

**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

SC2006 - Software Engineering

Lab 2 Deliverables

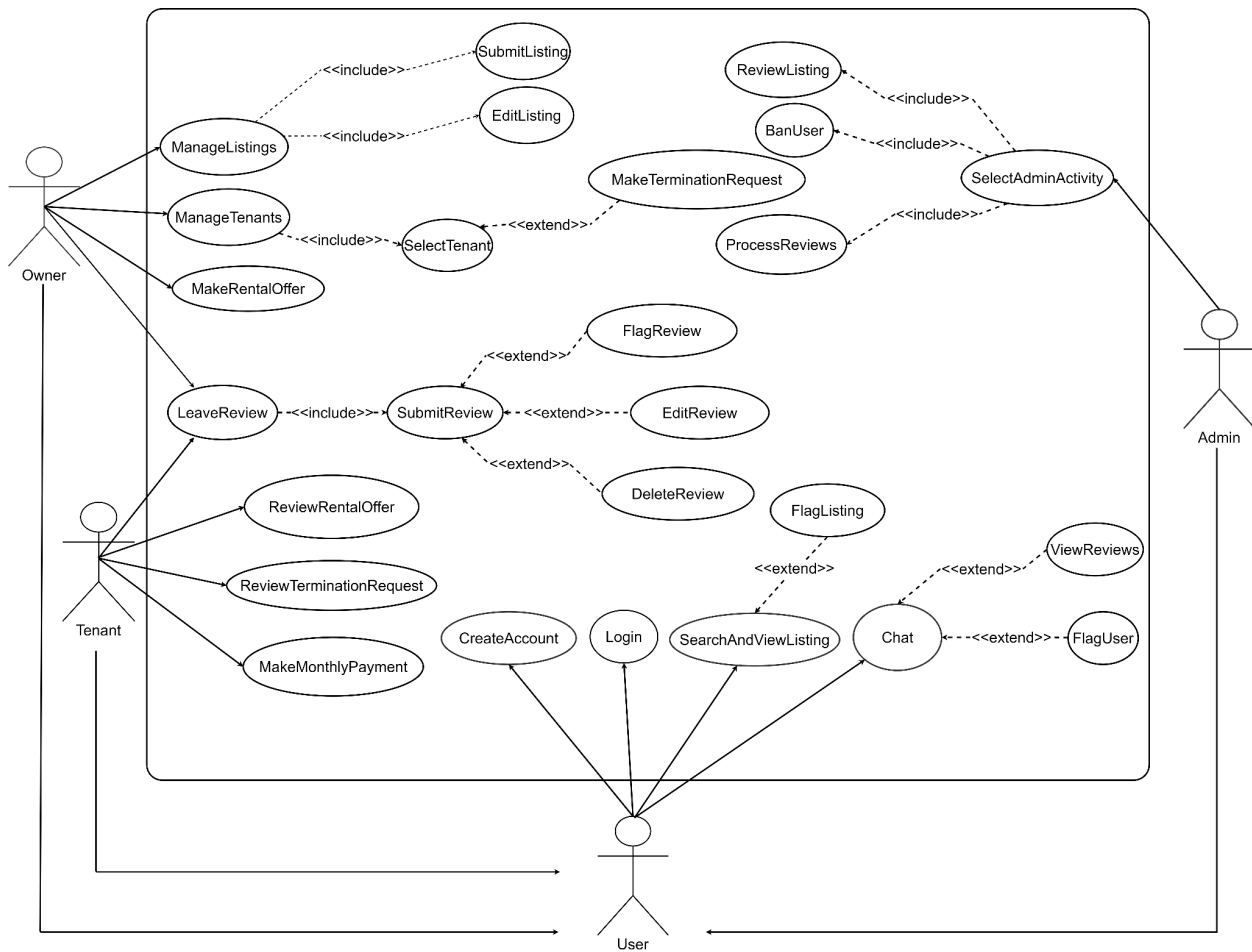
Lab Group	SDAD
Team	HomeGoWhere
Members	Lim Jing Rong (U2323260D)
	Loo Ping Wee (U2321814J)
	Goh Shuen Wei (U2321093D)
	Ng Jing En (U2321510K)
	Babu Sankar Nithin Sankar (U2323953B)
	Lau Zhan You (U2320608L)

1. Use Case Model and Use Case descriptions.....	4
A. Use Case Diagram.....	4
B. Use Case Descriptions.....	5
Functional Requirement #1.....	5
I.I SelectAdminActivity.....	5
I.II ReviewListing.....	7
I.III BanUser.....	8
I.IV ProcessReviews.....	9
Functional Requirement #2.....	10
II.I ManageListings.....	10
II.II SubmitListing.....	11
II.III EditListing.....	12
II.IV ManageTenants.....	13
II.V SelectTenant.....	14
II.VI MakeTerminationRequest.....	15
II.VII MakeRentalOffer.....	16
Functional Requirement #3.....	17
III.I ReviewRentalOffer.....	17
III.II MakeMonthlyPayment.....	19
III.III ReviewTerminationRequest.....	21
III.IV LeaveReview.....	22
III.V SubmitReview.....	24
III.VI FlagReview.....	25
III.VII EditReview.....	26
III.VIII DeleteReview.....	28
Functional Requirement #4.....	29
IV.I CreateAccount.....	29
IV.II Login.....	30
IV.III SearchAndViewListing.....	31
IV.IV FlagListing.....	32
IV.V Chat.....	33
IV.VI FlagUser.....	34
IV.VII ViewReviews.....	35
2. Key Classes Diagram.....	36
3. Entity, Control and Boundary Class Diagram.....	37
4. Sequence Diagrams of Use Cases.....	38
I. For Use Cases under I.....	38
I.I SelectAdminActivity.....	38
I.II ReviewListing.....	39

I.III BanUser.....	40
I.IV ProcessReviews.....	41
II. For Use Cases under II.....	42
II.I ManageListings.....	42
II.II SubmitListing.....	43
II.III EditListing.....	44
II.IV ManageTenants.....	45
II.V SelectTenant.....	46
II.VI MakeTerminationRequest.....	47
II.VII MakeRentalOffer.....	48
III. For Use Cases under III.....	49
III.I ReviewRentalOffer.....	49
III.II MakeMonthlyPayment.....	50
III.III ReviewTerminationRequest.....	51
III.IV LeaveReview.....	52
III.V SubmitReview.....	53
III.VI FlagReview.....	54
III.VII EditReview.....	55
III.VIII DeleteReview.....	56
IV. For Use Cases under IV.....	57
IV.I Create Account.....	57
IV.II Login.....	58
IV.III SearchAndViewListing.....	59
IV.IV FlagListing.....	60
IV.V Chat.....	61
IV.VI FlagUser.....	62
IV.VII ViewReviews.....	63
5. Initial Dialog map.....	64

1. Use Case Model and Use Case descriptions

A. Use Case Diagram



B. Use Case Descriptions

Functional Requirement #1

I.I SelectAdminActivity

Use Case ID:	#1-1		
Use Case Name:	SelectAdminActivity		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	Admin
Description:	Allows the admin to review listings posted by Owners, ban users for inappropriate behavior and facilitates the management of reviews posted by Owners and Tenants.
Preconditions:	Admin is logged in and is authenticated.
Postconditions:	None
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The system uses the included use case Login to verify the Owner.2. Once logged in, the system prompts the Admin to select the desired activity: PROCESS_REVIEW, BAN_USER or REVIEW_LISTING.3. If the admin selects the activity REVIEW_LISTING, then the admin uses the included use case ReviewListing to approve or reject a user's application to post a listing.4. If the admin selects the activity PROCESS_REVIEWS, then the Admin uses the included use case ProcessReviews to decide whether a review is to be removed.5. If the admin selects the activity BAN_USER, then the Admin uses the included use case BanUser to decide whether a user should be banned.6. If the Admin selects the activity QUIT, the system returns to the login screen.
Alternative Flows:	None
Exceptions:	None

Includes:	<ol style="list-style-type: none">1. ReviewListing2. BanUser3. ProcessReview
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

I.II ReviewListing

Use Case ID:	#1-2		
Use Case Name:	ReviewListing		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	Admin
Description:	Allows the admin to review listings posted by Owners for deletion or keeping.
Preconditions:	Admin is logged in and is authenticated.
Postconditions:	Flagged listing is deleted or ignored.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none">1. The HomeGoWhere lists the listings that have been flagged by it or Users.2. Admin selects the listing that the Admin wants to process.3. The system displays the content of the listing and prompts the Admin to select the "DELETE" or "IGNORE" button.4. If the Admin selects the "DELETE" button, the system will delete the flagged listing.
Alternative Flows:	<u>AF-S3: If the Admin selects IGNORE</u> <ol style="list-style-type: none">1. If the Admin selects the "IGNORE" button, the system will ignore the flagged listing and remove it from the list of flagged listings.2. The system will then return to the list of remaining flagged listings.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

I.III BanUser

Use Case ID:	#1-3		
Use Case Name:	BanUser		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	Admin
Description:	Allows the admin to ban a User by their user ID.
Preconditions:	Admin is logged in and is authenticated.
Postconditions:	Users are either banned or not banned by the admin.
Priority:	High
Frequency of Use:	Low
Flow of Events:	<ol style="list-style-type: none">1. Admin reviews the lists of flagged users that have been flagged by other users.2. Admin selects the flagged users case that the Admin wants to process.3. The system displays the content of the case and prompts the Admin to select the "IGNORE" or "BAN" button.4. If the Admin selects the "BAN" button, the system will ban the user.
Alternative Flows:	<u>AF-S3: If the Admin selects IGNORE</u> <ol style="list-style-type: none">1. If the Admin selects the "IGNORE" button, the system will ignore the flagged users case and remove it from the list of flagged listings.2. The system will then return to the list of remaining flagged users.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

I.IV ProcessReviews

Use Case ID:	#1-4		
Use Case Name:	ProcessReviews		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	Admin
Description:	Allows the admin to process a review for deletion or for keeping.
Preconditions:	Admin is logged in and is authenticated.
Postconditions:	Flagged review is deleted or ignored.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. The HomeGoWhere lists the reviews that have been flagged by it or Users. 2. Admin selects the review that the Admin wants to process. 3. The system displays the content of the review and prompts the Admin to select the "IGNORE" or "DELETE" button. 4. If the Admin selects the "DELETE" button, the system will delete the flagged review.
Alternative Flows:	<u>AF-S3: If the Admin selects IGNORE</u> <ol style="list-style-type: none"> 1. If the Admin selects the "IGNORE" button, the system will ignore the flagged review and remove it from the list of flagged reviews. 2. The system will then return to the list of remaining flagged reviews.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Functional Requirement #2

II.I ManageListings

Use Case ID:	#2-1		
Use Case Name:	ManageListings		
Created By:	Lim Jing Rong	Last Updated By:	Lim Jing Rong
Date Created:	30th August 2024	Date Last Updated:	30th August 2024

Actor:	Owner
Description:	Allows Owner to manage the listing for the properties they are renting out or selling.
Preconditions:	Owner is logged in and is authenticated.
Postconditions:	None
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">The Owner will be provided with 2 use cases<ol style="list-style-type: none">SubmitListingEditListingIf the Owner selects the activity SUBMIT_LISTING, then the Owner uses the included use case SubmitListing to post a listing.If the Owner selects the activity EDIT_LISTING, then the Owner uses the included use case EditListing to edit a listing.
Alternative Flows:	None
Exceptions:	None
Includes:	<ol style="list-style-type: none">SubmitListingEditListing
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

II.II SubmitListing

Use Case ID:	#2-2		
Use Case Name:	SubmitListing		
Created By:	Loo Ping Wee	Last Updated By:	Lim Jing Rong
Date Created:	26th August 2024	Date Last Updated:	30th August 2024

Actor:	Owner
Description:	Allows Owner to create a listing for the property they are renting out or selling, and it gets sent to the database for Users to view.
Preconditions:	<ol style="list-style-type: none"> 1. Owner is logged in and is authenticated 2. All required fields in the form are filled
Postconditions:	None
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. Owner fills all the required fields in the form presented to them. 2. Owner selects the "Submit" button, and SubmitListing is called. 3. Checks that all required fields are filled. 4. Formdata is sent to the database and saved.
Alternative Flows:	<u>AF-S3: Owner did not fill in the required fields of the form</u> <ol style="list-style-type: none"> 1. Owner has not filled in all the required fields of the form. 2. Owner selects the "SUBMIT" button, and SubmitListing is called. 3. Checks that all required fields are filled are not passed. 4. Formdata is not sent to the database, Owner is prompted to field in all required fields, no changes to the form.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

II.III EditListing

Use Case ID:	#2-3		
Use Case Name:	EditListing		
Created By:	Lim Jing Rong	Last Updated By:	Lim Jing Rong
Date Created:	30th August 2024	Date Last Updated:	30th August 2024

Actor:	Owner
Description:	Allows Owner to edit a listing for the property they are renting out or selling, and it updates the database for Users to view.
Preconditions:	<ol style="list-style-type: none"> 1. Owner is logged in and is authenticated. 2. All required fields in the form are filled.
Postconditions:	None
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. Owner selects on his listing, he will be allowed to edit the listing 2. Owner selects the "SUBMIT" button, and SubmitListing is called. 3. Checks that all required field are filled 4. Formdata is sent to the database and saved.
Alternative Flows:	<u>AF-S3: Owner did not fill in the required fields of the form</u> <ol style="list-style-type: none"> 1. Owner has not filled in all the required fields of the form. 2. Owner selects the "SUBMIT" button, SubmitPropertylisting is called. 3. Checks that all required fields are filled and are not passed. 4. Formdata is not sent to the database, Owner is prompted to field in all required fields, no changes to the form.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

II.IV ManageTenants

Use Case ID:	#2-4		
Use Case Name:	ManageTenants		
Created By:	Loo Ping Wee	Last Updated By:	Lim Jing Rong
Date Created:	26th August 2024	Date Last Updated:	31st August 2024

Actor:	Owner
Description:	Allows Owner view the information of all Tenants that have rented from the Owner.
Preconditions:	Owner is logged in and is authenticated.
Postconditions:	None
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. Owner navigates to the “Rental Information” screen of a specific rental property. 2. Under the “Manage Tenants” section, the System displays the details of all Tenants currently renting at the property. 3. The Owner is able to manage a specific Tenant by selecting on their name. 4. If the Owner selects the activity SELECT_TENANT, then the Owner uses the included use case SelectTenant.
Alternative Flows:	None
Exceptions:	None
Includes:	<ol style="list-style-type: none"> 1. SelectTenant
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

II.V SelectTenant

Use Case ID:	#2-5		
Use Case Name:	SelectTenant		
Created By:	Loo Ping Wee	Last Updated By:	Lim Jing Rong
Date Created:	26th August 2024	Date Last Updated:	31st August 2024

Actor:	Owner
Description:	Allows Owner to select a Tenant they wish to interact with.
Preconditions:	Owner is logged in and is authenticated.
Postconditions:	None
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The Owner can choose to terminate the Tenants that are currently renting from the Owner.2. If the Owner selects the activity MAKE_TERMINATION_REQUEST, then the Owner uses the included use case MakeTerminationRequest.
Alternative Flows:	None
Exceptions:	None
Includes:	<ol style="list-style-type: none">1. MakeTerminationRequest
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

II.VI MakeTerminationRequest

Use Case ID:	#2-6		
Use Case Name:	MakeTerminationRequest		
Created By:	Lim Jing Rong	Last Updated By:	Lim Jing Rong
Date Created:	30th August 2024	Date Last Updated:	30th August 2024

Actor:	Owner
Description:	Allows Owner to select a Tenant they wish to terminate.
Preconditions:	Owner is logged in and is authenticated.
Postconditions:	None
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none">1. The Owner will select the Tenant that he wishes to terminate.2. When the Owner selects the "Terminate Lease" button, it will prompt a request to terminate the lease.
Alternative Flows:	None
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

II.VII MakeRentalOffer

Use Case ID:	#2-7		
Use Case Name:	MakeRentalOffer		
Created By:	Lim Jing Rong	Last Updated By:	Lim Jing Rong
Date Created:	30th August 2024	Date Last Updated:	30th August 2024

Actor:	Owner
Description:	Allows Owner to make a rental offer to Tenant.
Preconditions:	Owner is logged in and is authenticated.
Postconditions:	None
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> The Owner can send a rental offer to the Tenant <ol style="list-style-type: none"> The rental offer will include the rent per month. The rental offer will include the rental deposit required. The rental offer will include the end of lease date. Owner selects the "Send Offer" button. Checks that all required fields are filled are present. Offer sent to Tenant to accept.
Alternative Flows:	<u>AF-S3: Required fields not filled in</u> <ol style="list-style-type: none"> Owner has not filled in all the required fields of the form. Owner selects the "Send Offer" button. Checks that all required fields are filled are not passed. Formdata is not sent to the database, Owner is prompted to field in all required fields, no changes to the form.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Functional Requirement #3

III.I ReviewRentalOffer

Use Case ID:	#3-1		
Use Case Name:	ReviewRentalOffer		
Created By:	Lim Jing Rong	Last Updated By:	Goh Shuen Wei
Date Created:	24th August 2024	Date Last Updated:	12th September 2024

Actor:	Tenant
Description:	Allows Tenants to review rental offer
Preconditions:	Tenant is logged in and is authenticated
Postconditions:	None
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The System displays the details of the rental offer, including rental price, amount required for rental deposit and lease expiry date.2. If the Tenant wants to accept the rental offer, the Tenant can select the "Accept" button to accept the rental offer.3. This will lead them to a card payment screen for the Tenant to pay the rental deposit. Tenant enters card details for payment.4. Tenant confirms the payment by selecting the "Confirm Payment" button.5. After paying, the System will display to the Tenant "Rental Offer Accepted".
Alternative Flows:	<p><u>AF-S2: When the Tenant selects "Reject"</u></p> <ol style="list-style-type: none">1. System will display "Rental Offer Rejected". <p><u>AF-S3: When the Tenant puts an invalid card number</u></p> <ol style="list-style-type: none">1. The Tenant inputs an invalid card number.2. System will display "Error, Invalid credit card details". <p><u>AF-S3: When the Tenant did not finish filling up the credit card information</u></p> <ol style="list-style-type: none">1. The Tenant did not input at least 1 of the required fields for the credit card details.

	<p>2. System will display "Missing required field".</p> <p><u>AF-S4: When the Tenant selects the "Cancel" icon</u></p> <p>1. System will cancel the transaction.</p>
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

III.II MakeMonthlyPayment

Use Case ID:	#3-2		
Use Case Name:	MakeMonthlyPayment		
Created By:	Lim Jing Rong	Last Updated By:	Goh Shuen Wei
Date Created:	24th August 2024	Date Last Updated:	12th September 2024

Actor:	Tenant
Description:	Allows Tenant to make monthly payment to Owner
Preconditions:	Tenant is logged in and is authenticated
Postconditions:	The payment is successfully made
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none">1. Tenant navigates to the "Rent Payment" screen.2. The system displays the outstanding amount for payment.3. Tenant selects the "Pay" button to proceed with payment.4. Tenant enters card details for payment.5. Tenant confirms the payment by selecting the "Confirm Payment" button.6. After paying, it will display to the Tenant "Payment Successful".
Alternative Flows:	<p><u>AF-S4: When the Tenant puts an invalid card number</u></p> <ol style="list-style-type: none">3. The Tenant inputs an invalid card number.4. System will display "Error, Invalid credit card details". <p><u>AF-S4: When the Tenant did not finish filling up the credit card information</u></p> <ol style="list-style-type: none">3. The Tenant did not input at least 1 of the required fields for the credit card details.4. System will display "Missing required field". <p><u>AF-S5: When the Tenant selects the "Cancel" icon</u></p> <ol style="list-style-type: none">1. System will cancel the transaction.
Exceptions:	None
Includes:	None
Special Requirements:	None

Assumptions:	None
Notes and Issues:	None

III.III ReviewTerminationRequest

Use Case ID:	#3-3		
Use Case Name:	ReviewTerminationRequest		
Created By:	Lim Jing Rong	Last Updated By:	Lim Jing Rong
Date Created:	24th August 2024	Date Last Updated:	24th August 2024

Actor:	Tenant
Description:	Allow the Tenant to review the termination request
Preconditions:	Tenant is logged in and is authenticated
Postconditions:	Lease will be terminated and Tenant to move out
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none">1. The Tenant will be notified about the termination request.2. The Tenant chooses to accept the termination request<ol style="list-style-type: none">a. The lease will be terminated.
Alternative Flows:	<u>AF-S2: The Tenant does not accept the termination request</u> <ol style="list-style-type: none">1. The lease will not be terminated.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

III.IV LeaveReview

Use Case ID:	#3-4		
Use Case Name:	LeaveReview		
Created By:	Lim Jing Rong	Last Updated By:	Goh Shuen Wei
Date Created:	24th August 2024	Date Last Updated:	12th September 2024

Actor:	Tenant/Owner
Description:	Allows the Tenant/Owner to leave a review for the Owner/Tenant that they interact with in the current rental.
Preconditions:	Tenant/Owner is logged in and is authenticated.
Postconditions:	Tenants can leave a review on the specified Owner and his house. Owner can leave a review on the specified Tenant.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. If user is Owner, <ol style="list-style-type: none"> a. Owner navigates to the "Rental Information" screen. b. Owner selects a specific Tenant under the "Manage Tenants" section. c. Owner can select the "Leave Review" button to leave a review for the corresponding Tenant. d. Owner uses the included use case SubmitReview to submit the review. 2. If user is Tenant, <ol style="list-style-type: none"> a. Tenant navigates to the "Rental Information" screen. b. Tenant can select the "Leave Review" button to leave a review for the corresponding rental Owner. c. Tenant uses the included use case SubmitReview to submit the review.
Alternative Flows:	None
Exceptions:	None
Includes:	1. SubmitReview
Special Requirements:	None
Assumptions:	None

Notes and Issues:	None
-------------------	------

III.V SubmitReview

Use Case ID:	#3-5		
Use Case Name:	SubmitReview		
Created By:	Lim Jing Rong	Last Updated By:	Goh Shuen Wei
Date Created:	24th August 2024	Date Last Updated:	12th September 2024

Actor:	Tenant/Owner
Description:	Tenant/Owner can submit a review of a Owner/Tenant.
Preconditions:	Tenant/Owner is logged in and is authenticated
Postconditions:	Review is successfully submitted
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none">1. The System will prompt the Tenant/Owner to enter a review title, text and a rating out of 5 stars.2. Tenant/Owner confirms the review by selecting the "Submit/Update Review" button.3. The System will save the review in the database and display that the review is submitted.
Alternative Flows:	<u>AF-S2: The Tenant/Owner selects the "Back" icon.</u> <ol style="list-style-type: none">1. The system will not save the review.
Exceptions:	<ol style="list-style-type: none">1. If any input field is left empty, display an error message.
Includes:	<ol style="list-style-type: none">1. FlagReview2. EditReview3. DeleteReview
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

III.VI FlagReview

Use Case ID:	#3-6		
Use Case Name:	FlagReview		
Created By:	Lim Jing Rong	Last Updated By:	Goh Shuen Wei
Date Created:	24th August 2024	Date Last Updated:	12th September 2024

Actor:	Tenant/Owner
Description:	Tenant/Owner can flag the review if it is inappropriate.
Preconditions:	Tenant/Owner is logged in and is authenticated.
Postconditions:	Flag is sent to the admins to process the disputes.
Priority:	Medium
Frequency of Use:	Low
Flow of Events:	<ol style="list-style-type: none">1. If the Tenant/Owner deems that the review is inappropriate, they can submit a request to flag the review to the Admin by selecting the “Flag” icon.2. System will display a confirmation message. The Tenant/Owner can select CONFIRM to proceed with the flagging of the review.
Alternative Flows:	<u>AF-S2: If the user selects CANCEL</u> <ol style="list-style-type: none">1. If the user selects CANCEL, the system will cancel the operation and return the Tenant/Owner to the review.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

III.VII EditReview

Use Case ID:	#3-7		
Use Case Name:	EditReview		
Created By:	Lim Jing Rong	Last Updated By:	Goh Shuen Wei
Date Created:	24th August 2024	Date Last Updated:	12th September 2024

Actor:	Tenant/Owner
Description:	Tenant/Owner can edit his/her review by uploading pictures, removing uploaded pictures or editing the review text.
Preconditions:	Tenant/Owner is logged in and is authenticated.
Postconditions:	The Review is successfully edited.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. Tenant/Owner navigates to the "Rental Information" screen. 2. For Tenant, the system will display a "Leave Review" button. For Owner, the system will display a "Leave Review" button after they select a specific Tenant. 3. Upon selecting the "Leave Review" button, Tenant/Owner will be brought to the review page. 4. The system will display their prior review title, text and rating, where they are able to make changes. 5. Tenant/Owner can select the "Submit/Update Review" button to save the changes made. 6. Tenant/Owner can select the "Delete Review" button to delete their review. The use case DeleteReview will be used.
Alternative Flows:	<u>AF-S4: If the "Back" icon is selected</u> <ol style="list-style-type: none"> 1. The system will not save the changes made. 2. The system will redirect the Tenant/Owner away from the Edit Review Screen.
Exceptions:	<ol style="list-style-type: none"> 1. If any input field is left empty, display an error message.
Includes:	None
Special Requirements:	None
Assumptions:	None

Notes and Issues:	None
-------------------	------

III.VIII DeleteReview

Use Case ID:	#3-8		
Use Case Name:	DeleteReview		
Created By:	Lim Jing Rong	Last Updated By:	Goh Shuen Wei
Date Created:	24th August 2024	Date Last Updated:	12th September 2024

Actor:	Tenant/Owner
Description:	Tenant/Owner can delete his/her review
Preconditions:	Tenant/Owner is logged in and is authenticated
Postconditions:	The Review has successfully been deleted
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. If the Tenant/Owner has already made a review, the system will display the "Delete Review" button beside the review. 2. When the Tenant/Owner selects on the "Delete Review" button, the System will display a confirmation message. The Tenant/Owner can select CONFIRM DELETE to proceed with the deletion. 3. When the Tenant/Owner selects CONFIRM DELETE, the system will delete this review and remove this review from the database.
Alternative Flows:	<u>AF-S2: If the Tenant/Owner selects CANCEL</u> <ol style="list-style-type: none"> 2. If the Tenant/Owner selects CANCEL, the system will cancel the operation and return the Tenant/Owner to the review.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Functional Requirement #4

IV.I CreateAccount

Use Case ID:	#4-1		
Use Case Name:	CreateAccount		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	User
Description:	Allows a person to create an Account in HomeGoWhere to become a user.
Preconditions:	None
Postconditions:	An Account with a specific role (Admin, Owner or Tenant) is created for the User.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The system prompts the User to enter their name, address, password, email, contact number, profile picture, and role.2. The User enters the required information and selects the "Create Account" button to confirm his/her inputs.3. An Account with a specific role (Admin, Owner or Tenant) is created for the User.
Alternative Flows:	<u>AF-S2: Returning to Login page</u> <ol style="list-style-type: none">1. If the User selects the back navigation icon, the system will navigate the User to the login page.
Exceptions:	<ol style="list-style-type: none">1. If any required information is missing or invalid, an error message is displayed.
Includes:	None
Special Requirements:	System needs to validate user input data.
Assumptions:	None
Notes and Issues:	None

IV.II Login

Use Case ID:	#4-2		
Use Case Name:	Login		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	User
Description:	Allows User to login to his/her HomeGoWhere Account using his/her email and password.
Preconditions:	None
Postconditions:	User is logged into the HomeGoWhere application and is navigated to the home screen of the application.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The system allows the User to login with either (a) Email and password, or (b) SingPass. 2. The User chooses to login with email and password. 3. The User enters his/her email and password. The password is masked as dots, but the User can choose to unmask it by selecting the eye icon. 4. The User selects the "Login" button.
Alternative Flows:	<u>AF-S1: User chooses to login with SingPass</u> <ol style="list-style-type: none"> 1. The User selects the "Login with Singpass" button. 2. The User logs into the system using Singpass login credentials using the included use case SingpassLogin.
Exceptions:	<ol style="list-style-type: none"> 1. If any required information is missing or invalid, an error message is displayed. 2. If email and password do not match when the User tries to login in Step 4, HomeGoWhere shall display "Email and password do not match" to the user.
Includes:	SingpassLogin
Special Requirements:	System needs to validate user input data.
Assumptions:	The User has an existing HomeGoWhere Account.
Notes and Issues:	None

IV.III SearchAndViewListing

Use Case ID:	#4-3		
Use Case Name:	SearchAndViewListing		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	User
Description:	Allows users to search for homes by name, type or location via a search bar and get a filtered list of houses.
Preconditions:	The User must be logged in and authenticated.
Postconditions:	A filtered list of homes is returned.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The User can search for homes according to their block, type or location via a search bar.2. The system will return a filtered list of homes for the User.
Alternative Flows:	None
Exceptions:	<ol style="list-style-type: none">1. If no home data matches the search criteria, the system shall display a message stating no property is found.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

IV.IV FlagListing

Use Case ID:	#4-4		
Use Case Name:	FlagListing		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

Actor:	User
Description:	Allows User to flag a listing for the Admin to process for deletion.
Preconditions:	<p>Entry Condition: Called by the use case SearchAndViewListing when the "Flag Listing" function is selected.</p> <p>The User must be logged in and authenticated.</p>
Postconditions:	A listing is flagged.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The system displays an option to flag a listing for every listing that the User sees. 2. If the User flags the listing, the system will prompt the user to input the reason as to why they have flagged the listing. 3. If the user selects CONFIRM, the system will flag the listing and add it to the list of flagged listings for the Admins to process.
Alternative Flows:	<p><u>AF-S3: If the user selects CANCEL</u></p> <ol style="list-style-type: none"> 1. If the user selects CANCEL, the system will cancel the operation and return the User back to the listings.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

IV.V Chat

Use Case ID:	#4-5		
Use Case Name:	Chat		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	1st September 2024

Actor:	User
Description:	Allows Users to communicate with Owners via text in the in-built Chat function.
Preconditions:	The User must be logged in and authenticated.
Postconditions:	A chat is initiated with the Owner.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. User navigates to the desired listing screen.2. User selects the Chat icon to initiate a conversation with the Owner via text.
Alternative Flows:	<p><u>AF-S2: If the user selects the BACK navigation icon</u></p> <ol style="list-style-type: none">1. If the user selects BACK, the system will cancel the operation and return the User back to the listings. <p><u>AF-S2: If the user selects the STAR ratings icon</u></p> <ol style="list-style-type: none">1. If the user selects the STAR ratings, the system will navigate the user to a page with a list of reviews for that person.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

IV.VI FlagUser

Use Case ID:	#4-6		
Use Case Name:	FlagUser		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	28th August 2024	Date Last Updated:	28th August 2024

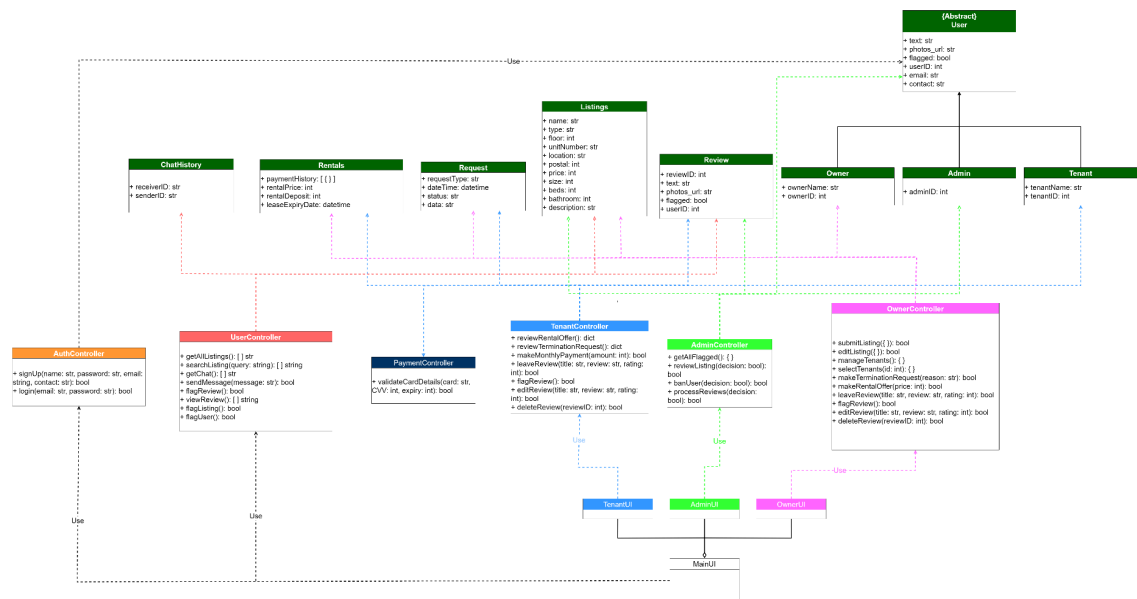
Actor:	User
Description:	Allows users to flag either party .
Preconditions:	Entry Condition: Called by the use case Chat when the "Flag User" function is selected. The User must be logged in and authenticated.
Postconditions:	User is flagged.
Priority:	Low
Frequency of Use:	Medium
Flow of Events:	1. When either party has behaved inappropriately or in an offensive manner, either party shall flag the User and add the User to the list of flagged users.
Alternative Flows:	None
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

IV.VII ViewReviews

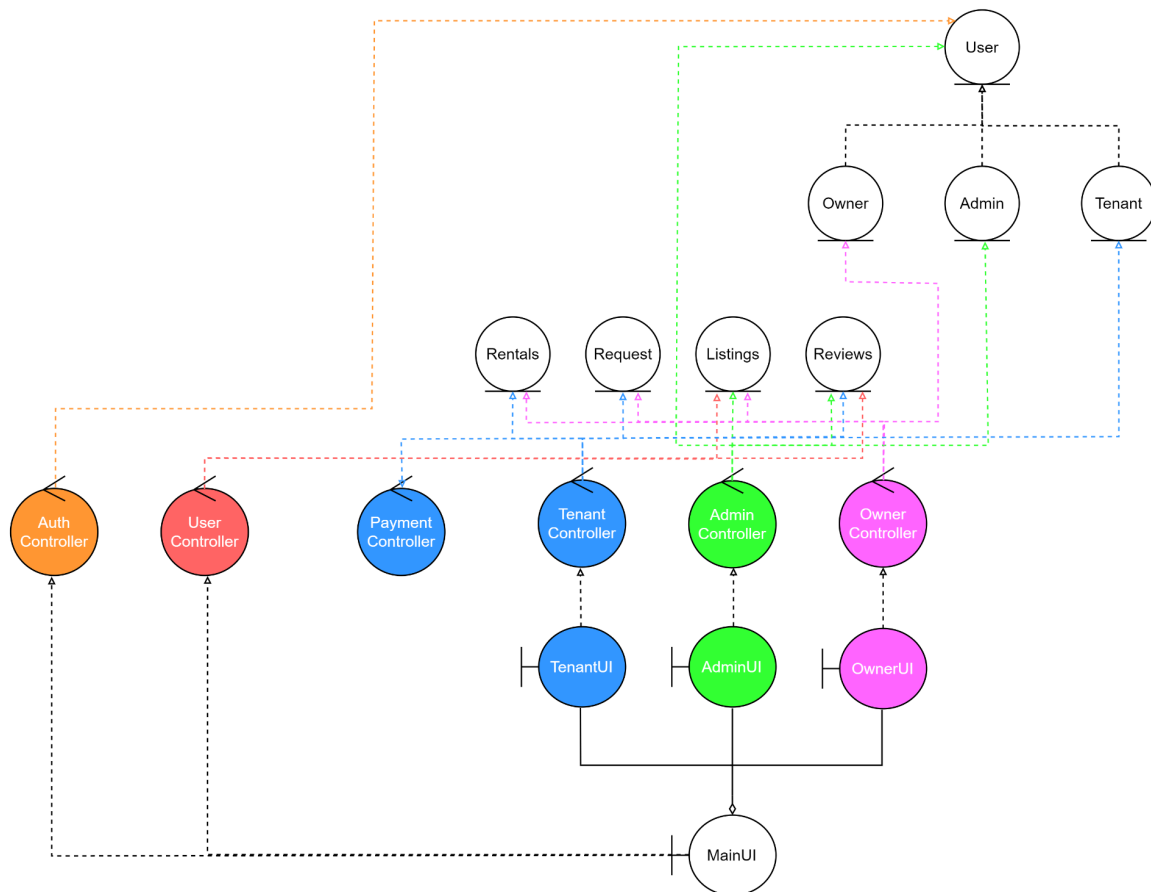
Use Case ID:	#4-6		
Use Case Name:	ViewReviews		
Created By:	Lau Zhan You	Last Updated By:	Lau Zhan You
Date Created:	1st September 2024	Date Last Updated:	1st September 2024

Actor:	User
Description:	Allows Users to view the ratings and reviews that other Users have given a particular person.
Preconditions:	<ol style="list-style-type: none">1. The User must be logged in and authenticated.2. User has selected the STAR ratings icon.
Postconditions:	Successfully display all ratings and reviews of the specified user.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The system will display the most updated list of reviews and star ratings that have been given to a specified user by other users.
Alternative Flows:	<u>AF-S1: If the user selects the BACK navigation icon</u> <ol style="list-style-type: none">1. If the user selects BACK, the system will cancel the operation and return the User back to the chat.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

2. Key Classes Diagram



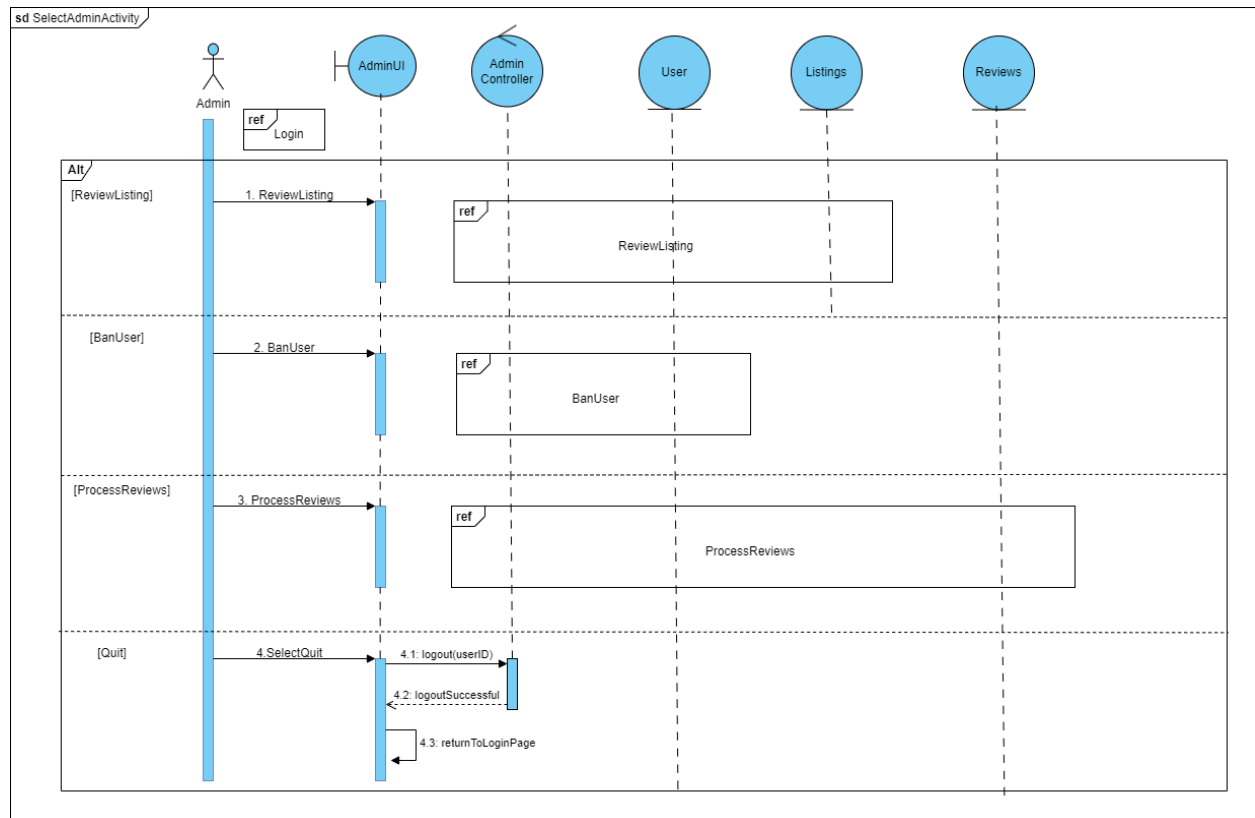
3. Entity, Control and Boundary Class Diagram



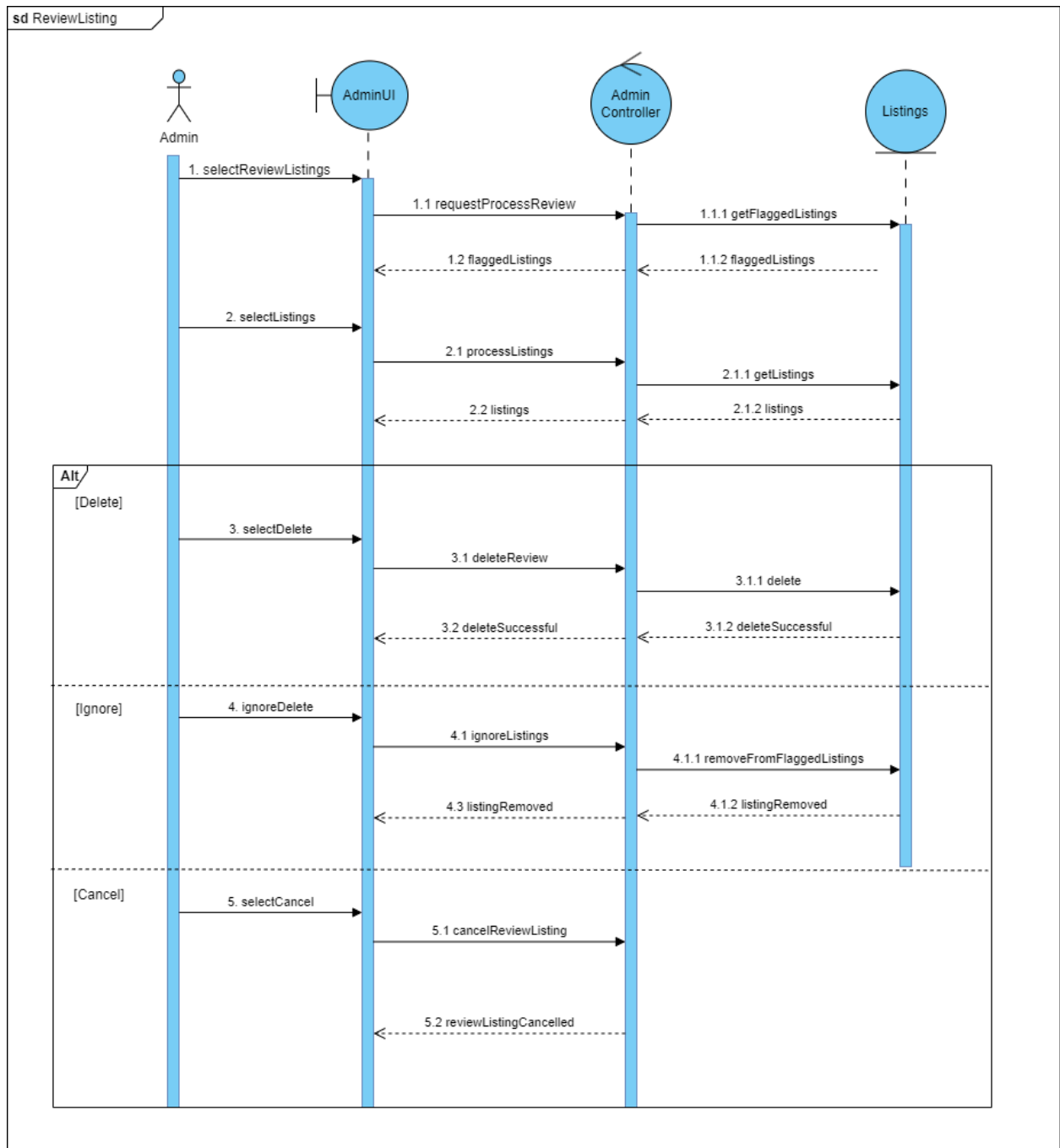
4. Sequence Diagrams of Use Cases

I. For Use Cases under I

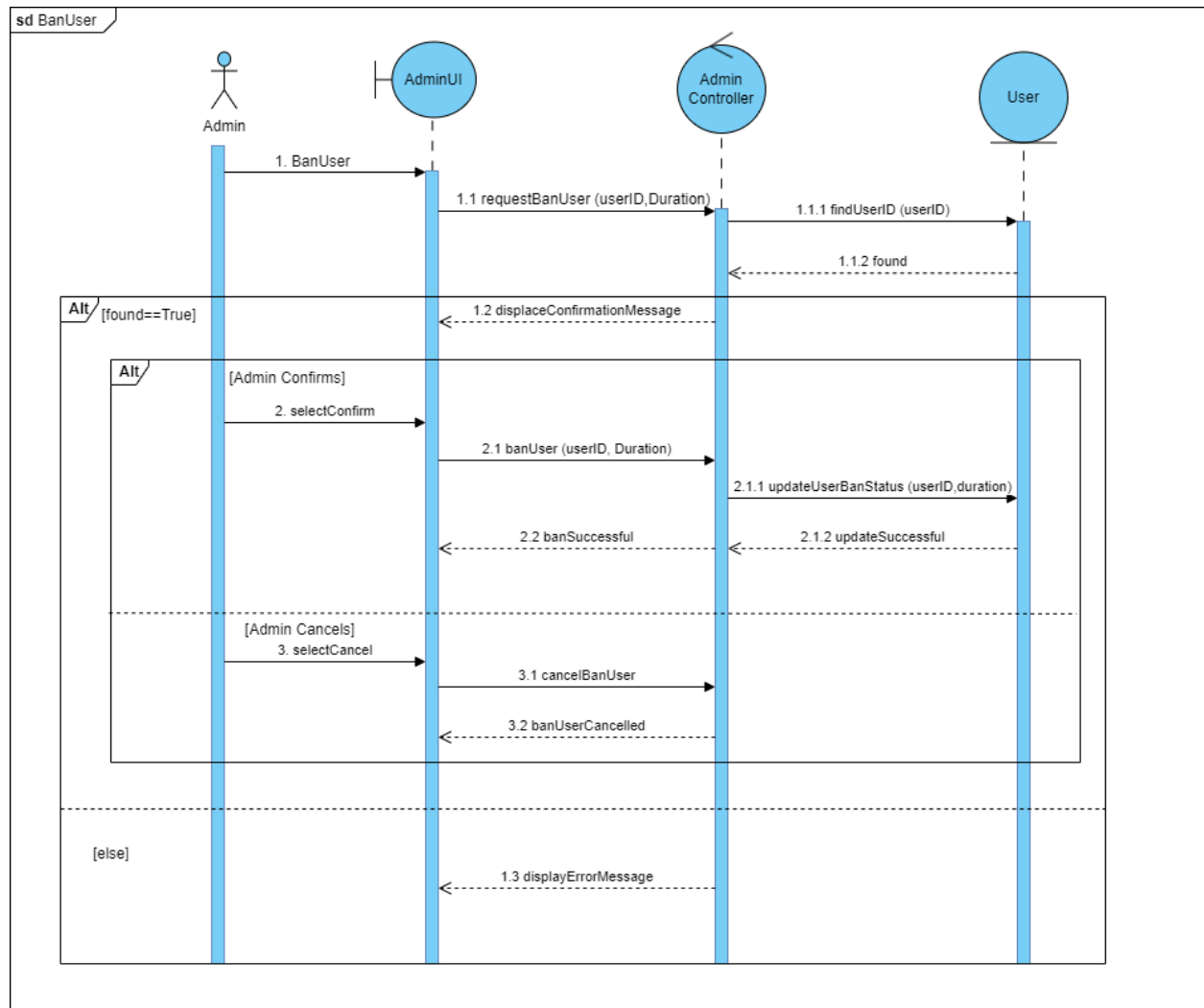
I.I SelectAdminActivity



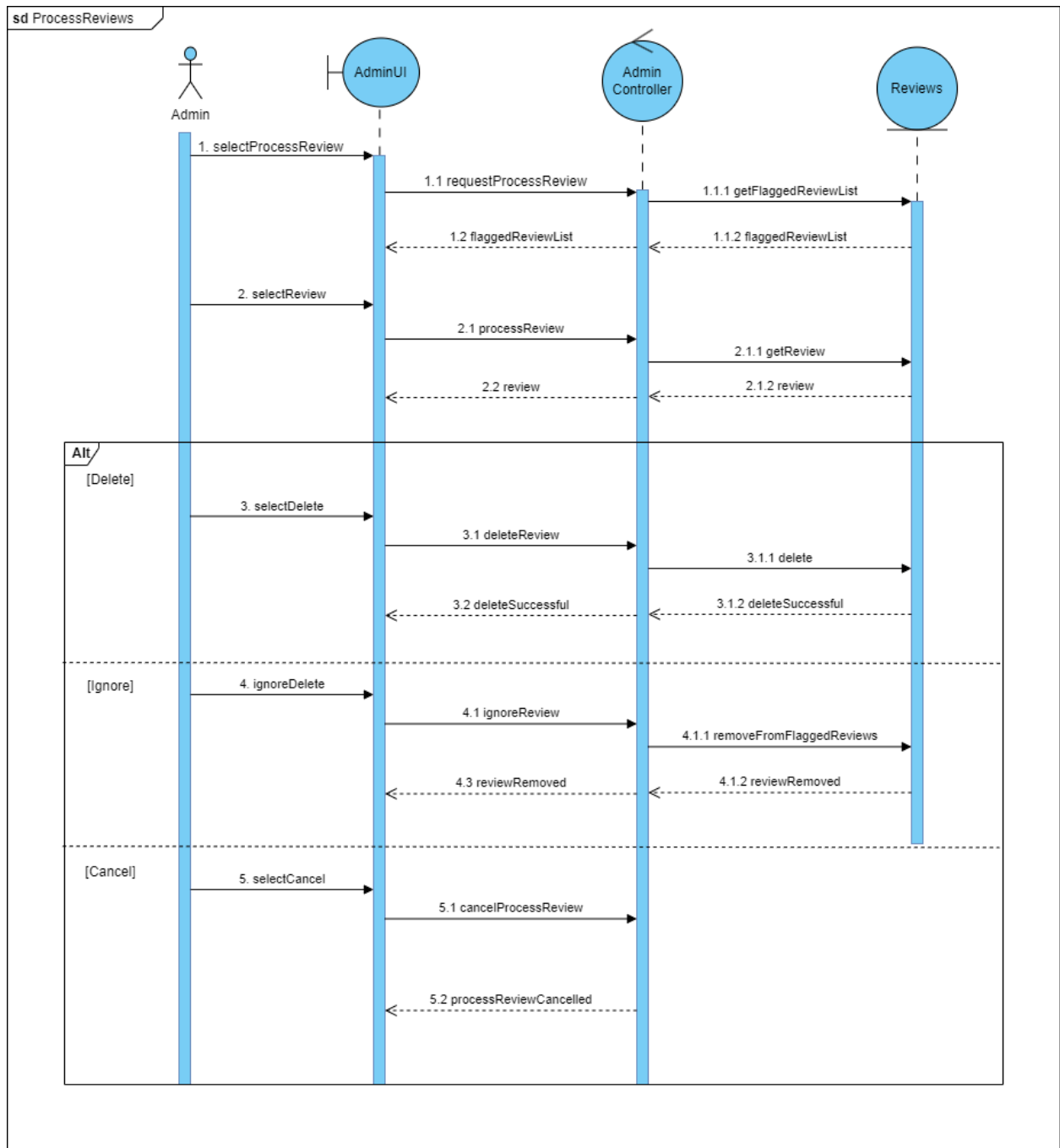
I.II ReviewListing



I.III BanUser

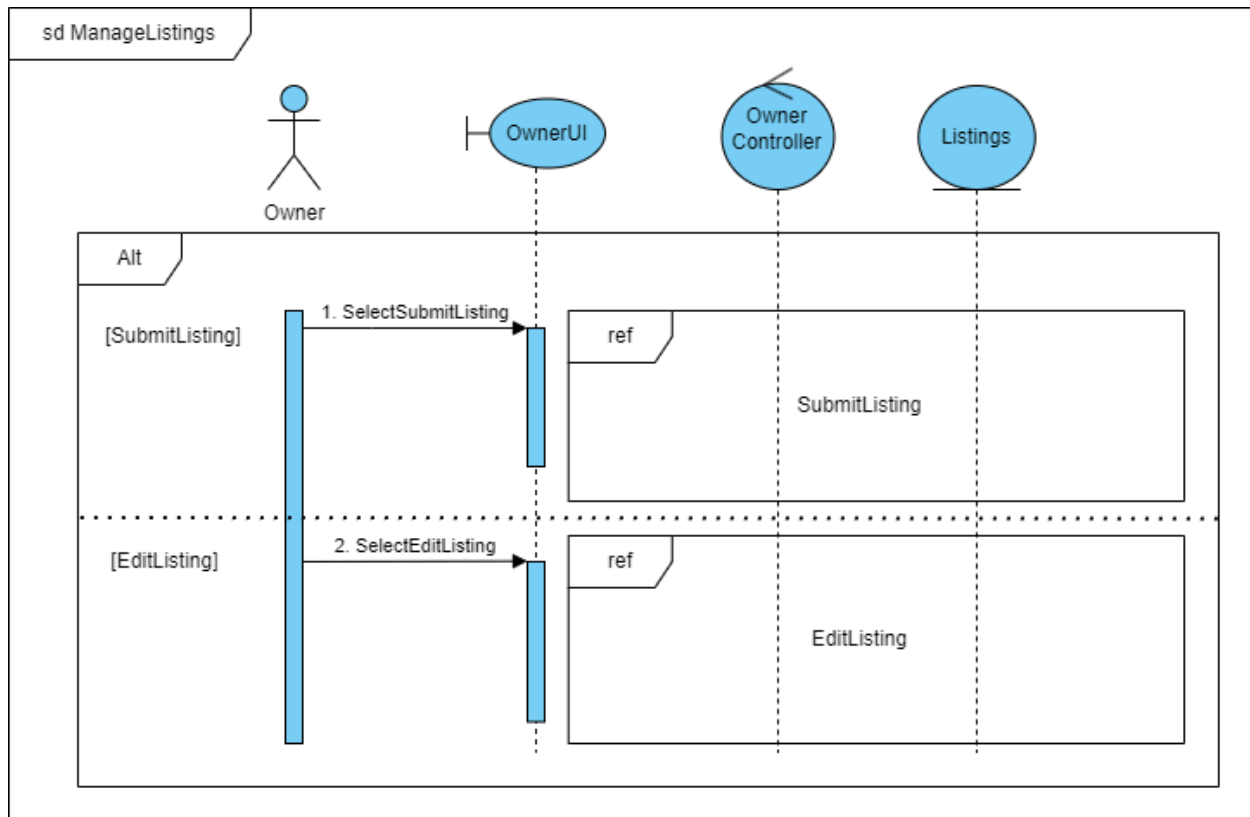


I.IV ProcessReviews

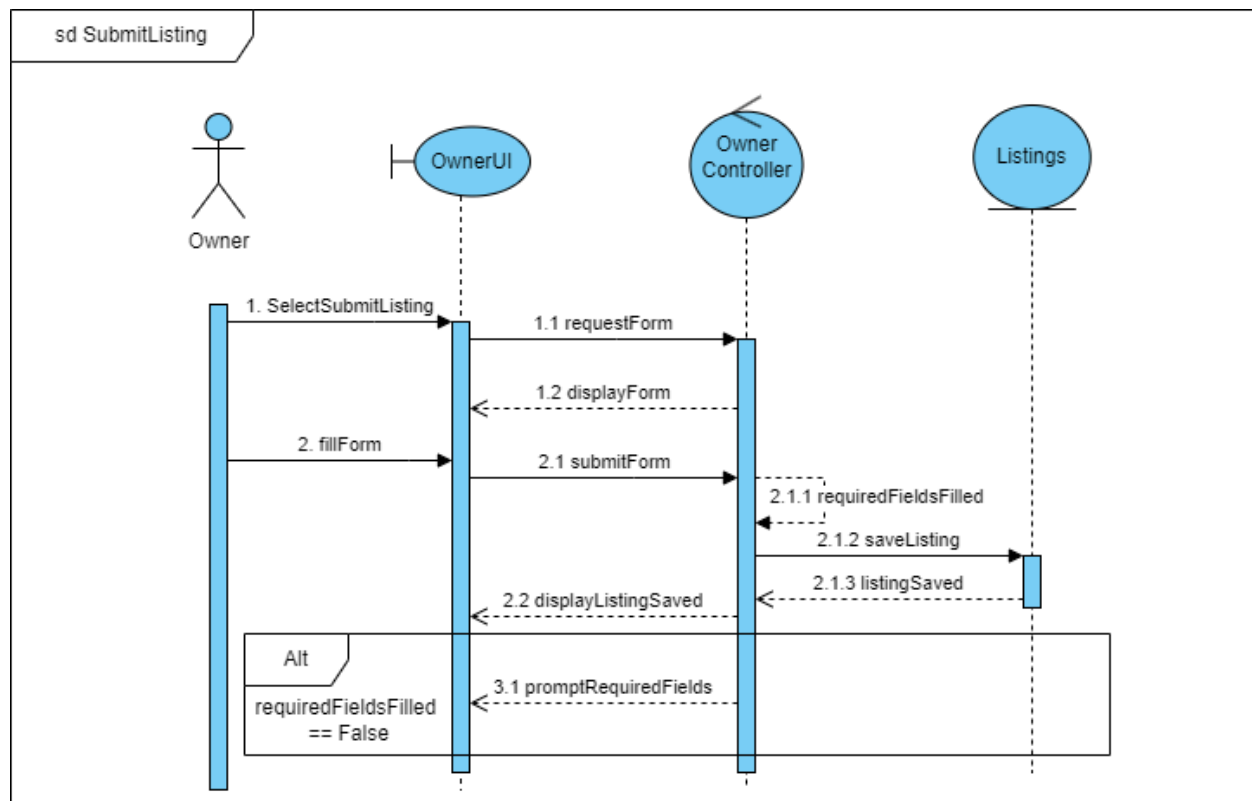


II. For Use Cases under II

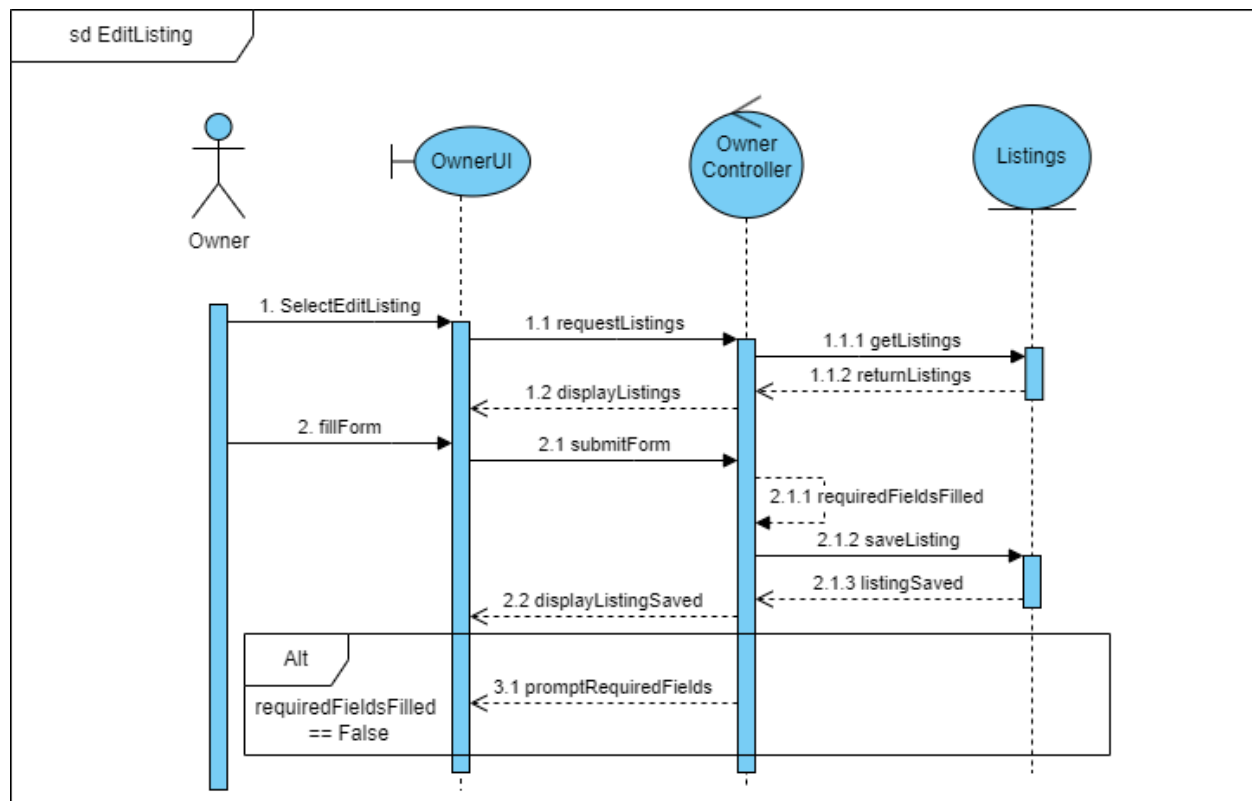
II.I ManageListings



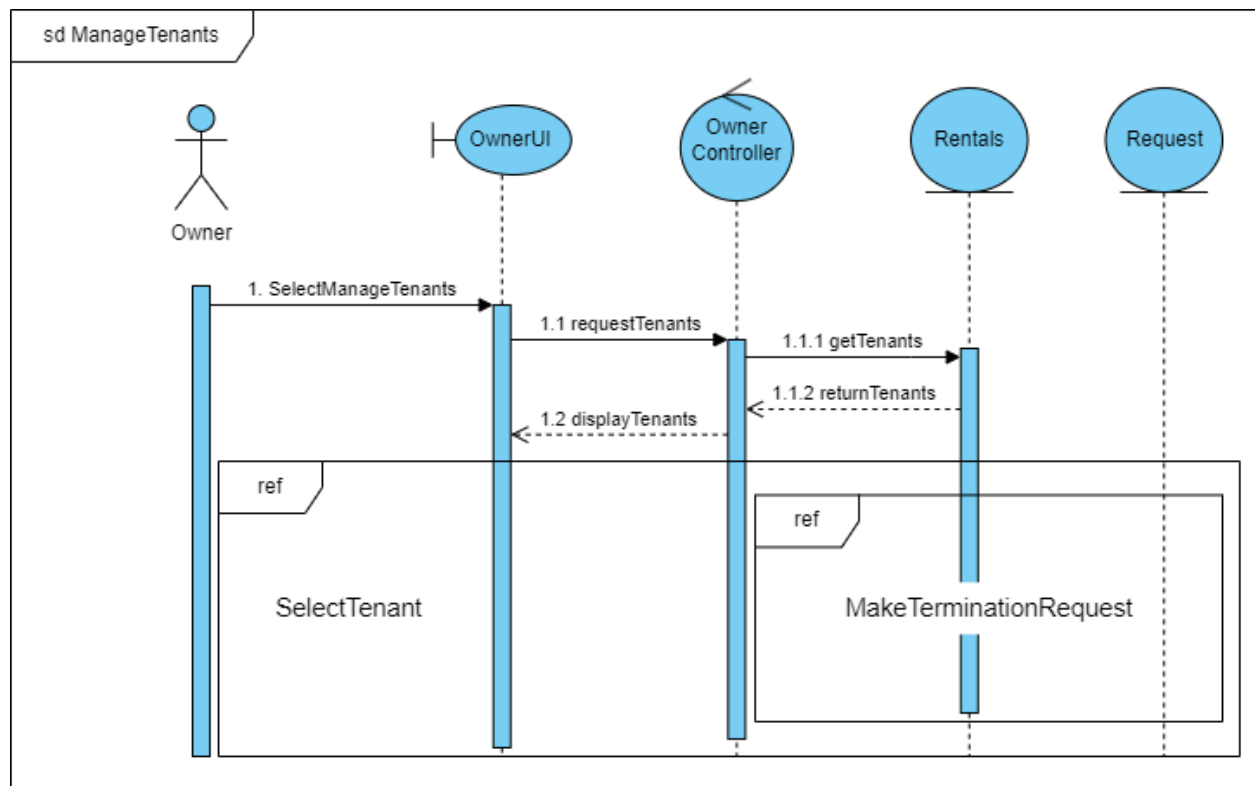
II.II SubmitListing



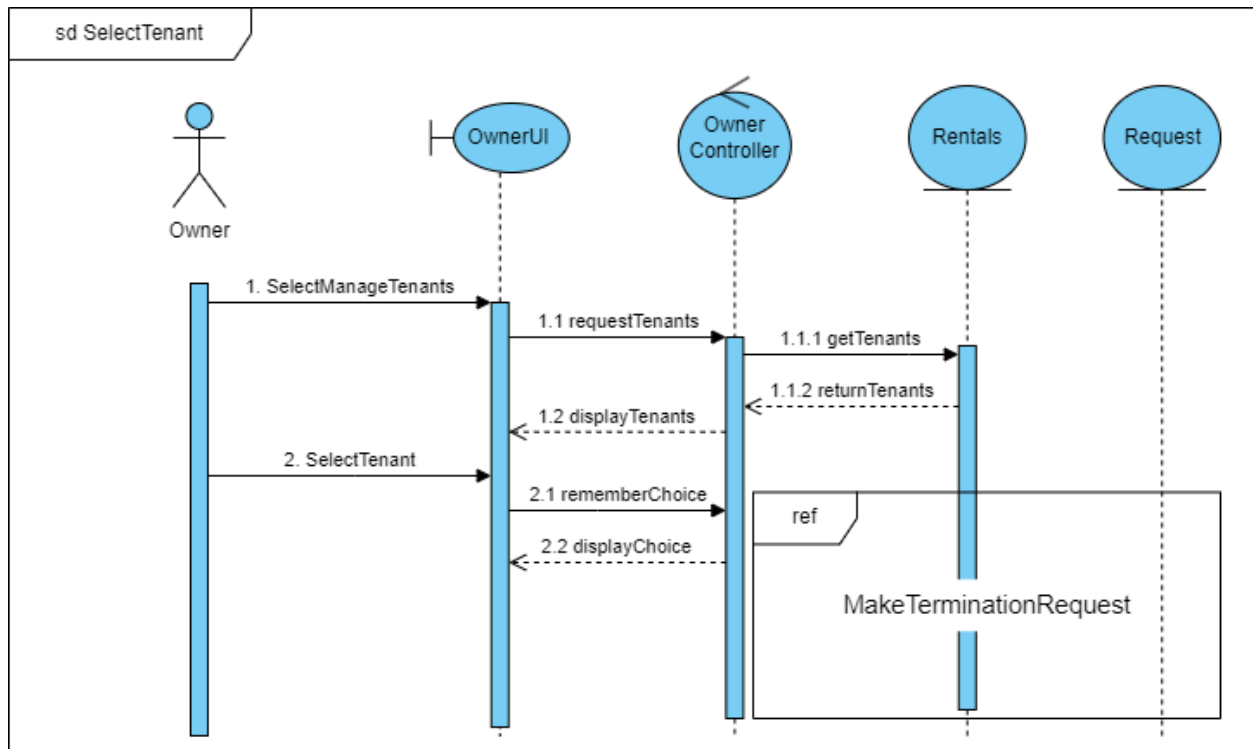
II.III EditListing



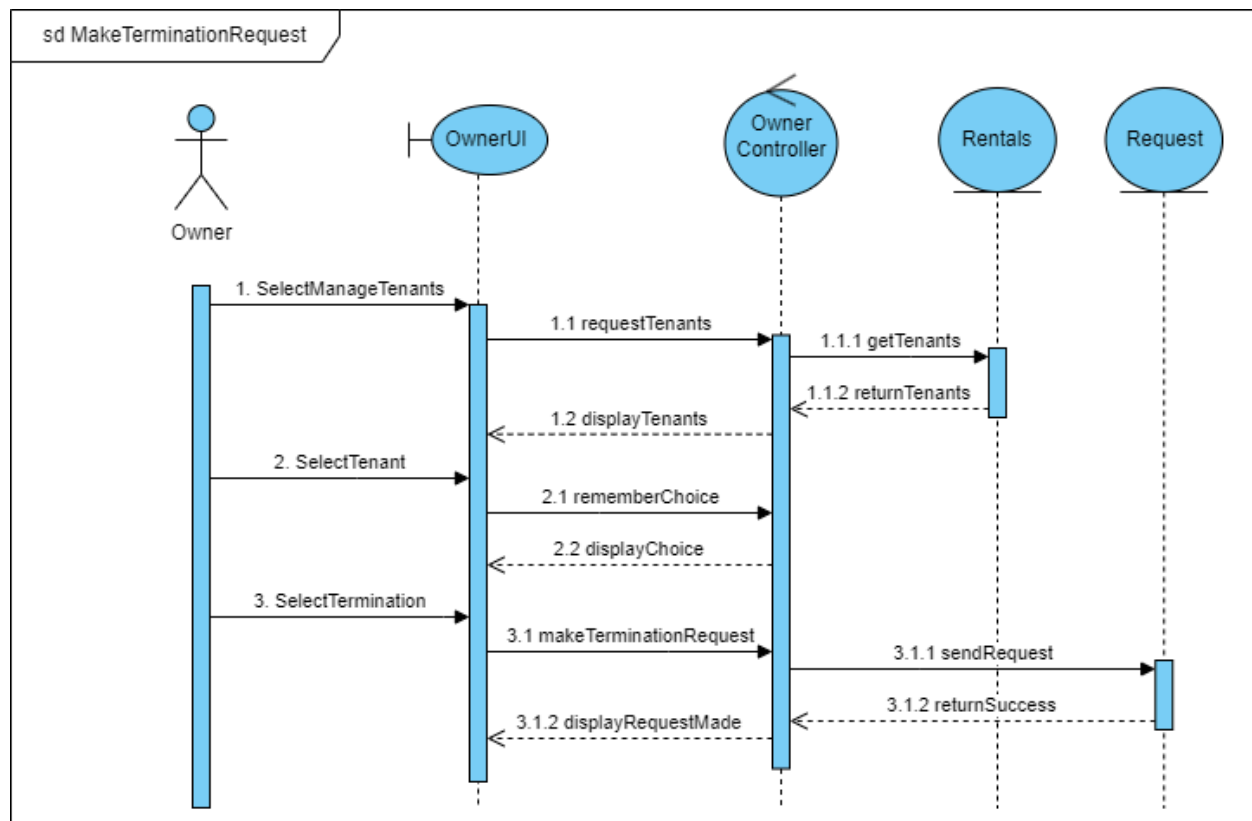
II.IV ManageTenants



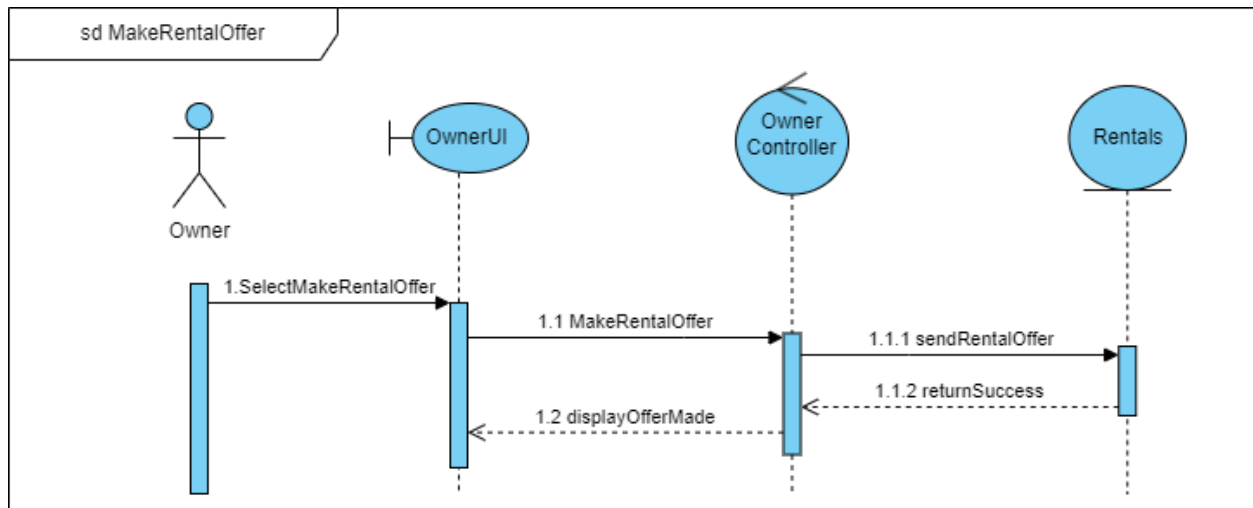
II.V SelectTenant



II.VI MakeTerminationRequest

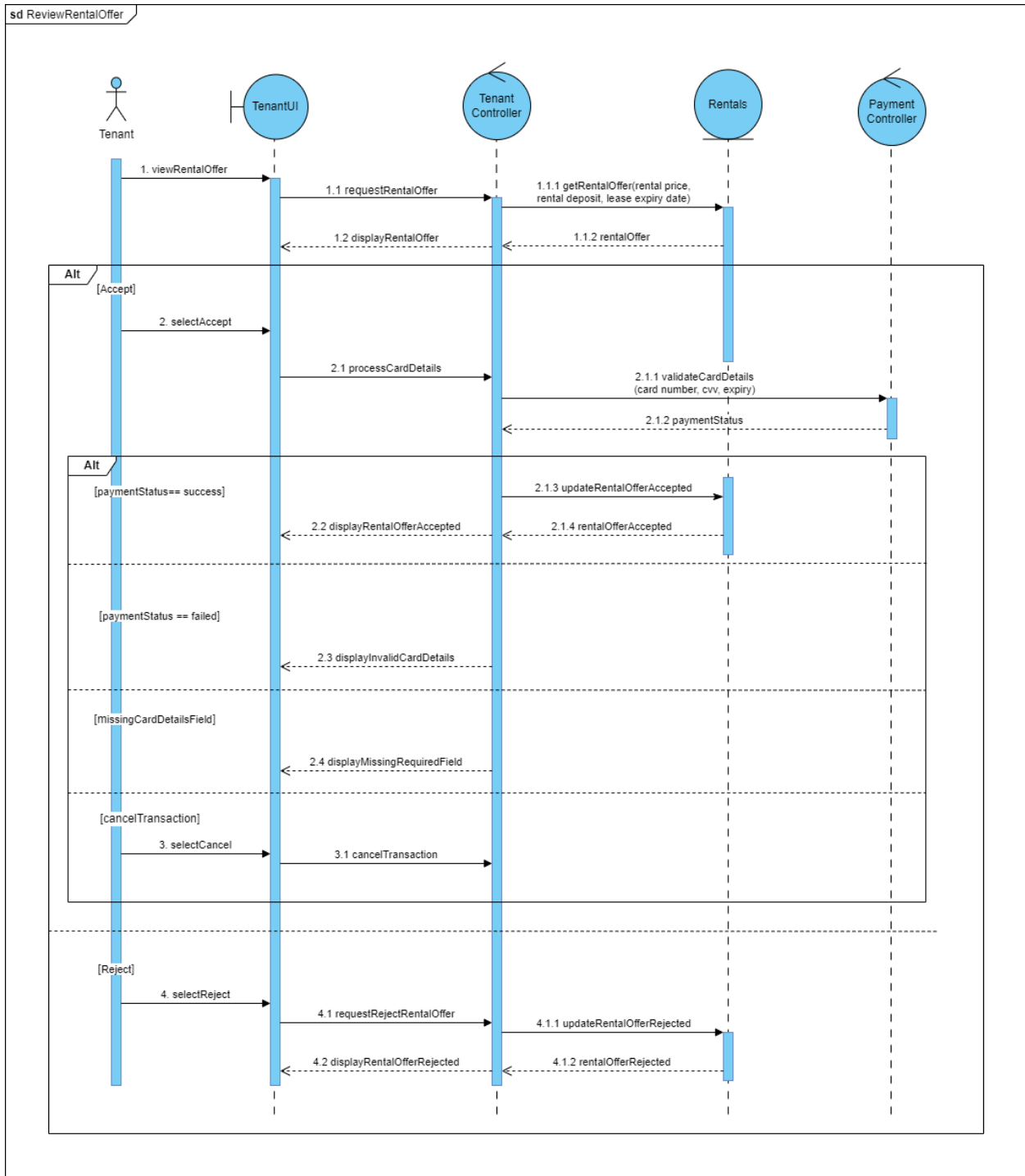


II.VII MakeRentalOffer

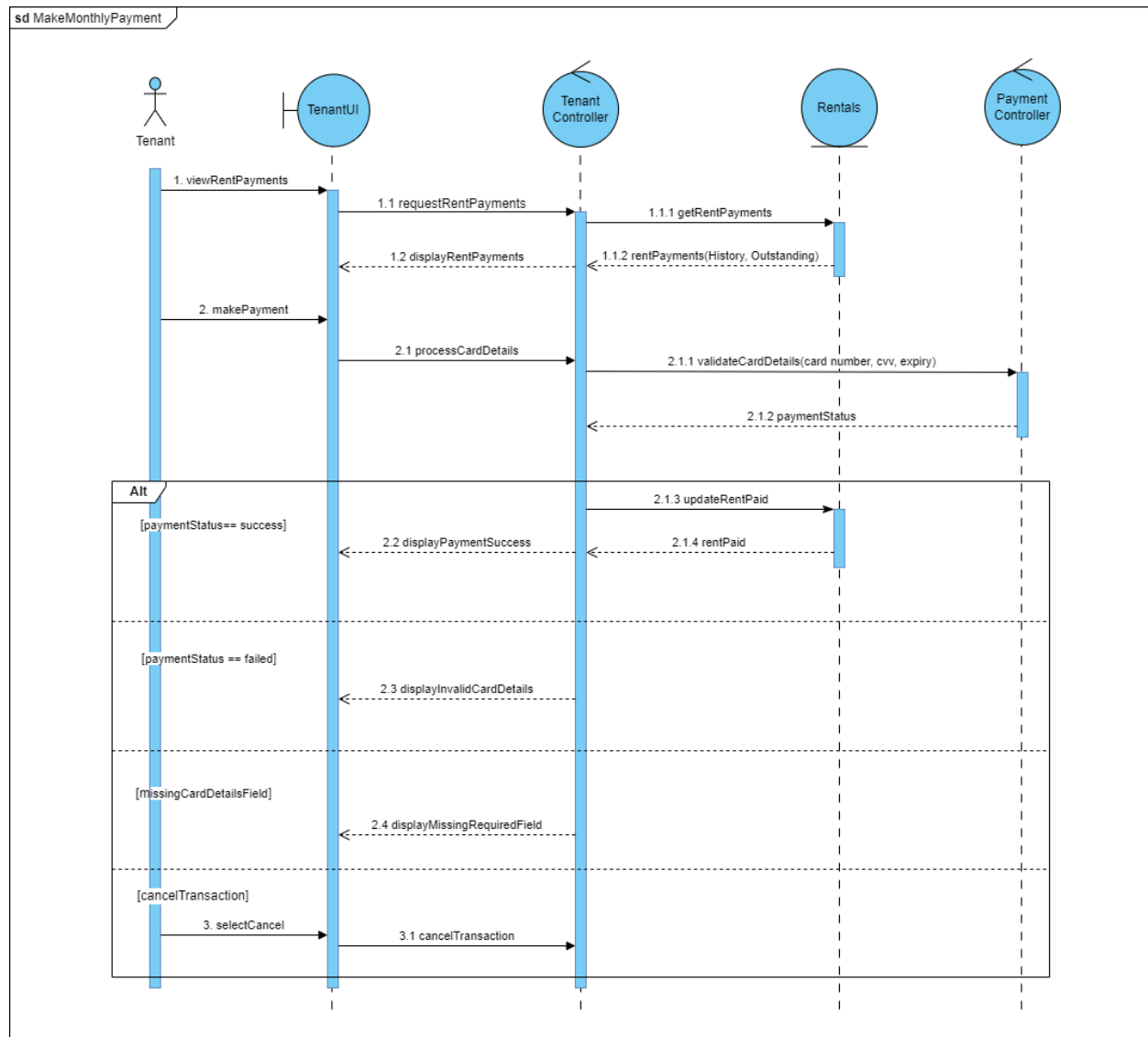


III. For Use Cases under III

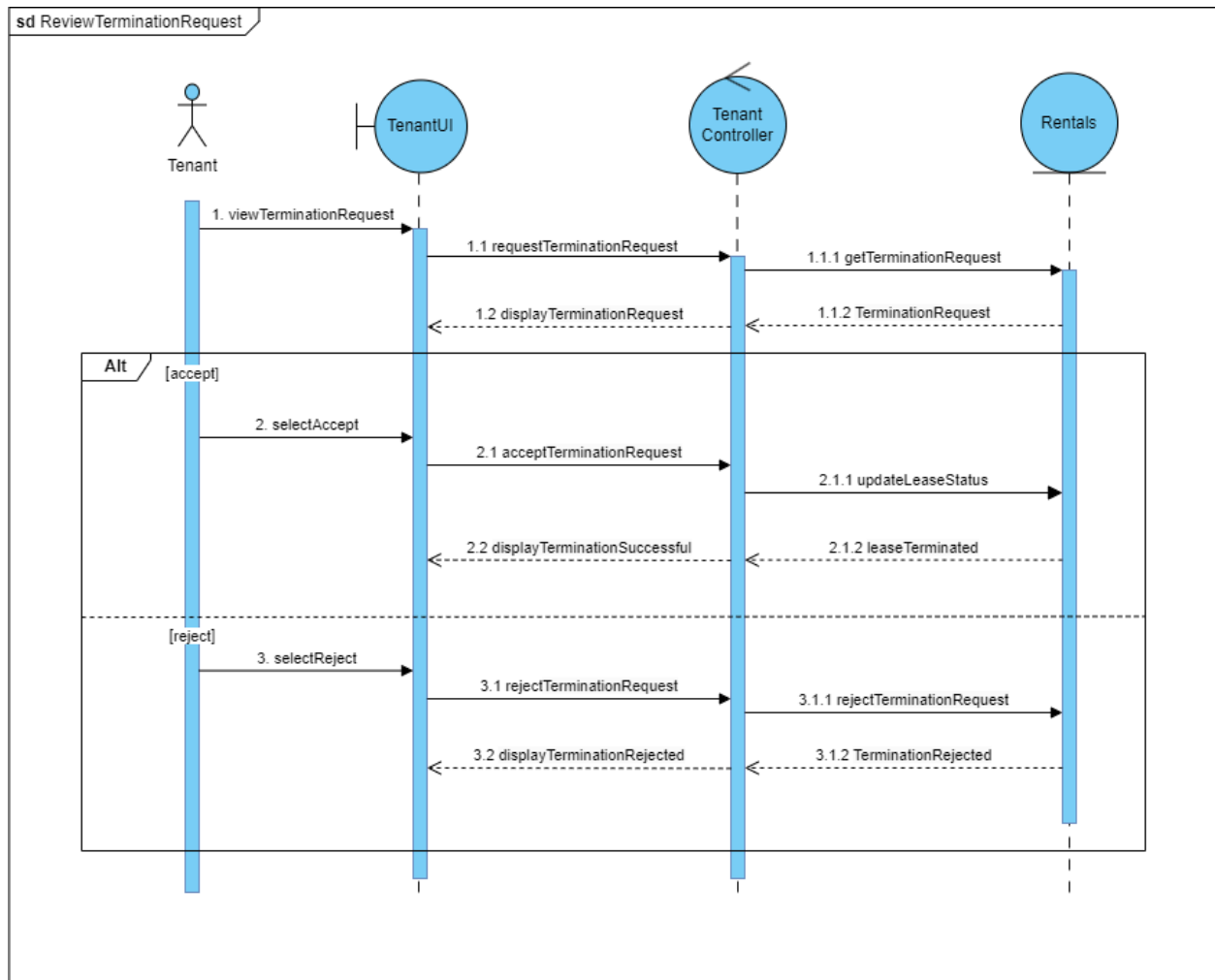
III.I ReviewRentalOffer



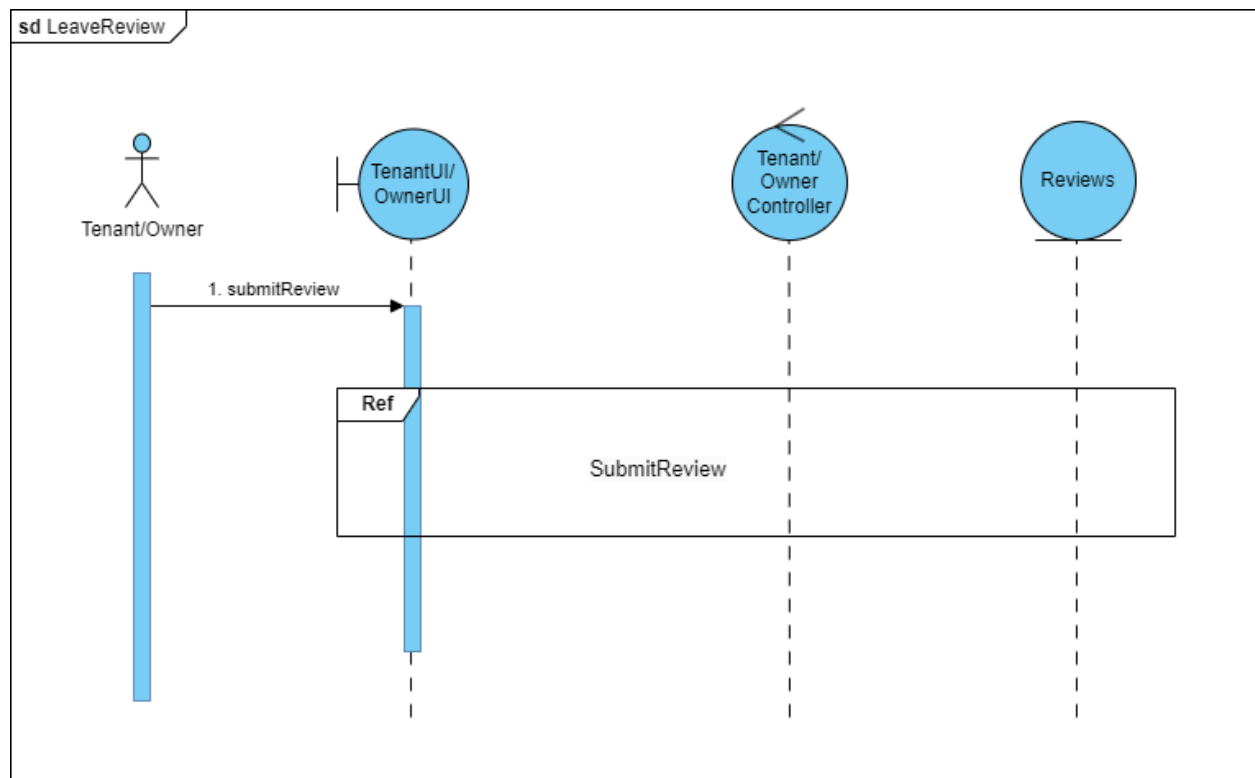
III.II MakeMonthlyPayment



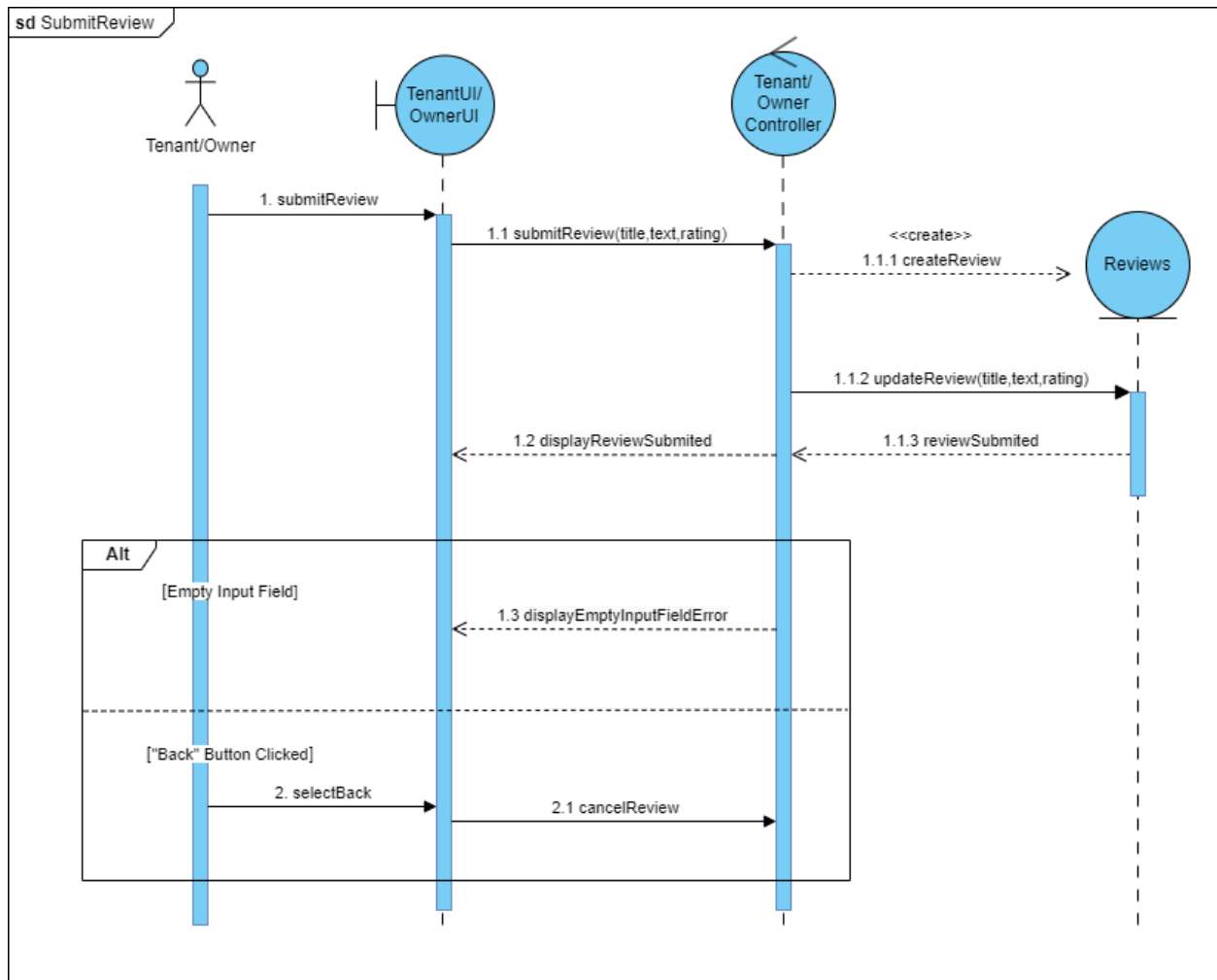
III.III ReviewTerminationRequest



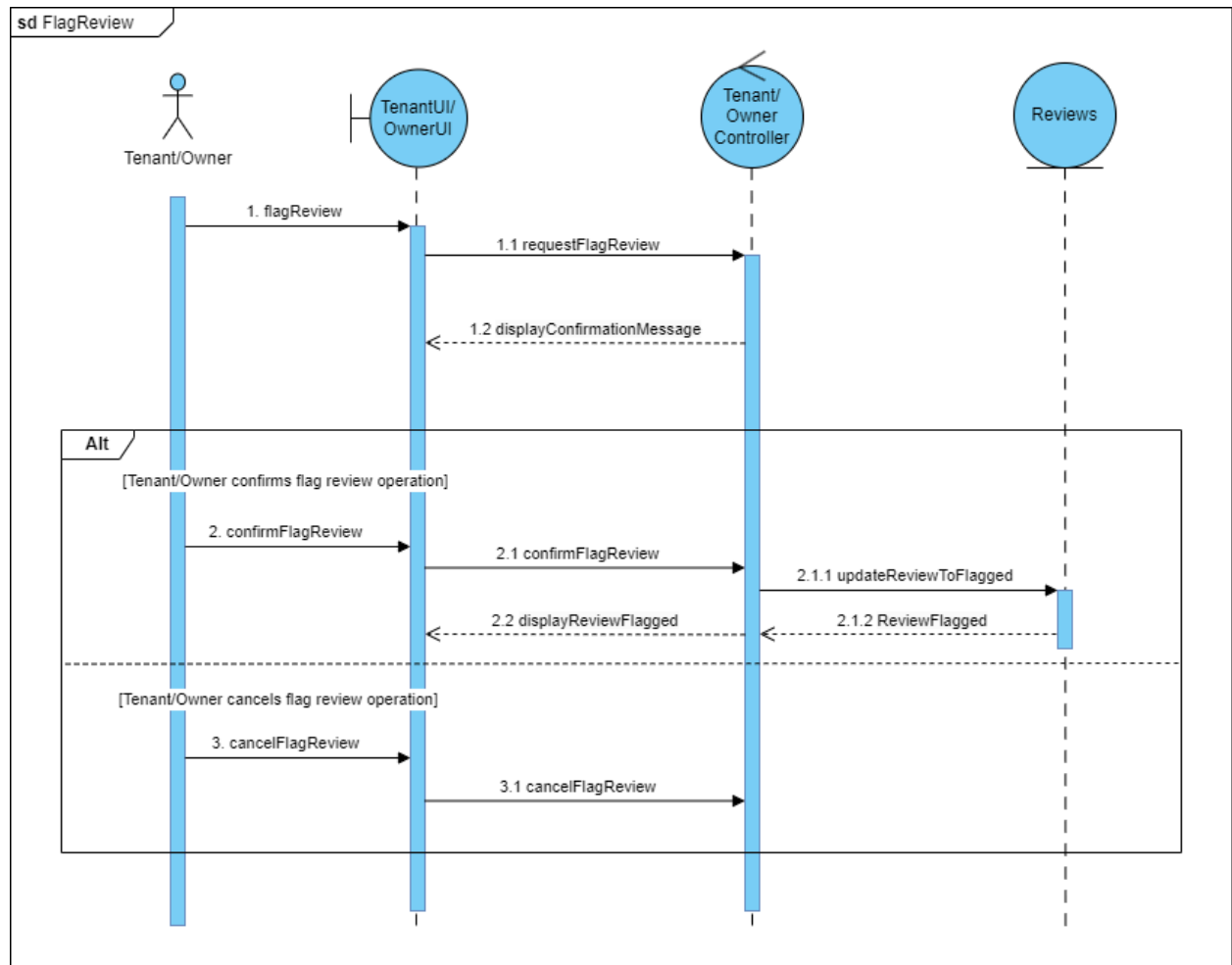
III.IV LeaveReview



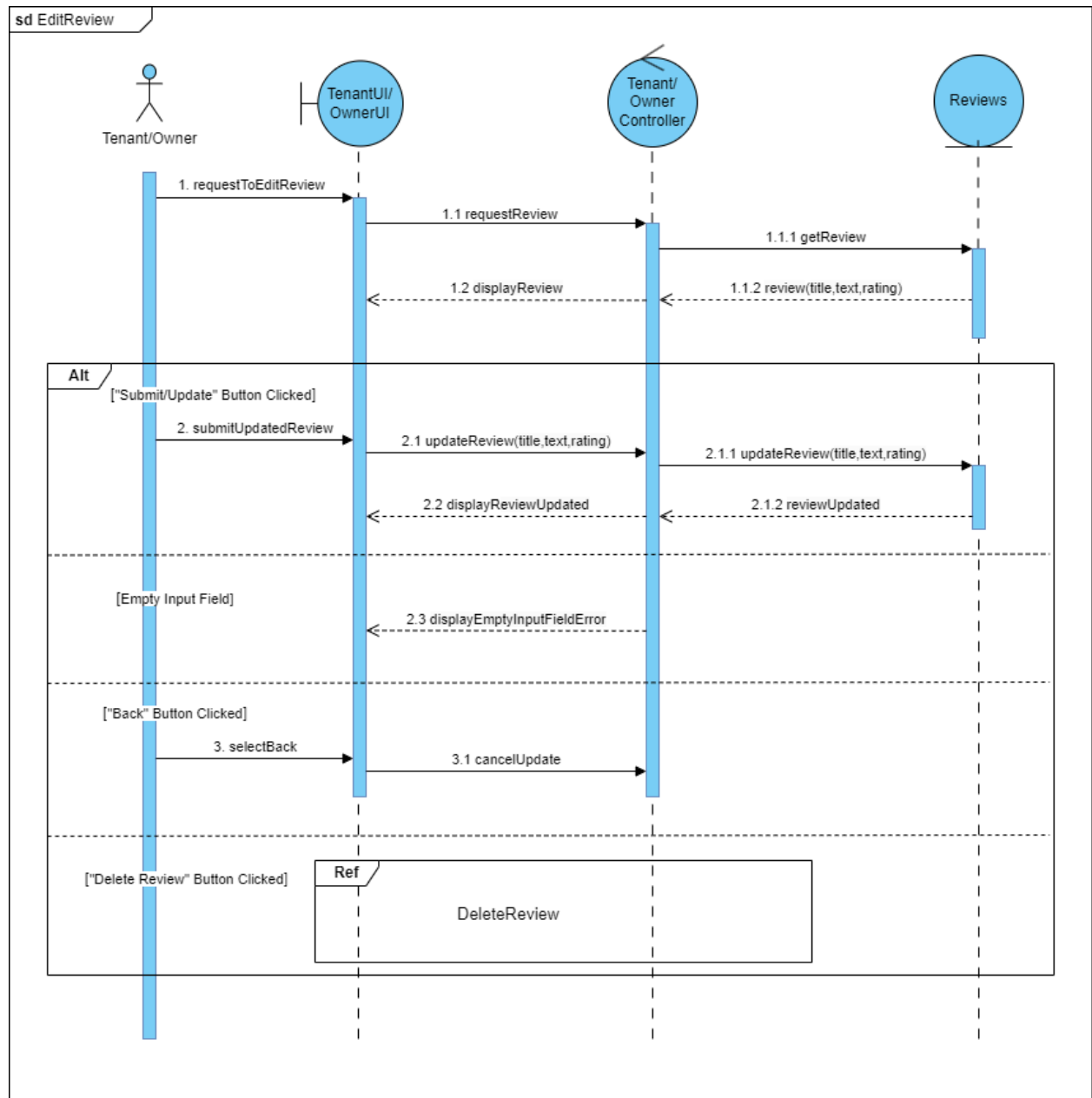
III.V SubmitReview



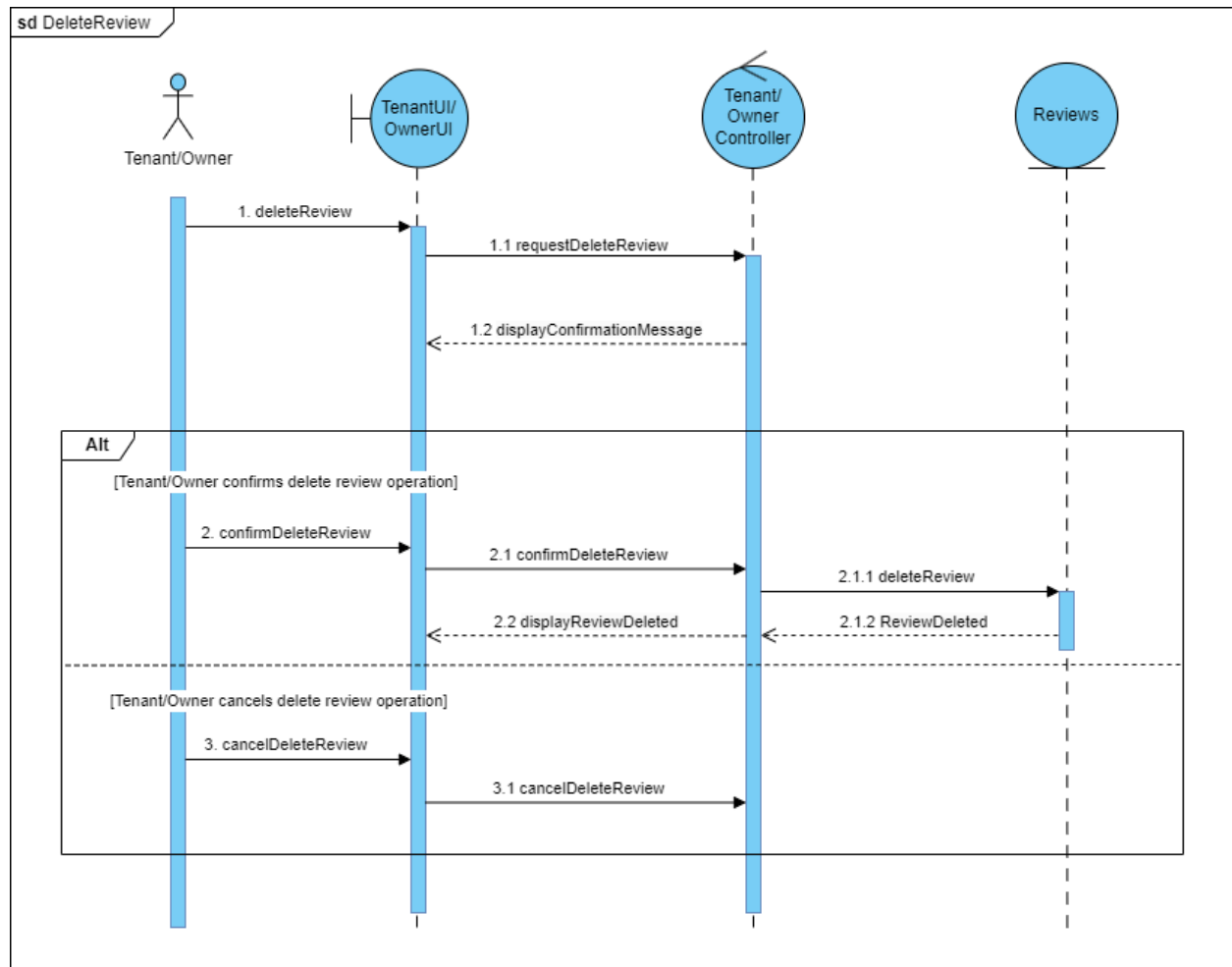
III.VI FlagReview



III.VII EditReview

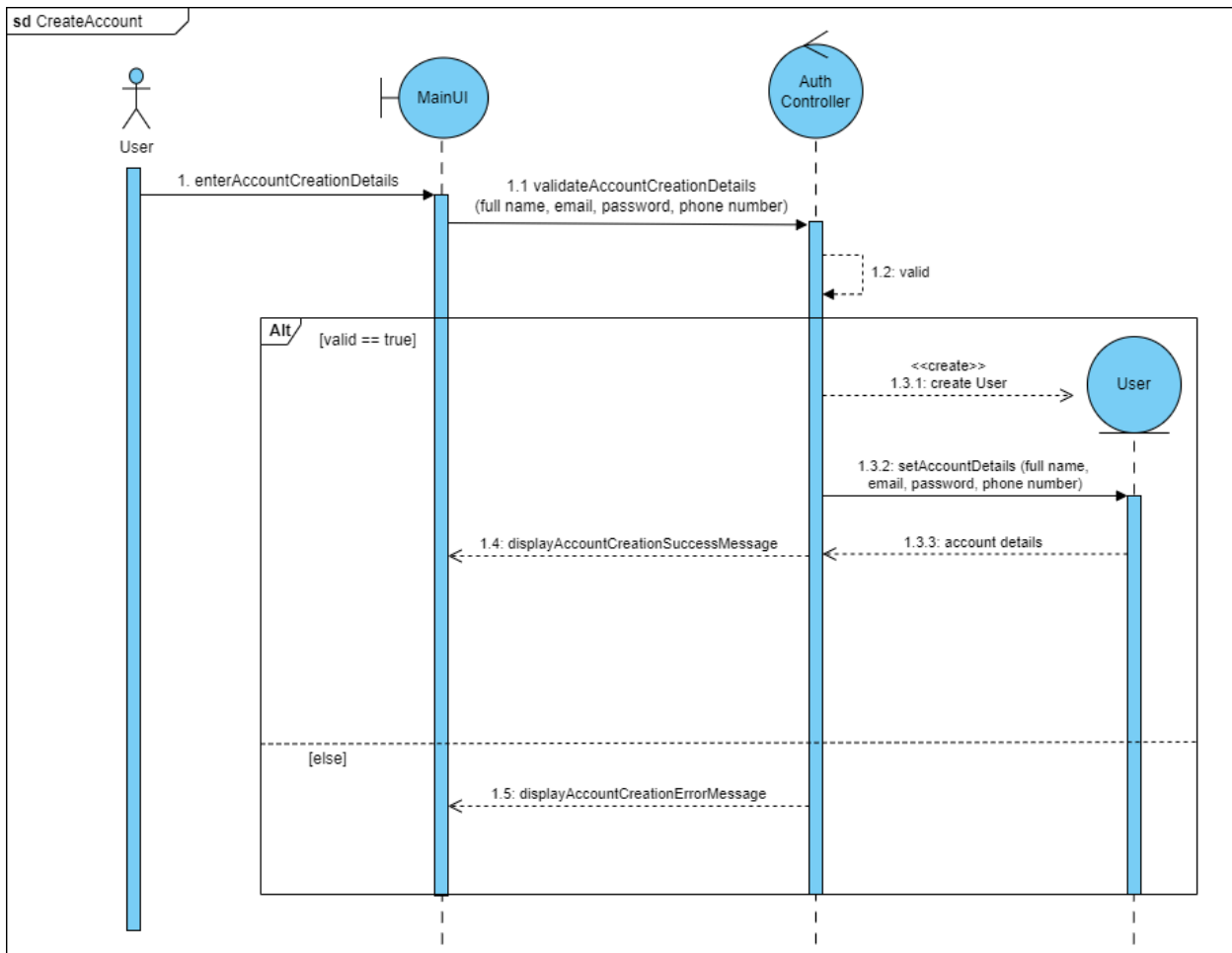


III.VIII DeleteReview

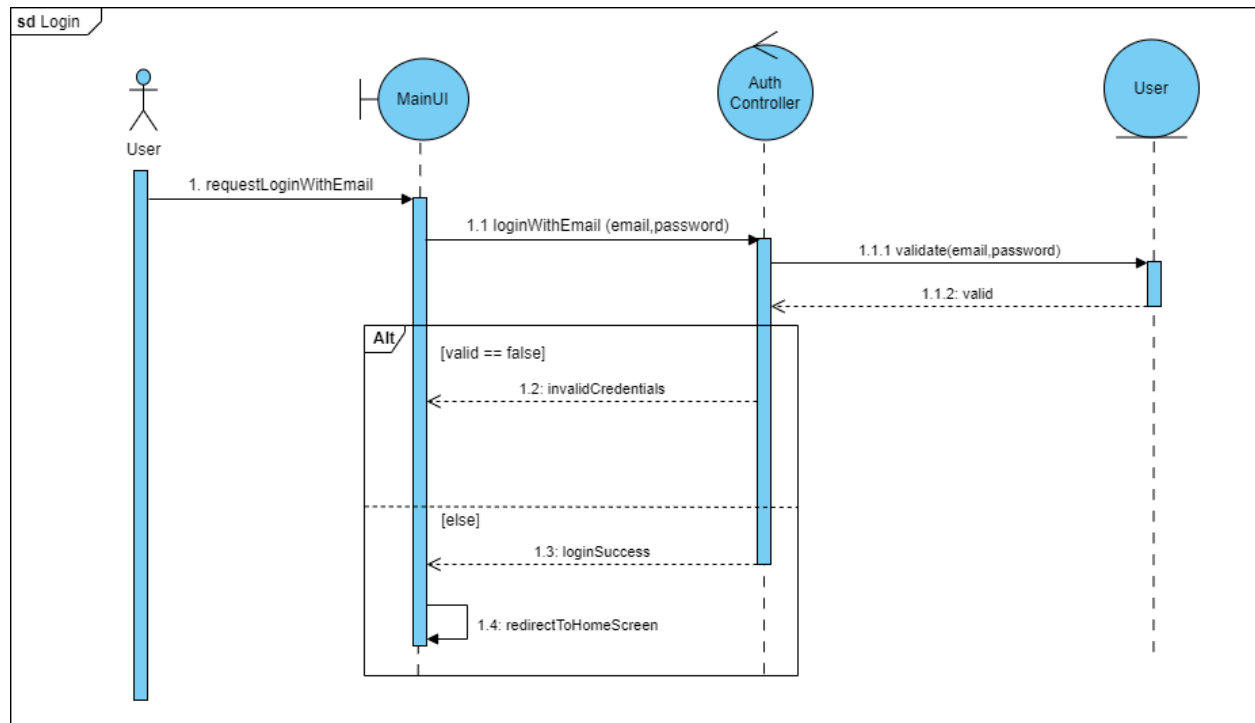


IV. For Use Cases under IV

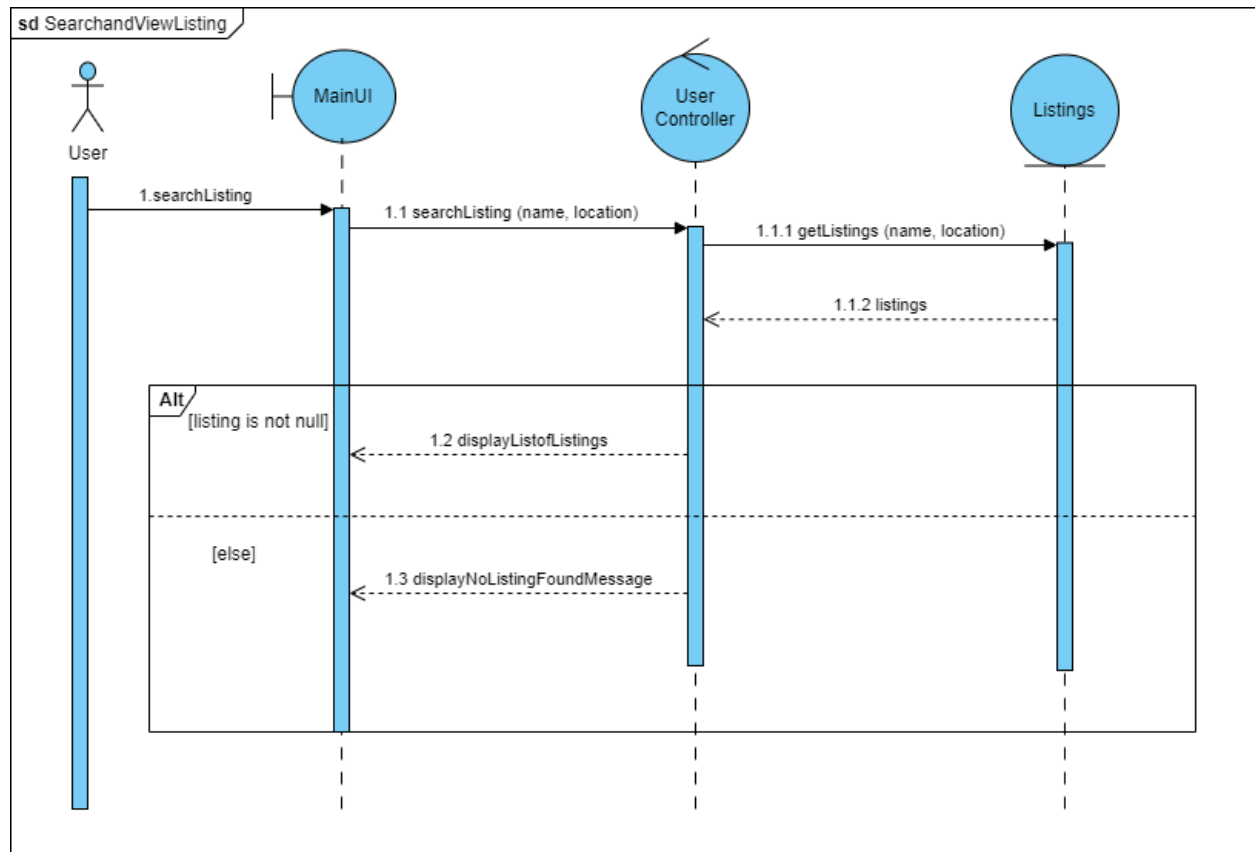
IV.I Create Account



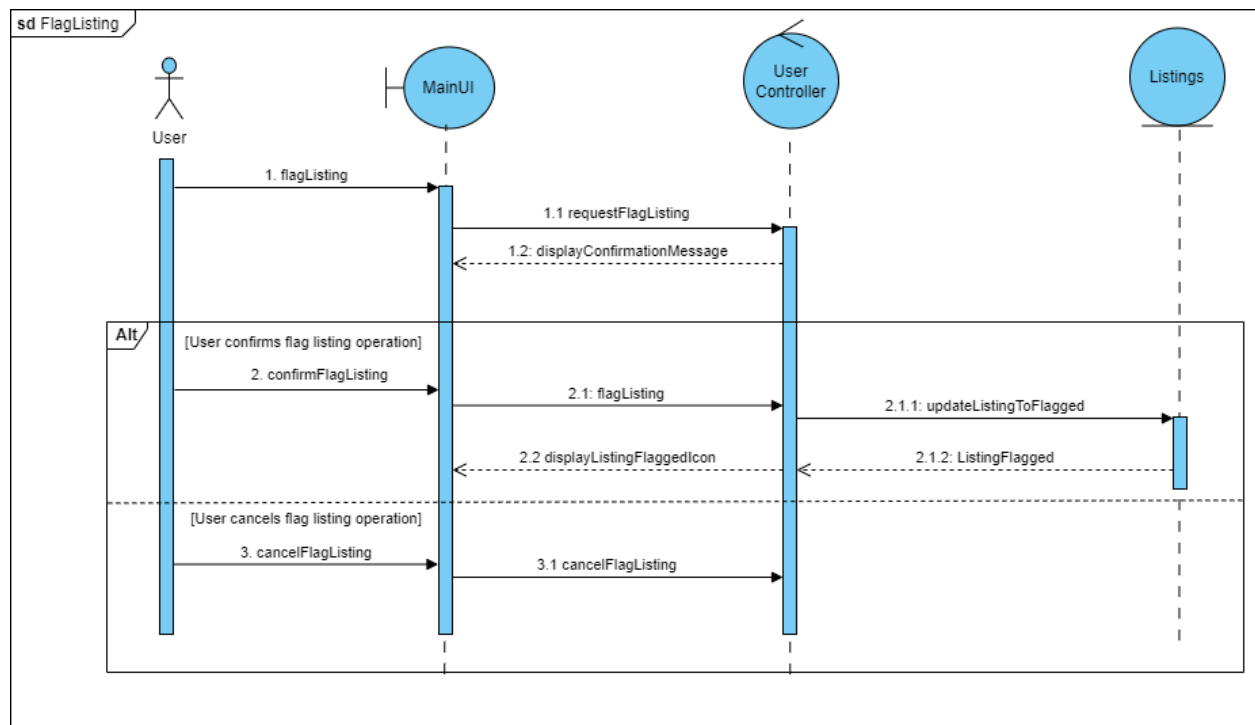
IV.II Login



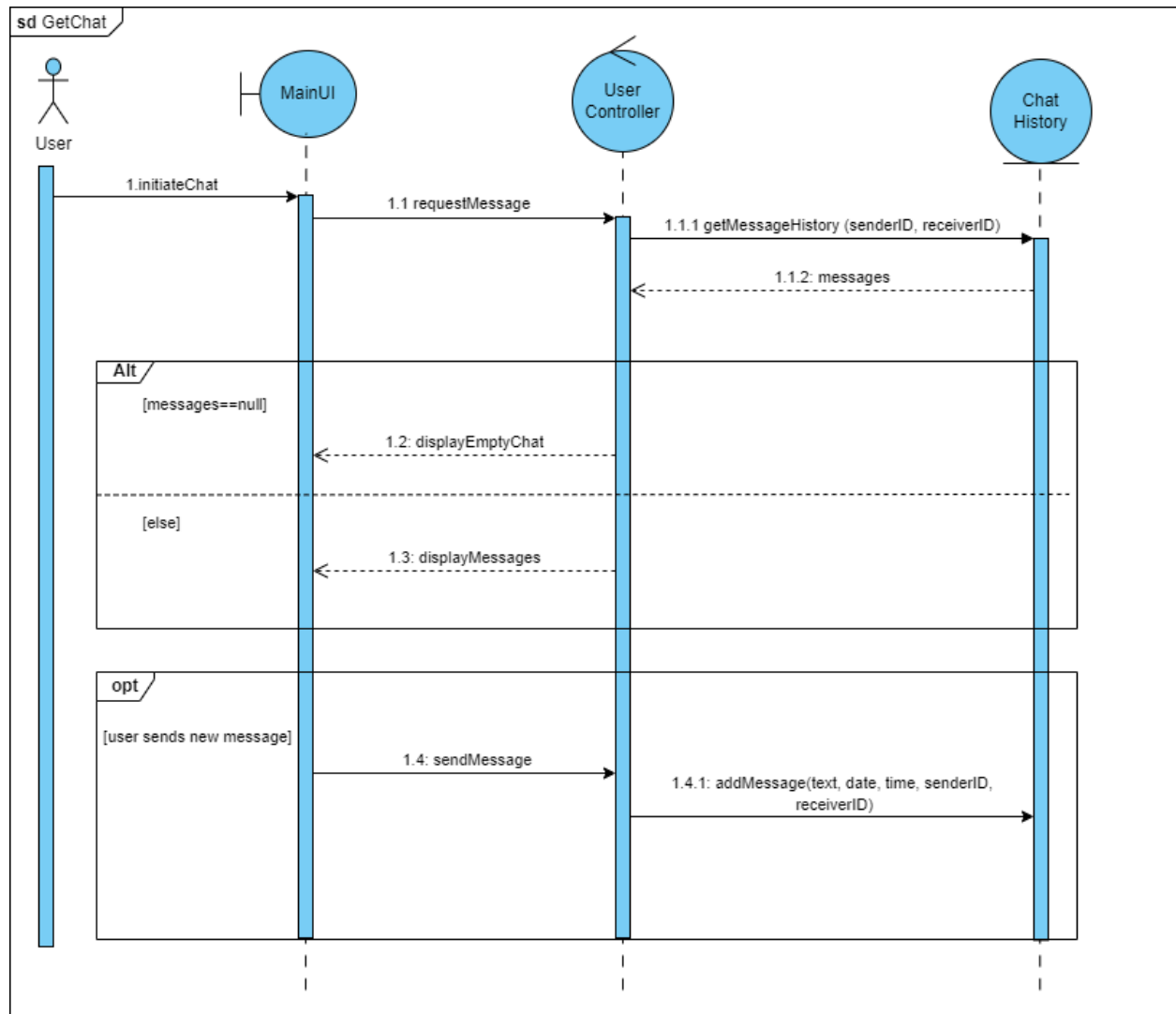
IV.III SearchAndViewListing



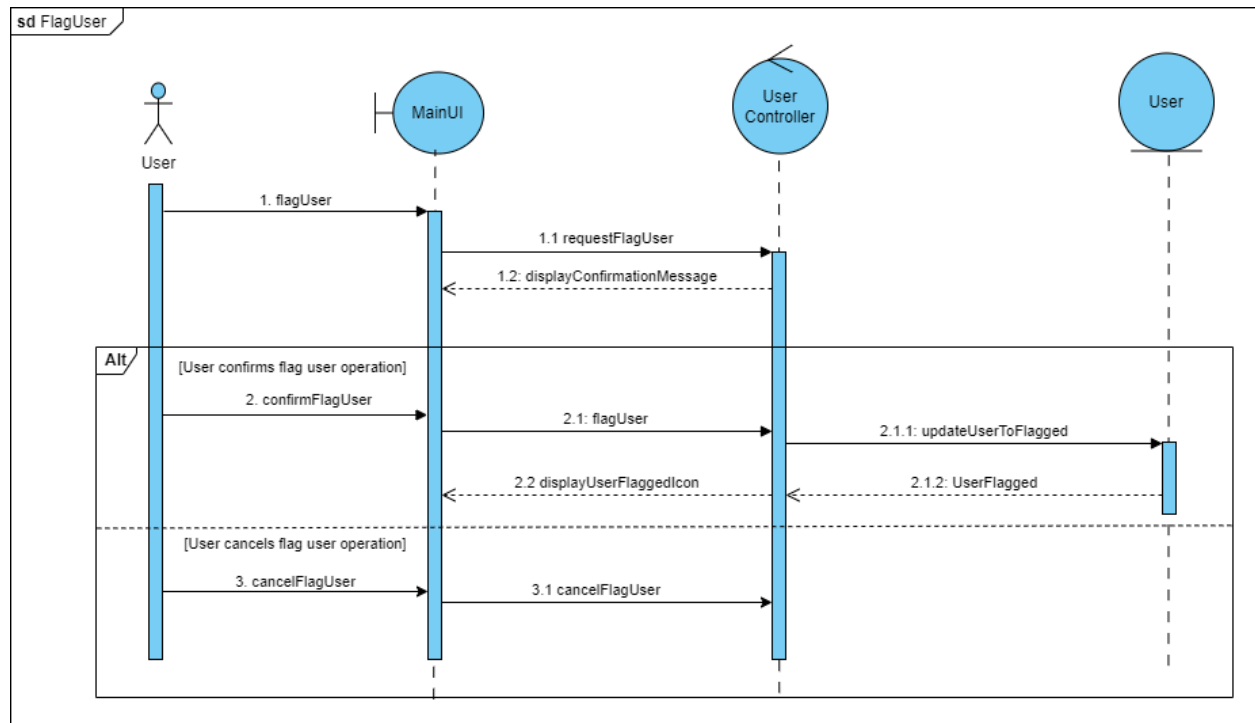
IV.IV FlagListing



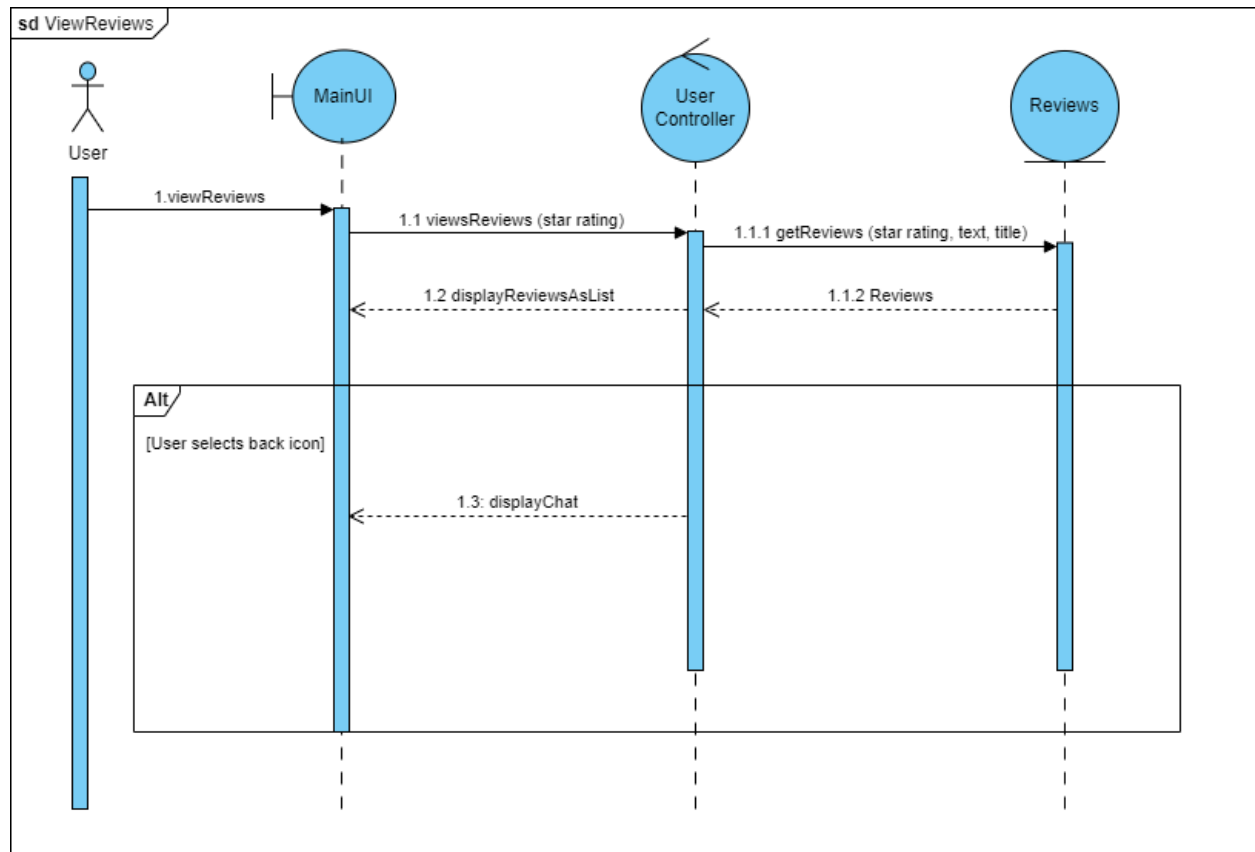
IV.V GetChat



IV.VI FlagUser



IV.VII ViewReviews



5. Initial Dialog map

