting is successful and a confirmation dialog message will pop up.
Alternative Flows:	AF-S1: 1. If posting is unsuccessful, a review failed dialog message will pop up.
Exceptions:	None.
Includes:	None.
Special Requirements:	None.
Assumptions:	- Toilet exists in the database.
Notes and Issues:	None

Use Case ID:	#3-4		
Use Case Name:	ReportReview		
Created By:	Tan Kai Hooi	**Last Updated By:**	Tan Kai Hooi
Date Created:	25/2/2025	**Date Last Updated:**	25/2/2025

Actor:	User
Descriptions:	Allows users to report a chosen review.
Preconditions:	1. The user must be logged in and authenticated. 2. The user is on the toilet menu page for that specific toilet.
Postconditions:	The chosen review is flagged by the system.

Priority:	High
Frequency of Use:	Low
Flow of Events:	1. The User selects the "Report" button. 2. A pop up will appear, confirming if the user wants to report the user. 3. A successful dialog message will be displayed.
Alternative Flows:	AF-S1: 1. If the report failed, a report failed dialog message will be displayed.
Exceptions:	None.
Includes:	None.
Special Requirements:	None.
Assumptions:	- Review exists in the database.
Notes and Issues:	None

Use Case ID:	#3-5		
Use Case Name:	ReportNonExistence		
Created By:	Joyce Lee	**Last Updated By:**	Joyce Lee
Date Created:	25/2/2025	**Date Last Updated:**	25/2/2025

Actor:	User
Descriptions:	Allows users to report a non-existent toilet.
Preconditions:	1. The User must be logged in and authenticated.
Postconditions:	The chosen toilet is flagged by the system.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	1. The User selects the "Report" feature button. 2. The system will flag the toilet.

Alternative Flows:	None.
Exceptions:	None.
Includes:	None.
Special Requirements:	None.
Assumptions:	1. Toilet exists in the database.
Notes and Issues:	None

Use Case ID:	#3-6		
Use Case Name:	UpdateToilet		
Created By:	Joyce Lee	**Last Updated By:**	Jun Shen
Date Created:	09/02/2025	**Date Last Updated:**	25/02/2025

Actor:	User
Descriptions:	The user can update the details of an existing toilet, including its location, status, or any additional information stored in the system.
Preconditions:	1. The user is logged in and authenticated. 2. Toilet data is available in the system. 3. The user is viewing the toilet details by invoking **ToiletMenu.**
Postconditions:	The toilet information is successfully updated.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	1. The user selects the option to "Update Toilet Info" on the toilet's detail page. 2. The system displays the toilet's information with editable fields: toilet name, toilet location, location status and bidet availability, with "Update" and

	"Cancel" buttons. 3. The user may input into the fields that need changes.

Alternative Flows:	AF-S3:
	   1. If the user selects the "Update" button, the system saves the updated details and updates the toilet's detail page.
	AF-S3:
	   1. If the user selects the "Cancel" button, the system discards the user's changes and returns to the toilets details page.

Exceptions	1. If the user selects the "Update" button with empty fields, the system prompts an "Invalid Input" error dialog. 2. If the app is not connected to the Internet, the system prompts the user with a "No Internet Connection" error dialog.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

4. For Functional Requirement #4 (Users)

Use Case ID:	#4-1

Use Case Name:	UserMenu		
Created By:	Adam Soh	Last Updated By:	Adam Soh
Date Created:	9/2/2025	Date Last Updated:	9/2/2025

Actor:	User
Descriptions:	Display User Menu.
Preconditions:	The User must be logged in and authenticated.
Postconditions:	Successfully display a detailed list of toilets closest to the user's current location.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The user permits the request to detect the current location. 2. The system displays a real-time map that displays the nearby locations of all toilets by invoking **ViewToiletOnMap** use case. 3. If the user wishes to add a new toilet, the system will invoke the **AddToilet** use case.
Alternative Flows:	None
Exceptions:	1. User doesn't allow permission to detect the current location.
Includes:	1. ViewToiletOnMap 2. AddToilet
Special Requirements:	None.
Assumptions:	None.
Notes and Issues:	None

Use Case ID:	#4-2		
Use Case Name:	AddToilet		
Created By:	Joyce Lee	Last Updated By:	Joyce Lee
Date Created:	11/02/2025	Date Last Updated:	25/02/2025

Actor:	User
Descriptions:	Allows users to add a new toilet.
Preconditions:	1. User is logged in and authenticated.
Postconditions:	A new toilet is added.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	1. The user selects the option to "Add Toilet" on UserMenu. 2. The system prompts the user to enter toilet details, including name, status, bidet availability, and optional reviews. 3. The user enters the required details. 4. The app will use the user's current location as the location for the new toilet. 5. The system saves the new toilet data. 6. The system confirms that the toilet has been added successfully.
Alternative Flows:	1. If the user selects CANCEL at any time, the system returns back to the **UserMenu** use case.
Exceptions	1. If the admin enters an invalid search value (e.g., empty search or special characters), the system prompts for a valid input. 2. If the system fails to add the toilet due to a technical error, it displays a message prompting the admin to retry.
Includes:	None
Special Requirements:	None

Assumptions:	None
Notes and Issues:	None

Use Case ID:	#4-3
Use Case Name:	SearchToilet

Created By:	Tan Kai Hooi	**Last Updated By:**	Joyce Lee
Date Created:	9/2/2025	**Date Last Updated:**	25/2/2025

Actor:	User
Descriptions:	Allows users to search for toilets in any location.
Preconditions:	1. The User must be logged in and authenticated.
Postconditions:	1. Successfully display the exact toilet location on the map.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. Users intend to search the toilet and input the toilet's name and location. 2. If the toilet exists, the system will invoke the **ViewToiletOnMap** use case.
Alternative Flows:	None
Exceptions:	1. If the toilet is not found, the system will display an error message.
Includes:	1. ViewToiletOnMap
Special Requirements:	None
Assumptions:	1. The toilet is in Singapore.
Notes and Issues:	None

Use Case ID:	#4-4		
Use Case Name:	ViewToiletOnMap		
Created By:	Tan Kai Hooi	Last Updated By:	Joyce Lee
Date Created:	9/2/2025	Date Last Updated:	25/2/2025

Actor:	User
Descriptions:	Display the nearby toilet on the map depending on the input location.
Preconditions:	1. The User must be logged in and authenticated.
Postconditions:	1. Successfully display the nearby toilet location on the map.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The system will retrieve the user location or toilet location. 2. The system will find the nearby toilets. 3. The nearby toilets are displayed on the map. 4. The user can select any toilet on the map and the system will invoke **ToiletMenu** use case.
Alternative Flows:	None
Exceptions:	None
Includes:	1. ToiletMenu
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

3. Class diagram of Entity Classes
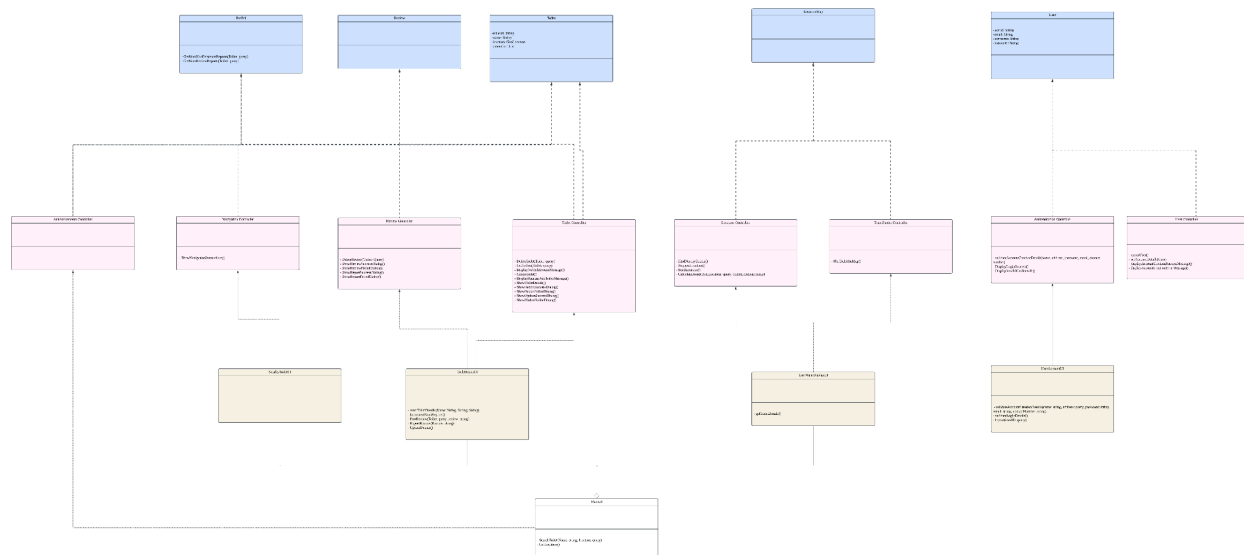
4. Key boundary classes and control classes
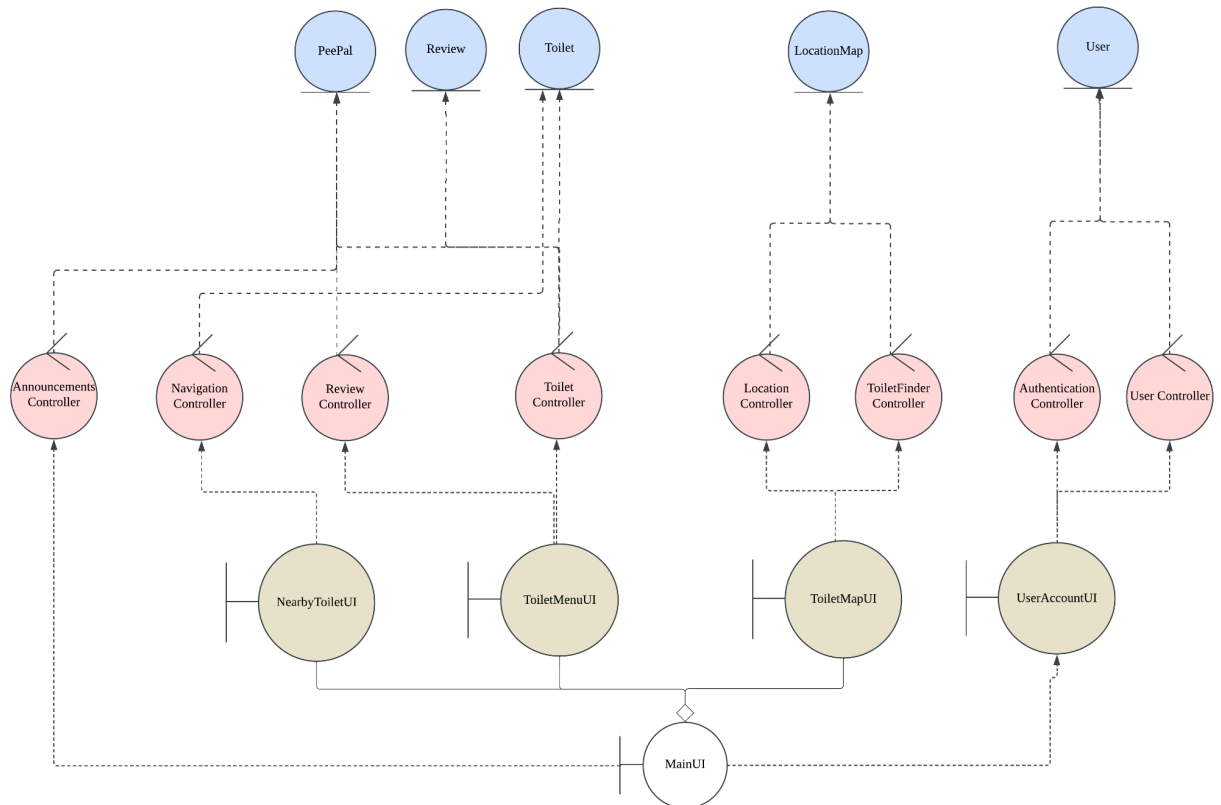
5. Sequence diagrams of some use cases

I.   For Use Cases under I (PeePal)

I.I ShowGovHealthAnnouncements
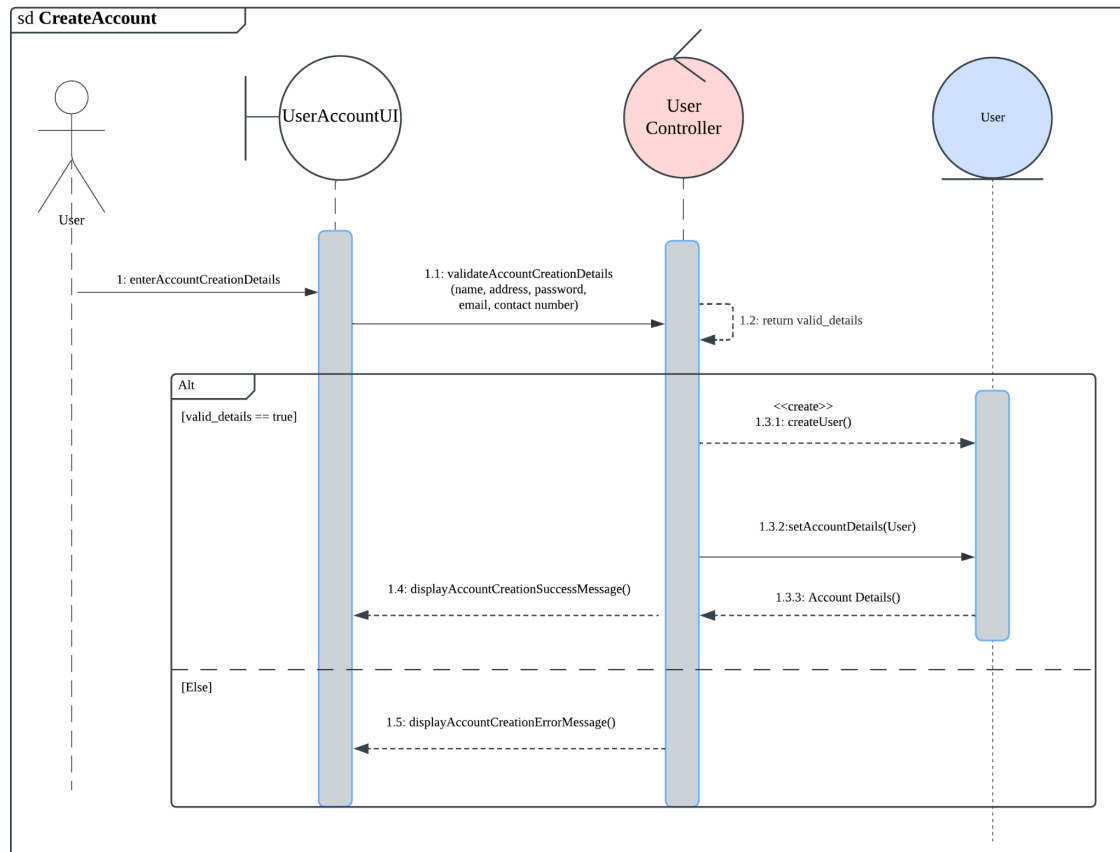**I.II ShowHealthTipPopup**

I.III PeePalAutoDeleteSystem
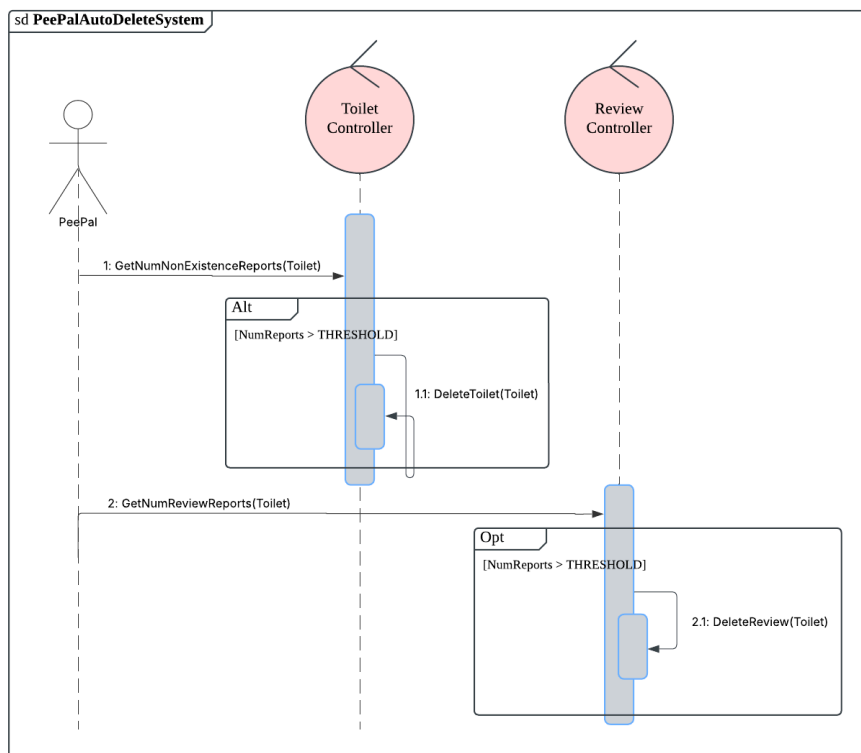
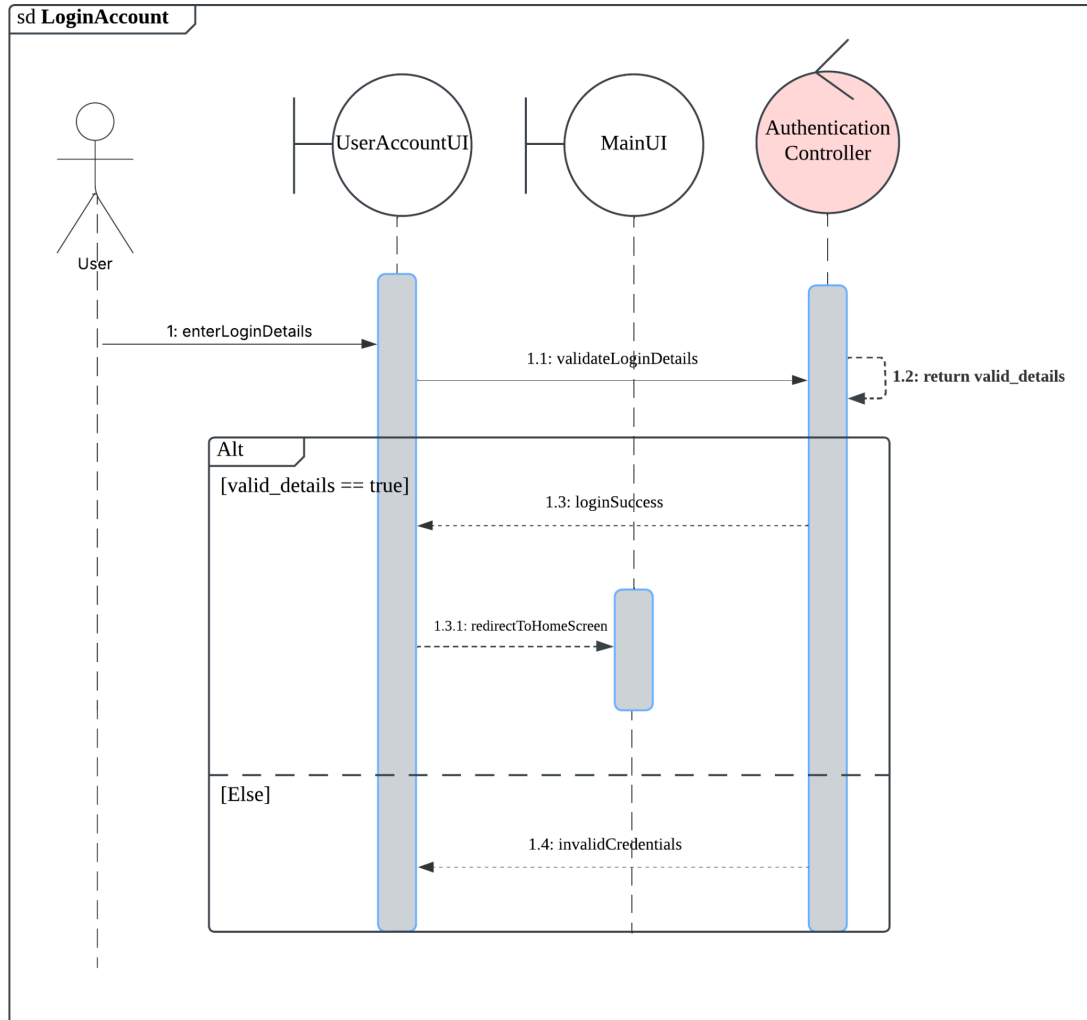## II. For Use Cases under II (User Authentication)
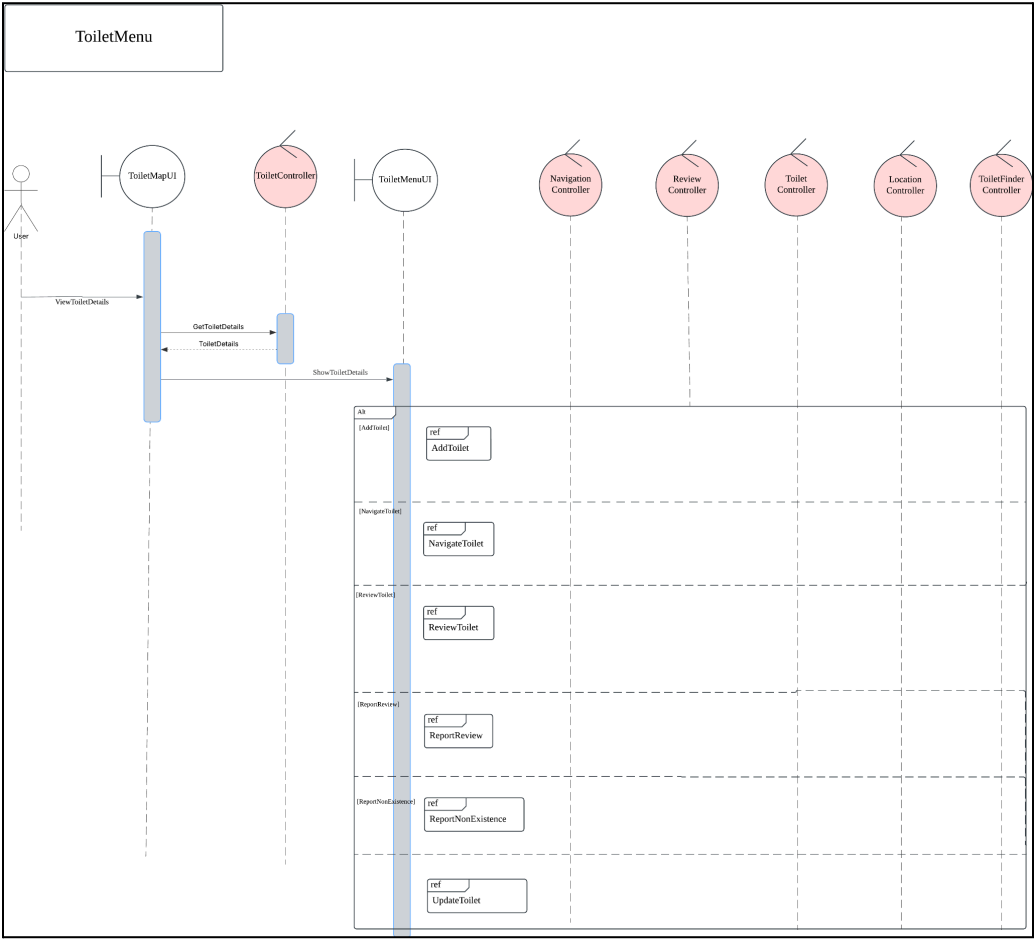
II.I AccountAuthentication
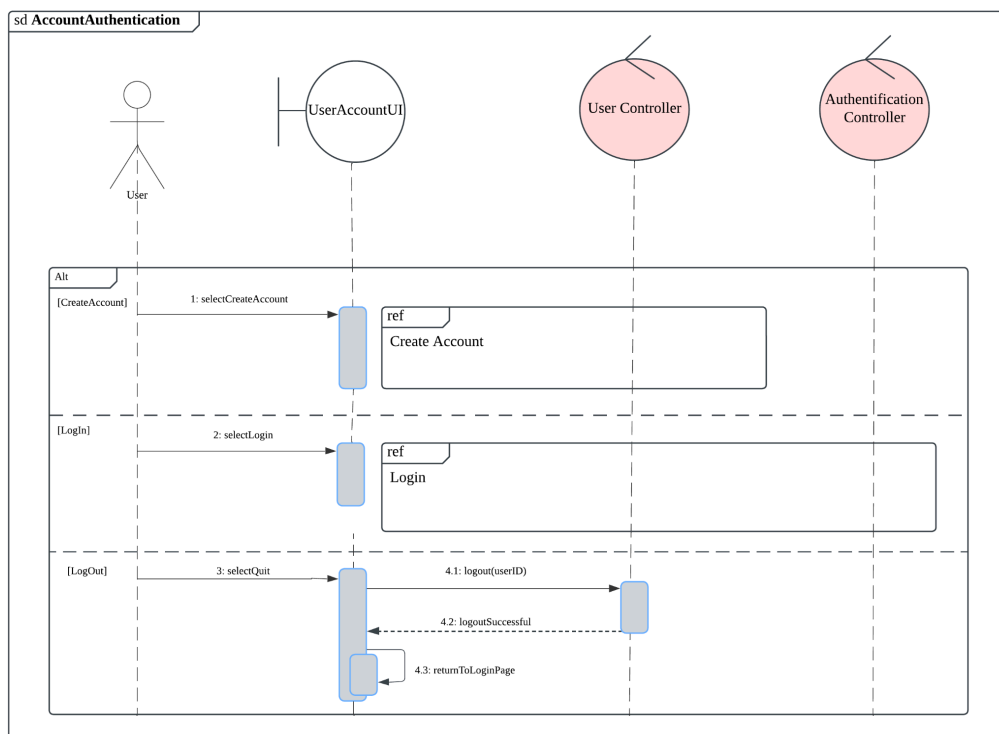
II.II CreateAccount

II.III LoginAccount

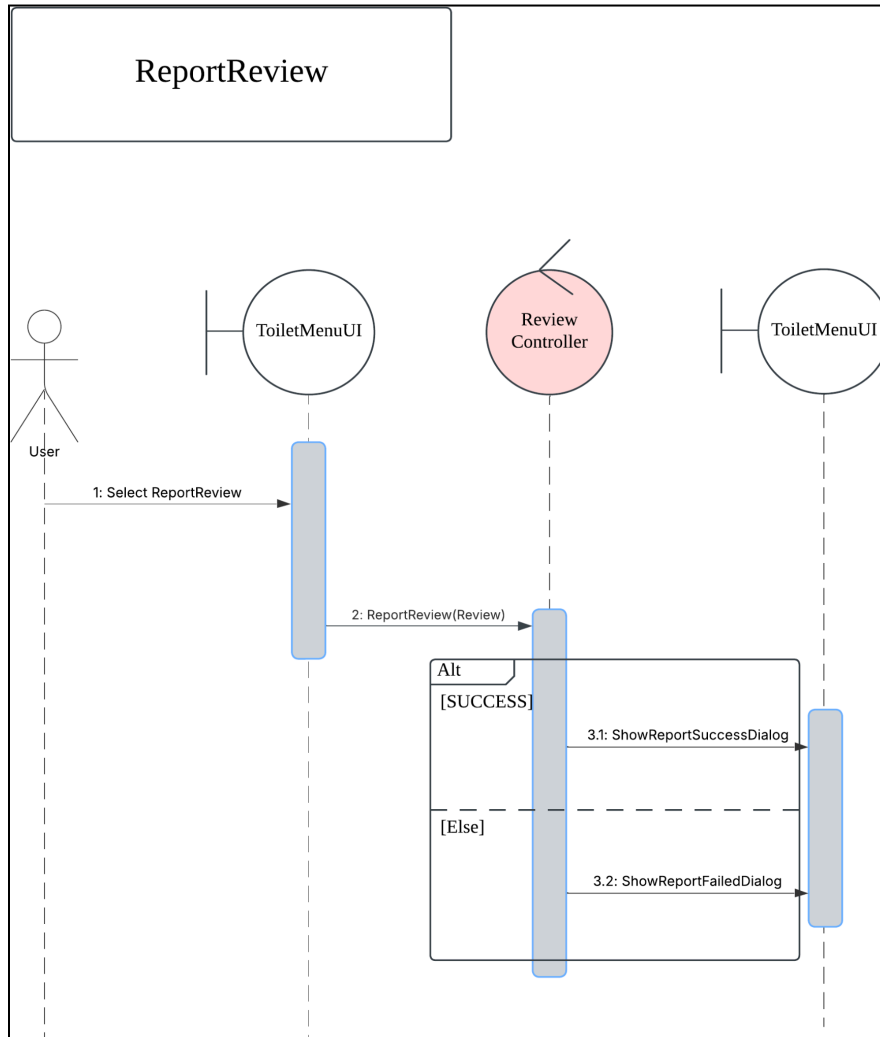# III. For Use Cases under III (ToiletFeatures)
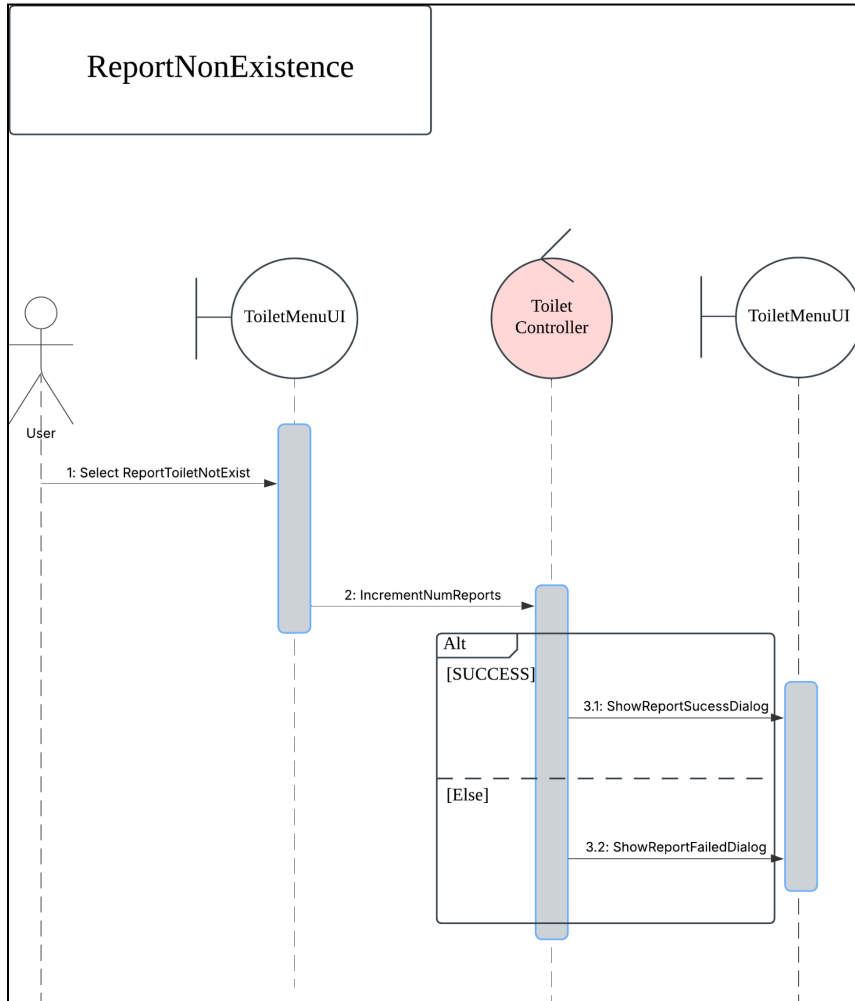
III.I ToiletMenu
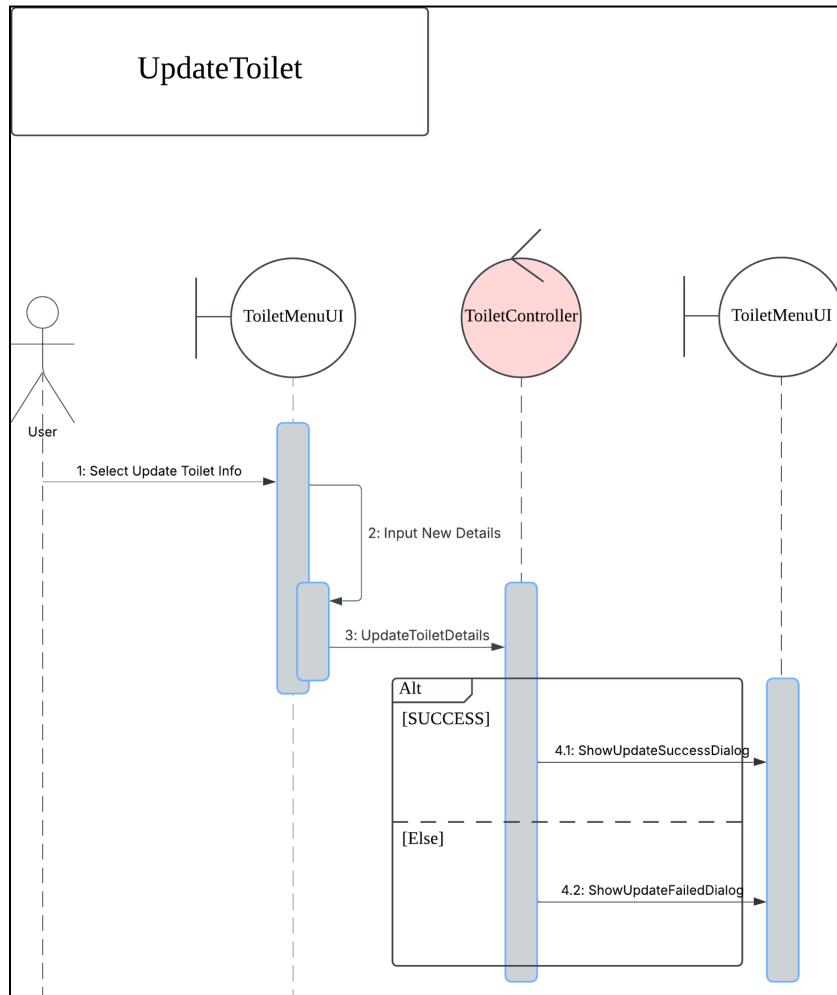
III.II NavigateToilet

III.III ReviewToilet

ReviewToilet

User

ToiletMenuUI

Toilet Controller

Review Controller

ToiletMenuUI

1: SubmitReview

3: PostReview(Toilet, Review)

2: GetToiletDetails

ToiletDetails

Alt

[SUCCESS]

4.1: ShowReviewSuccessDialog

[Else]

4.2: Show ReviewFailedDialog
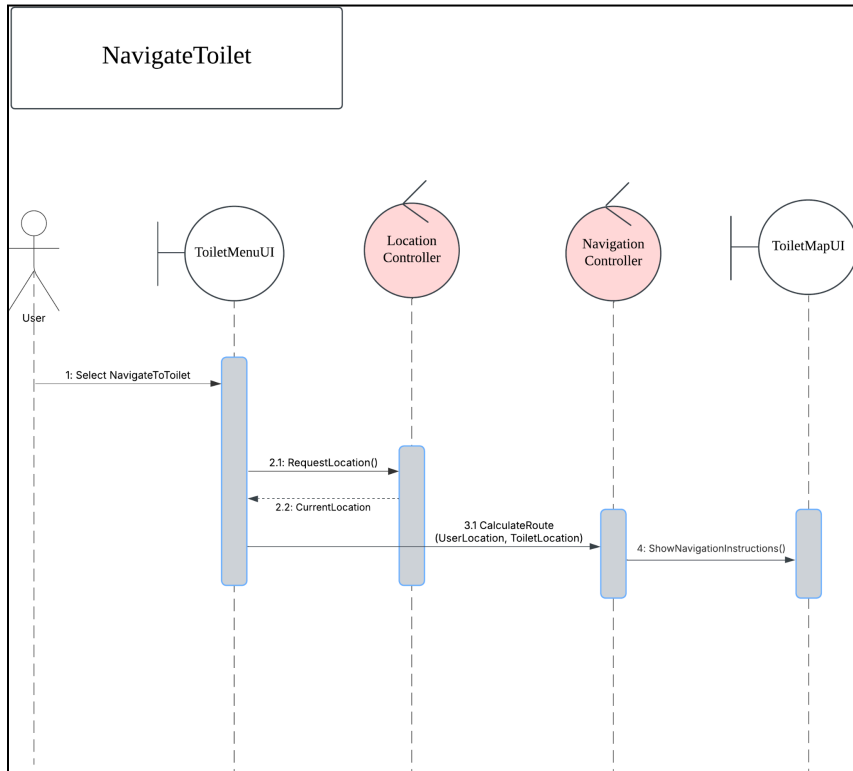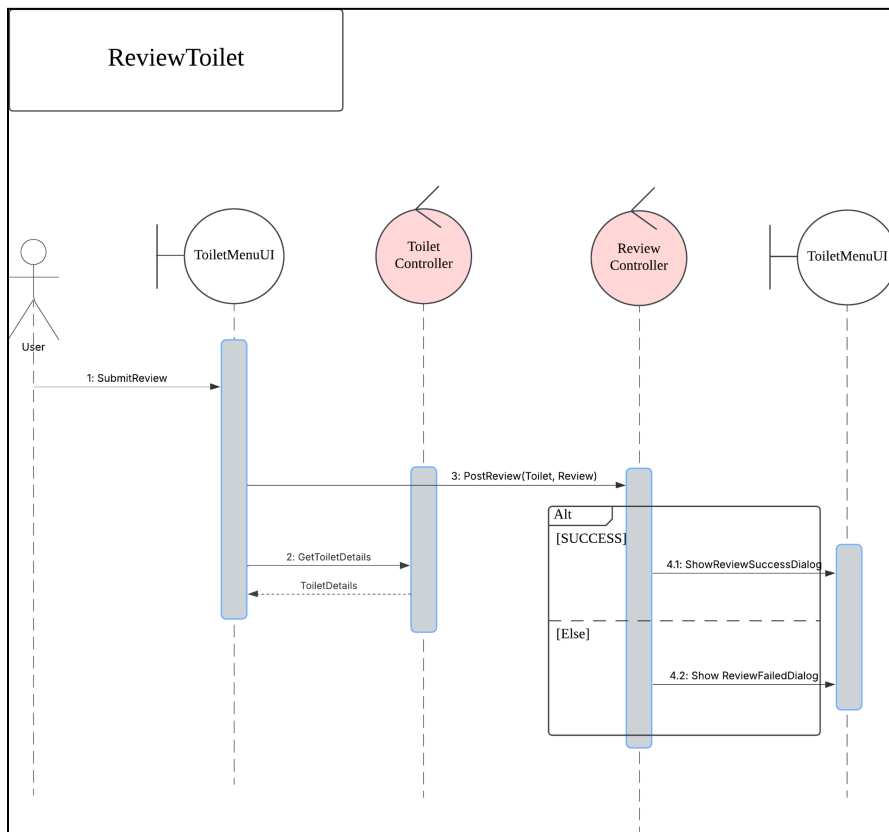
III.IV ReportReview

III.V ReportNonExistence

III.VI UpdateToilet

UpdateToilet

User → ToiletMenuUI: 1: Select Update Toilet Info

ToiletMenuUI → ToiletMenuUI: 2: Input New Details

ToiletMenuUI → ToiletController: 3: UpdateToiletDetails

Alt

[SUCCESS]

ToiletController → ToiletMenuUI: 4.1: ShowUpdateSuccessDialog

[Else]

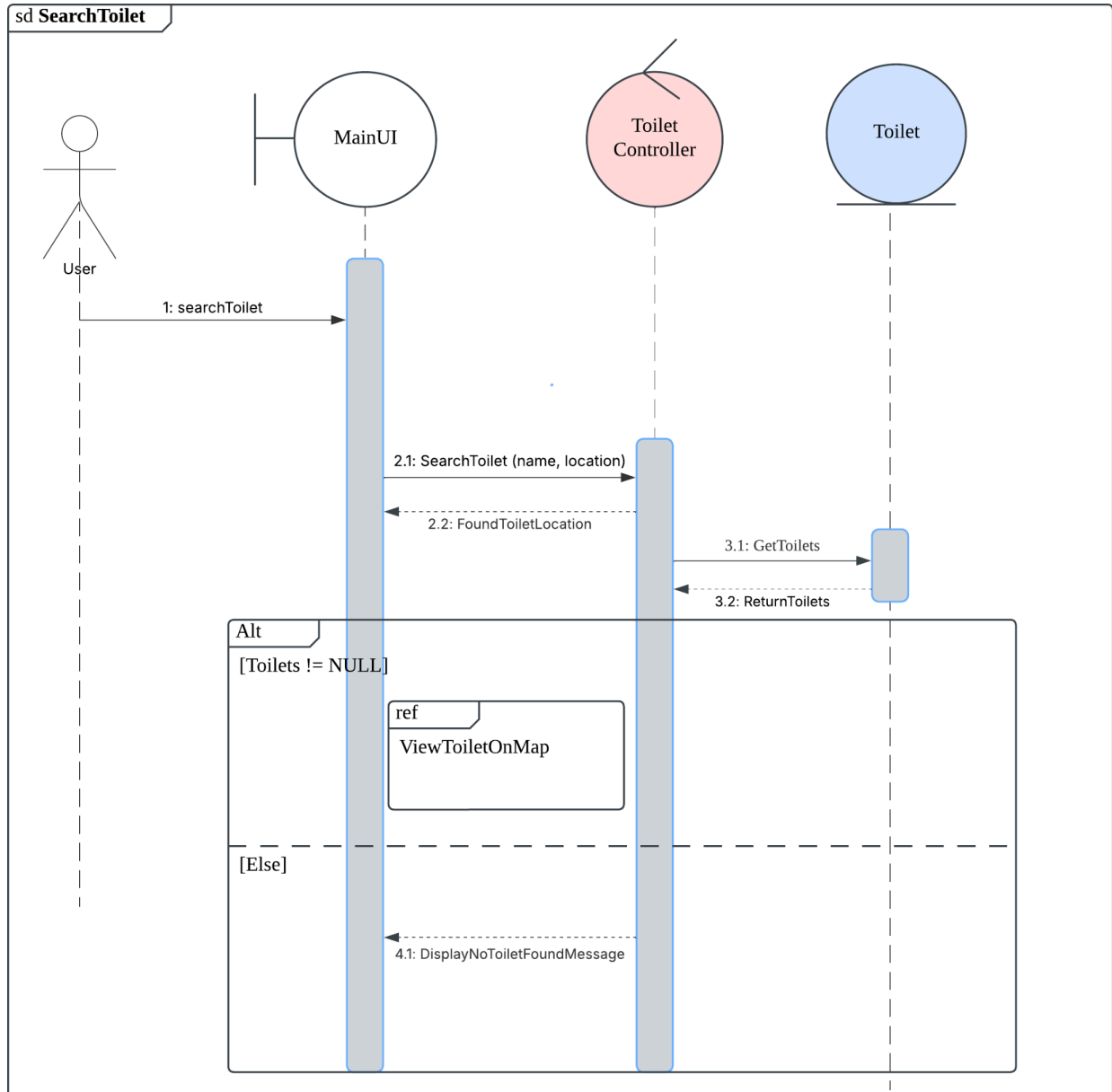ToiletController → ToiletMenuUI: 4.2: ShowUpdateFailedDialog

IV. For Use Cases under IV (Users)
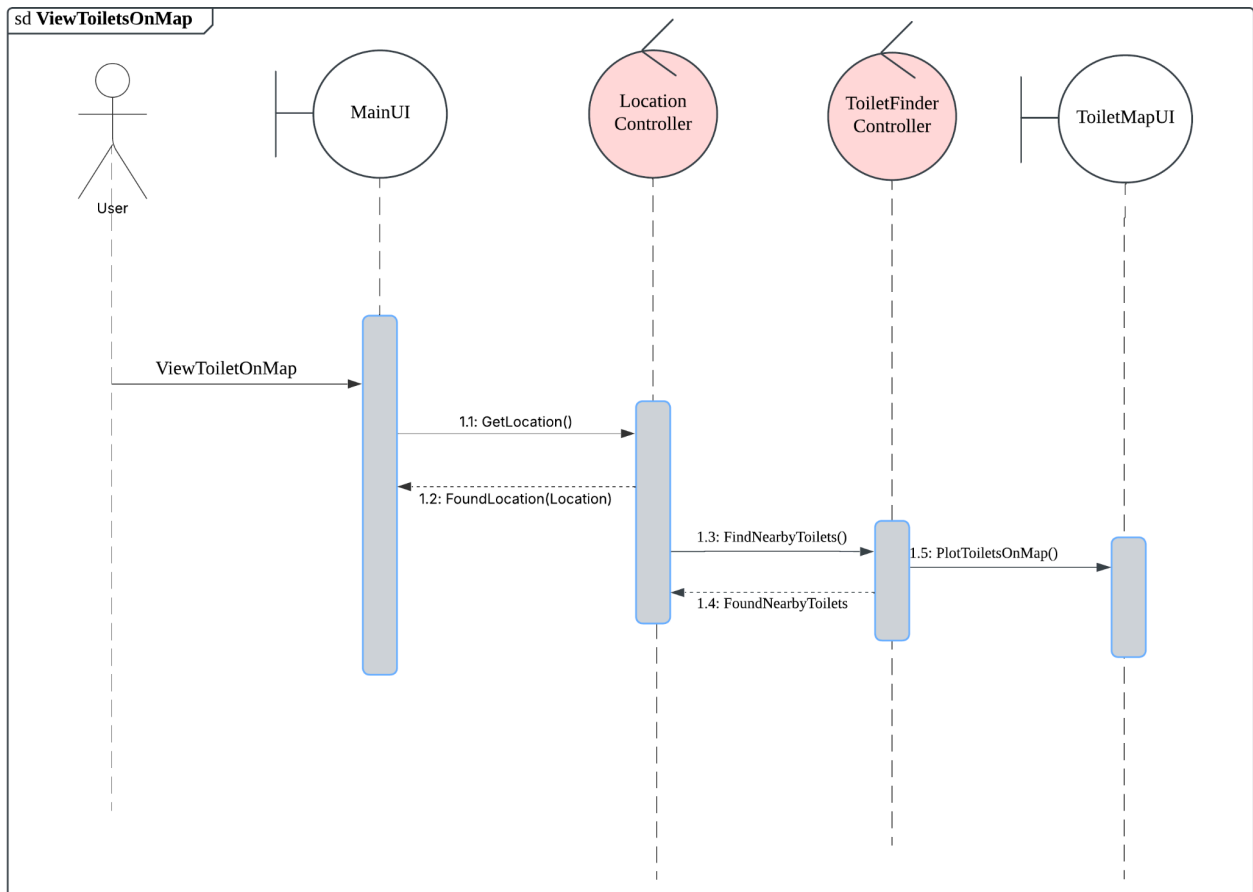
IV.I UserMenu

IV.II AddToilet
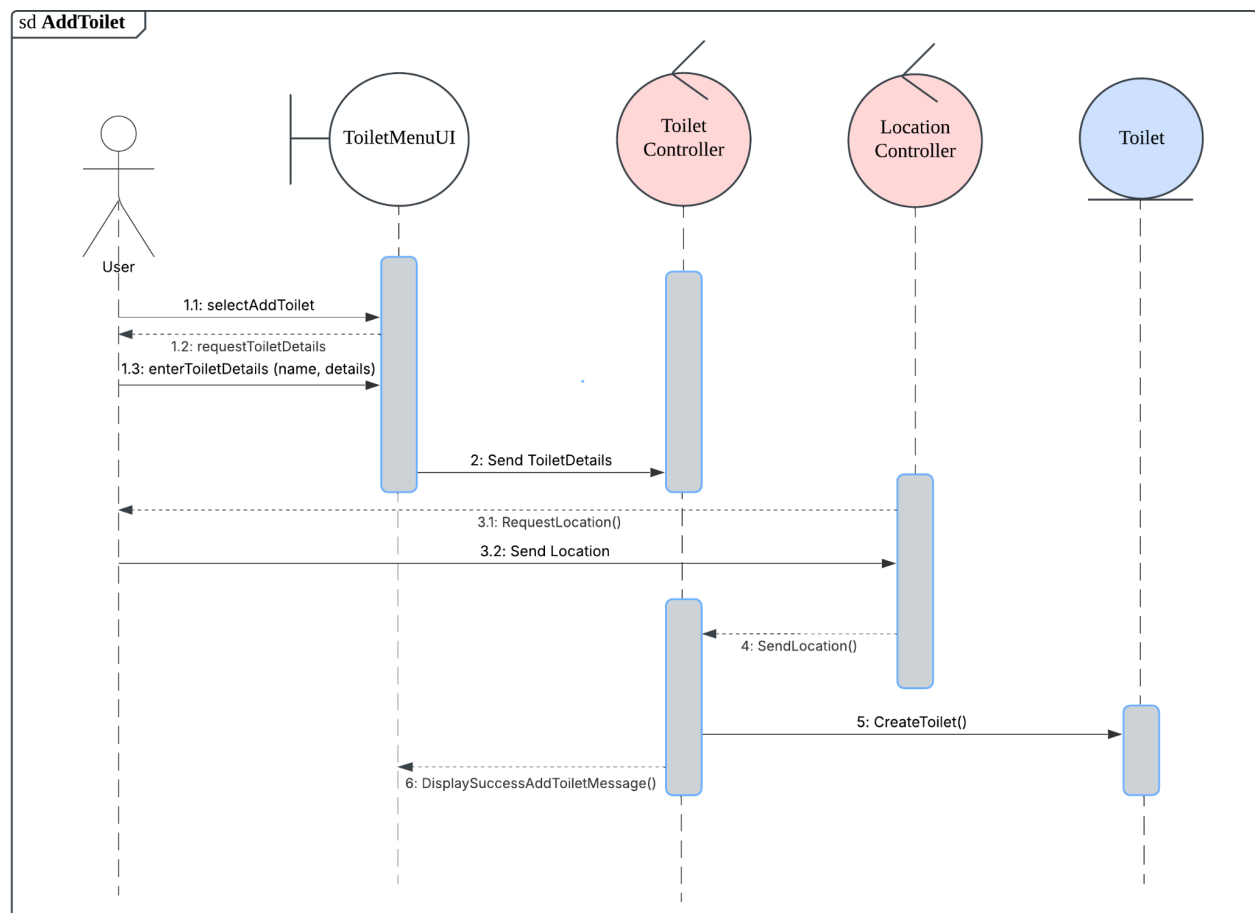
IV.III SearchToilet

IV.IV ViewToiletOnMap

6. Dialog diagram