

**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

**SC2006 Software Engineering
Lab 1 Deliverables**

Dream Neighbourhood

Group Member	Matric Number
How En Hsin	U2421325J
Veda Ho Yong Qian	U2321709B
Jarrel Ng Tze Xuan	U2421691L
Isaac Teo Yi Jay	U2421749A
Sean Winston Susanti	U2423834G
Joel Ng Wei Heng	U2422193L

Table of Contents

1. Documentation of functional and non-functional requirements	2
A. Functional Requirements	2
B. Non-Functional Requirements	5
2. Data Dictionary	6
3. Use Case Model (Use Case Diagram and Descriptions)	8
A. Use Case Diagram	8
B. Use Case Descriptions	8
I. Search Properties	8
II. View Property Listings	10
III. Manage Favourites	11
A. Save Property Listing	13
IV. User Interactions	14
A. Report Comment	15
V. Account Creation	16
A. Verify Email	17
VI. Authentication	19
A. Reset Password	20
VII. Manage Users	21
VIII. Manage Comments	22
UI Mockups	24

1. Documentation of functional and non-functional requirements

A. Functional Requirements

1. Dream Neighbourhood shall allow the Users to perform property search functions.
 - 1.1. Dream Neighbourhood shall first search for property listings with Basic Filters via a search bar.
 - 1.1.1. Dream Neighbourhood shall allow the Users to specify a budget within a User-defined price range.
 - 1.1.2. Dream Neighbourhood shall allow the Users to specify the property type from a dropdown list (HDB, Condominium, Private Property).

- 1.1.3. Dream Neighbourhood shall allow Users to clear or reset Basic Filters at any time.
 - 1.1.4. Dream Neighbourhood shall apply the selected Basic Filters to property listings and include only those that satisfy all criteria.
 - 1.2. Dream Neighbourhood shall then search for property listings with User-defined Key Locations.
 - 1.2.1. Dream Neighbourhood shall allow Users to have the ability to add multiple specific locations (e.g. workplace, school) by postal code or name.
 - 1.2.2. Dream Neighbourhood shall allow Users to remove Key Locations from their selection.
 - 1.2.3. Dream Neighbourhood shall allow Users to specify a maximum acceptable distance from each Key Location for filtering.
 - 1.2.4. Dream Neighbourhood shall compute the distance from each property listing to each specified location.
 - 1.2.5. Dream Neighbourhood shall apply Key Location filters to property listings and include only those that satisfy the distance criteria.
 - 1.3. Dream Neighbourhood shall lastly search for property listings with User-specified Amenities.
 - 1.3.1. Dream Neighbourhood shall allow the Users to specify the desired Amenities through queries including keywords (e.g., “park”, “hawker centre”).
 - 1.3.2. Dream Neighbourhood shall allow Users to remove selected Amenities from their filter selection.
 - 1.3.3. Dream Neighbourhood shall allow Users to specify a maximum acceptable distance (default: 4 km) from each selected Amenity.
 - 1.3.4. Dream Neighbourhood shall apply Amenity filters to property listings and display only those that satisfy the distance criteria.
 - 1.4. Dream Neighbourhood shall provide visual feedback to Users indicating which filters are currently active.
 - 1.4.1. Dream Neighbourhood shall allow Users to modify any active filter and automatically update search results.
 - 1.4.2. Dream Neighbourhood shall allow Users to clear all active filters in a single action.
 - 1.4.3. Dream Neighbourhood shall allow Users to sort property listings by price, distance, or average rating.
 - 1.4.4. Dream Neighbourhood shall display property listings in pages of 20 items, with the option to load more.
2. Dream Neighbourhood shall allow the Users to view and query detailed information about the property listings in a card-based format.
 - 2.1. Dream Neighbourhood shall display a list of filtered property listings.
 - 2.2. For each property listing, Dream Neighbourhood shall allow the Users to view the Property name, location, postal code, and price.
 - 2.3. Dream Neighbourhood shall allow Users to save property listings as favourites while browsing.
 - 2.4. When a property listing is selected from the card-based list, Dream Neighbourhood shall display a split-view interface.
 - 2.4.1. The left side of the split view shall display the selected property's detailed information (name, price, location, postal code, images, and key attributes).
 - 2.4.2. The right side of the split view shall display an interactive map showing the selected property's location.

- 2.4.3. Dream Neighbourhood shall allow Users to zoom in and out, as well as pan the map, while viewing the selected property.
 - 2.4.4. Dream Neighbourhood shall highlight the selected property marker on the map.
 - 2.4.5. Dream Neighbourhood shall allow Users to click other property markers on the map to update the details shown in the left panel.
 - 2.4.6. Dream Neighbourhood shall dynamically update the map based on active filters, Key Locations, and selected Amenities.
- 3. Dream Neighbourhood shall allow Users to manage their favourite property listings.
 - 3.1. Users shall be able to view a list of all saved favourite property listings on a dedicated Favourites page.
 - 3.2. Users shall be able to navigate to a saved property from the Favourites page to view its details.
 - 3.3. Users shall be able to remove a listing from their favourites.
 - 3.4. Dream Neighbourhood shall retain saved favourites across User sessions.
- 4. Dream Neighbourhood must allow Users to interact with other Users on the platform.
 - 4.1. On a selected property listing, Users must be able to do the following:
 - 4.1.1. Create comments
 - 4.1.2. Edit comments
 - 4.1.3. Delete comments
 - 4.1.4. Create a review with a rating score out of 10, where a higher rating implies a greater degree of satisfaction.
- 5. Dream Neighbourhood shall allow Users to create an account
 - 5.1. Dream Neighbourhood shall allow Users to input their Username, password, and email.
 - 5.2. Dream Neighbourhood shall enforce password strength rules (minimum length, uppercase).
 - 5.3. Dream Neighbourhood shall require email verification for new accounts.
- 6. Dream Neighbourhood shall allow Users to log in using the account they have created
 - 6.1. Dream Neighbourhood shall allow Users to enter their email and password to log in.
 - 6.2. Dream Neighbourhood shall mask the password entered by the Users by replacing text with dots, unless the Users choose to unmask it.
 - 6.3. If email and password do not match, Dream Neighbourhood shall display "Email and password do not match" to the User.
 - 6.4. If email and password match, Dream Neighbourhood shall log the User in and direct the User to the homepage.
- 7. Dream Neighbourhood shall allow Users to reset their password via email.
- 8. Dream Neighbourhood shall allow Users to report comments.
 - 8.1. Dream Neighbourhood shall display an option on every comment to allow Users to report a comment
 - 8.2. When a comment is reported, Dream Neighbourhood shall add that comment to a list of reported comments for the Admin to process.
- 9. Dream Neighbourhood shall allow Admins to create an account with Admin privileges
 - 9.1. Dream Neighbourhood shall allow Admins to input their name, password, and email
 - 9.2. Dream Neighbourhood shall create an account with the "Admin" role with the information entered by the Admin.

- 9.3. Admins shall be able to view and process reported comments.
- 9.4. Admins shall be able to deactivate or delete User accounts.

B. Non-Functional Requirements

1. In order to meet users' performance requirements, Dream Neighbourhood must satisfy the following criteria:
 - 1.1. When a button that takes the user to another menu is pressed, the new screen must be fully displayed within 3 seconds.
 - 1.2. Locations and pins on the map appear within no more than 3 seconds after the map is displayed.
 - 1.3. Retrieval, followed by displaying real-time data takes a maximum of 2 seconds.
 - 1.4. Real-time data sources employed by Dream Neighbourhood must have a 99.9% Network Reliability
 - 1.5. Dream Neighbourhood must be able to support at least 50000 user accounts.
2. To meet users' ease of use requirements, Dream Neighbourhood must satisfy the following criteria:
 - 2.1. Text and buttons use contrasting colours to ensure visibility to the user
 - 2.2. Visual feedback is given when buttons are held or pressed by the user.
 - 2.2.1. The colour of the button darkens while it is held.
 - 2.2.2. The size of the button is reduced to 95% of its original size while it is held.
 - 2.3. While Dream Neighbourhood is processing a user's input, a circular loading indicator is displayed to tell the user that the app recognises their input.
 - 2.4. Icons are used to label the following items on the map:
 - 2.4.1. Different icons represent the different property types.
 - 2.4.2. Different icons represent the different amenity types, such as train stations, bus stations, parks, libraries, schools, etc.
 - 2.5. In addition to displaying the value, displaying the distance between a property and an amenity on the map uses different colours depending on its magnitude.
 - 2.5.1. A distance of greater than 5km exclusive is represented by a red line.
 - 2.5.2. A distance between 2km and 5km inclusive is represented by a yellow line.
 - 2.5.3. A distance less than 2km exclusive is represented by a green line.
 - 2.6. Dream Neighbourhood shall adapt its layout to fit devices with screen widths ranging from 320px to 1920px without horizontal scrolling.
 - 2.7. All text, buttons, and icons shall scale appropriately on mobile devices so that content is clipped or obscured.
 - 2.8. Images, maps, and other visual content shall scale proportionally to the screen size to ensure proper display on mobile devices.
3. In compliance with PDPA standards, Dream Neighbourhood must support the following:

- 3.1. 2-Factor Authentication (2FA) for user logins
- 3.2. User must be able to delete account information

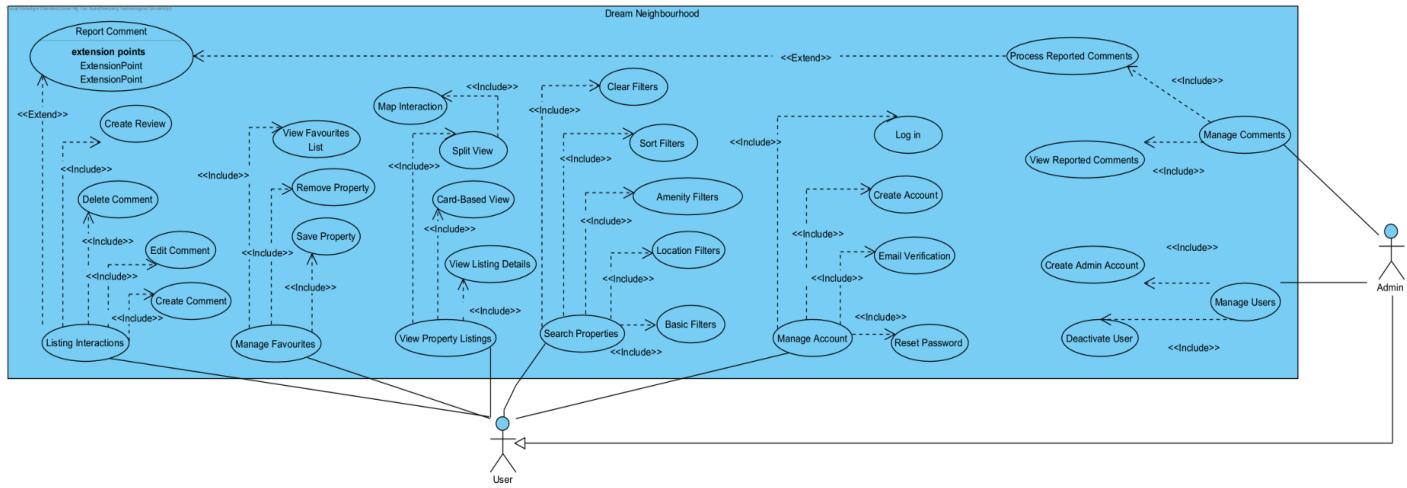
2. Data Dictionary

Term	Definition
Property Listing	An entry representing a house or apartment available for sale or rent, with details such as price, property type, and number of rooms.
Budget	The price range for buying or renting a property specified by the User within which property listings must fall.
Property Type	The category of private or public (state-owned) housing selected by the User, e.g. HDB, Condominium, or Landed Property.
Postal Code	A six-digit code administered by Singapore Post used to identify the location of a property or a User-defined key location.
Key Locations	Important and specific places defined by a Postal Code by the User (e.g. workplace, school, mall) used to measure proximity when searching for properties.
Amenities	User-specified general facilities or services of interest (e.g. park, hawker centre, shopping mall) which User will search by the general terms.
Amenity Distance	The maximum distance acceptable from a property to each amenity.
Favourite Listings	Property listings that the User chooses to save for later reference.
Search Filters	The set of conditions defined by the User (e.g. budget, property type, key locations, amenities) to refine property searches.
Search Results	The list of property listings returned after applying the User's search filters.
User	An individual using the Dream Neighbourhood app to search,

	filter, and save property listings.
Map	An interactive map that displays the geographical location of the Search Results and distance of the listings to the Key Locations and Amenities.
Comment	Users can leave comments on listings to add anecdotal experiences that can help other users in their property search.
Admin	A user with special administrative privileges who manages and oversees the features of an application. Admin user have the authority to perform administrative tasks.

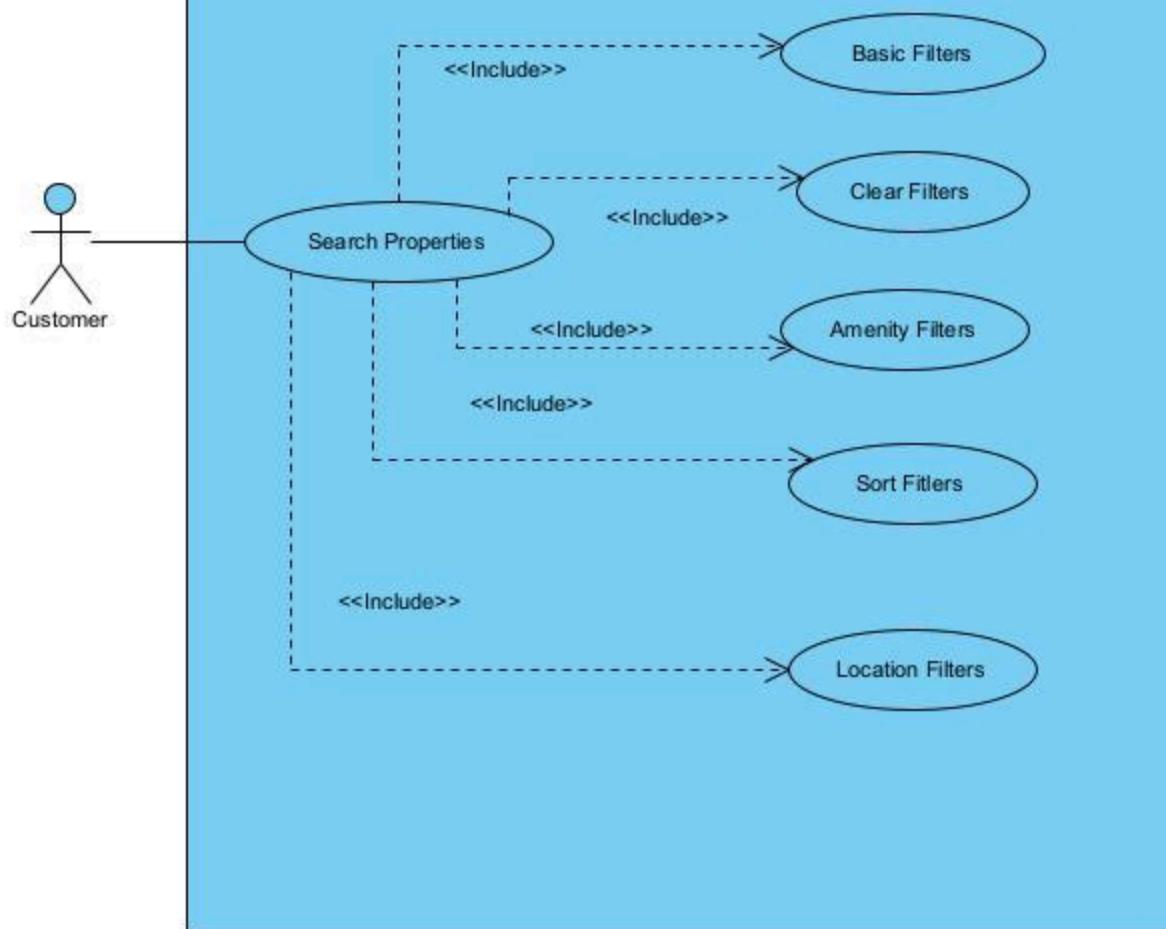
3. Use Case Model (Use Case Diagram and Descriptions)

A. Use Case Diagram



B. Use Case Descriptions

I. Search Properties



Use Case ID	01		
Use Case Name	Search Properties		
Created By	Jarrel Ng	Last Updated By	Isaac Teo
Date Created	1 September 2025	Last Date Updated	1 September 2025

Actor	User
Description	Allows Users to search for properties with filters
Preconditions	User is logged in

Postconditions	System displays filtered and sorted property listings
Priority	High
Frequency of Use	High
Flow of Events	<ol style="list-style-type: none"> 1. User enters keywords in the search bar 2. User applies Basic Filters (budget, property type) 3. User applies key location filters (postal code, workplaces, schools, distance) 4. User applies Amenity filters (e.g., parks) 5. System displays results in pages of 20 items 6. User can sort results by price, distance, or average rating 7. User can clear filters at any time
Alternative Flows	<ol style="list-style-type: none"> 1. No listings are found, the system displays, "No properties found." 2. Filter values are invalid; the system will prompt for correction
Exceptions	<ol style="list-style-type: none"> 1. Network or database failure, the system will display an error message.
Includes	Basic Filters, Location Filter, Amenity Filter, Sort Listings
Special Requirements	Filters must update results dynamically
Assumptions	The property database is available
Notes and Issues	None

II. View Property Listings

Use Case ID	02		
Use Case Name	View Property Listings		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	User
Description	Allows Users to view detailed property info with map interaction
Preconditions	User has performed a search
Postconditions	User views property info and can save to favourites
Priority	High
Frequency of Use	High
Flow of Events	<ol style="list-style-type: none"> 1. System displays property cards with name, price, location, postal code, and thumbnail. 2. User selects a card 3. System shows split view: left = property info, right = interactive map. 4. User can zoom, pan, and select other property markers. 5. Selecting another marker updates the left panel details. 6. User saves the property to favourites.
Alternative Flows	<ol style="list-style-type: none"> 1. Map fails to load, the system will display a static image or an error message
Exceptions	<ol style="list-style-type: none"> 1. Database failure, the system will display an error message.
Includes	Card-Based View, Split-View Details, Map Interaction
Special Requirements	Map integration: the property card must show essential details
Assumptions	Map service is available
Notes and Issues	None

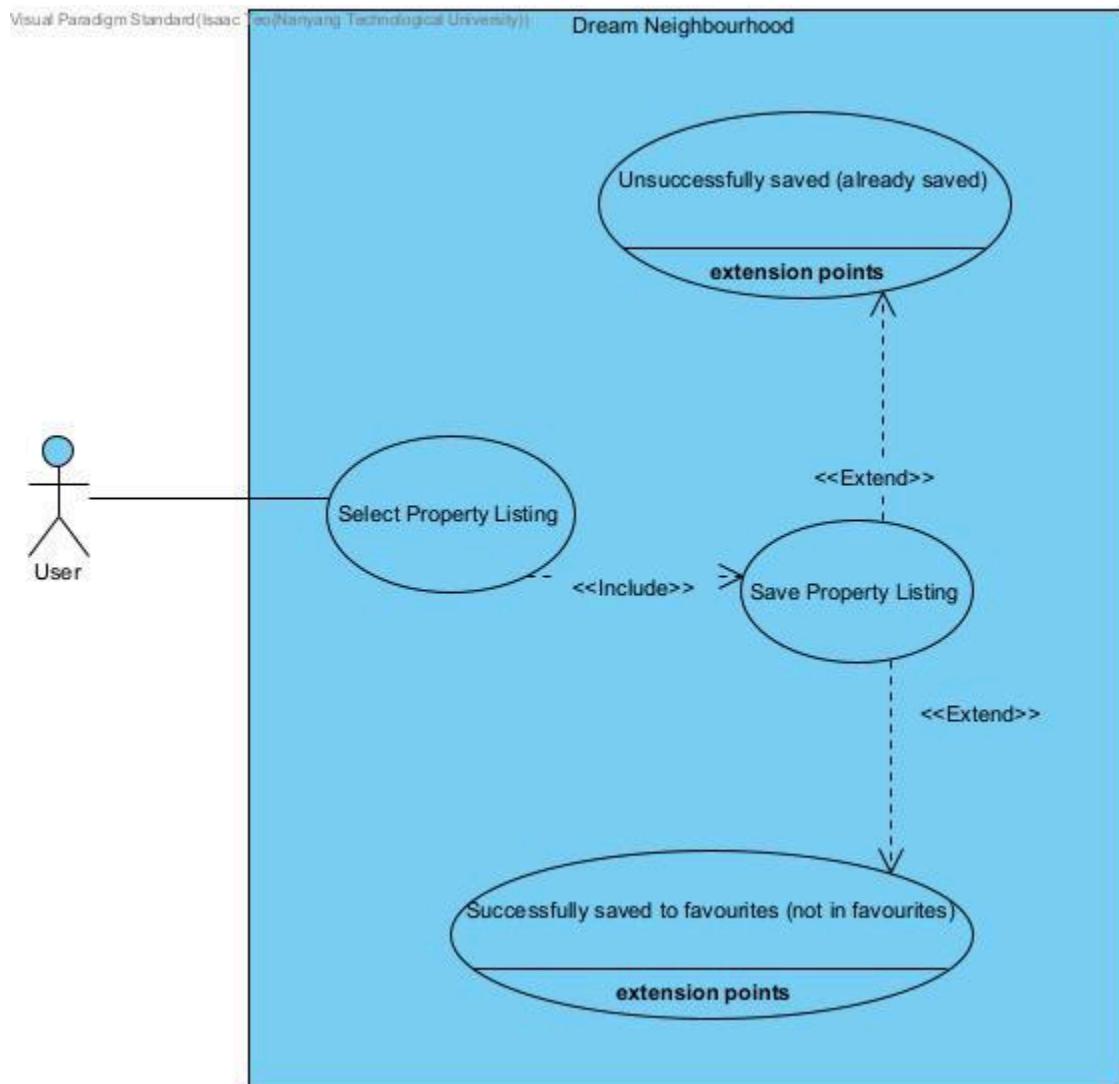
III. Manage Favourites

Use Case ID	03		
Use Case Name	Manage Favourites		
Created By	Jarrel Ng	Last Updated By	

Date Created	1 September 2025	Last Date Updated	
--------------	------------------	-------------------	--

Actor	User
Description	Save and manage favourite properties
Preconditions	User has viewed property listings
Postconditions	Favourites saved across sessions
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ul style="list-style-type: none"> 1. User saves the property to favourites by clicking the star icon, and the system adds it to the favourites list 2. User view the favourites page and the system displays saved properties. 3. User shall remove the property from the favourites list if desired
Alternative Flows	
Exceptions	<ul style="list-style-type: none"> 1. Database failure, system will display error message.
Includes	Save, View, Remove Favourites
Special Requirements	None
Assumptions	None
Notes and Issues	None

A. Save Property Listing



Use Case ID	03A		
Use Case Name	Save Property Listing		
Created By	Isaac Teo	Last Updated By	Isaac Teo
Date Created	30 August 2025	Last Date Updated	30 August 2025

Actor	User
Description	The user selects a property listing and saves it under

	favourite properties
Preconditions	None
Postconditions	A new property listing is added to the list of favourite properties, and it gets a pin highlighted on the map
Priority	High
Frequency of Use	High
Flow of Events	<ol style="list-style-type: none"> 1. The user clicks on the property icon on the map, bringing up an informative pop-up about the listing. 2. The user selects the option to save it 3. The listing is added to the favourites list, and the listing gains a “favourited” pin
Alternative Flows	<ol style="list-style-type: none"> 1. The selected listing is already in the favourites list, and the option to save it is greyed out.
Exceptions	<ol style="list-style-type: none"> 1. The user attempts to save an already saved property listing, despite the button to do so being greyed out.
Includes	None
Special Requirements	None
Assumptions	The User has an existing Dream Neighbourhood Account.
Notes and Issues	None

IV. User Interactions

Use Case ID	04		
Use Case Name	User Interactions		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	User
-------	------

Description	Create comments, review, and report inappropriate comments
Preconditions	User has selected a property listing
Postconditions	Comments are visible and reported comments are flagged
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. User creates, edits, and deletes their own comments 2. User submits review (rating from 1-5) 3. User reports comment will trigger the Admin process to review it
Alternative Flows	
Exceptions	<ol style="list-style-type: none"> 1. Comment fails to be processed, the system will display an error message
Includes	Report Comment
Special Requirements	None
Assumptions	None
Notes and Issues	None

A. Report Comment

Use Case ID	04A		
Use Case Name	Report Comment		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	User
Description	Allows reporting of inappropriate comments
Preconditions	Comment exists

Postconditions	Comment added to the report comment list for Admin to review
Priority	Medium
Frequency of Use	Low
Flow of Events	<ol style="list-style-type: none"> 1. User clicks “Report” on the comment 2. System flags comment 3. Admin shall review the reported comment
Alternative Flows	None
Exceptions	None
Includes	None
Special Requirements	None
Assumptions	None
Notes and Issues	None

V. Account Creation

Use Case ID	05		
Use Case Name	Create Account		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	User
Description	Allows the User to register a new account in Dream Neighbourhood
Preconditions	User has accessed the registration page
Postconditions	User account is created and verified, and is able to log in
Priority	High

Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. The User clicks “Sign Up.” 2. User inputs username, email, and password 3. System shall enforce password strength rules 4. System triggers Verify Email after clicking submit 5. User clicks verification link from their email address that he/she have entered during registration 6. System confirms account creation
Alternative Flows	<ol style="list-style-type: none"> 1. Email is already registered. The system will prompt the user to log in or use another email 2. The password is invalid, and the system will prompt for correction
Exceptions	<ol style="list-style-type: none"> 1. Email fails to deliver to User's email address, allow to resend the verification email
Includes	Verify Email
Special Requirements	Password strength enforcement and email verification are required
Assumptions	The User has a valid email address
Notes and Issues	Ensure email uniqueness and handle network failures during verification

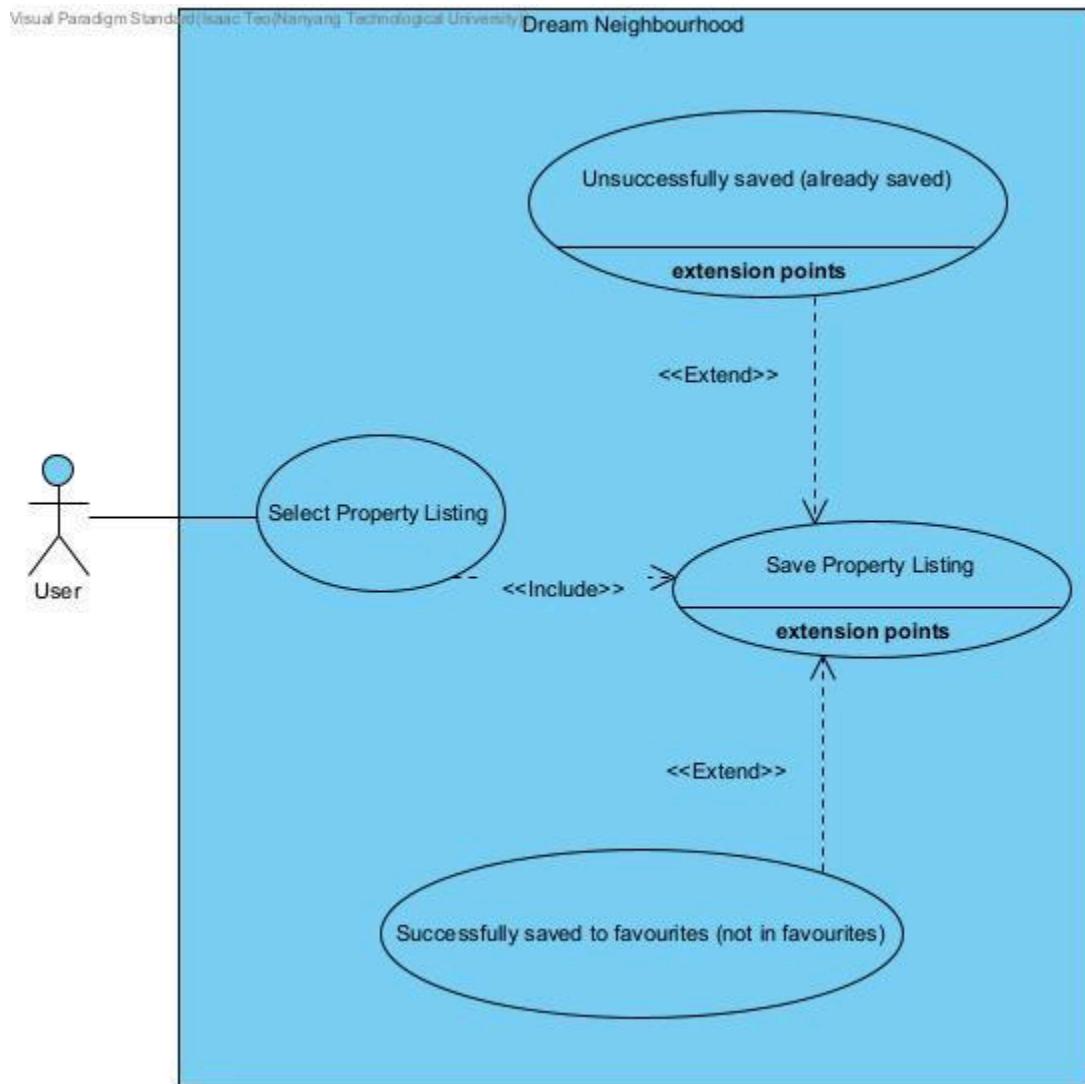
A. Verify Email

Use Case ID	05A		
Use Case Name	Verify Email		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	User
Description	Ensures User's email is valid before activating the account
Preconditions	User has submitted the registration form

Postconditions	User account is verified
Priority	High
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. The system sends a verification email with a link 2. User clicks the link 3. System validates the token and confirms verification
Alternative Flows	<ol style="list-style-type: none"> 1. Token has expired, System shall allow resend 2. Email is undelivered. System shall allow resend
Exceptions	<ol style="list-style-type: none"> 1. System fails to generate a token and shall log the error and notify the admin
Includes	None
Special Requirements	Email service must be reliable
Assumptions	User checks their email
Notes and Issues	None

VI. Authentication



Use Case ID	06		
Use Case Name	Login		
Created By	Jarrel Ng	Last Updated By	Isaac Teo
Date Created	29 August 2025	Last Date Updated	31 August 2025

Actor	User
Description	Allows the User to log into his/her Dream Neighbourhood

	account using his/her email and password.
Preconditions	None
Postconditions	User is logged into his/her Dream Neighbourhood application and is directed to the home page of the application
Priority	High
Frequency of Use	High
Flow of Events	<p>7. The User enters his/her email and password. The password is masked as dots, but the User can choose to unmask it by clicking the eye icon.</p> <p>8. The User clicks the login button</p>
Alternative Flows	None
Exceptions	<p>2. If any of the required information is missing or invalid, an error message is displayed</p> <p>3. If the email and password do not match when the User tries to log in in step 2, Dream Neighbourhood shall display "Email and password do not match" to the User</p>
Includes	None
Special Requirements	The system needs to validate user input data
Assumptions	The User has an existing Dream Neighbourhood Account.
Notes and Issues	None

A. Reset Password

Use Case ID	06A		
Use Case Name	Reset Password		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	User
Description	Allows the User to reset a forgotten password.
Preconditions	User has a registered email
Postconditions	User can log in with a new password
Priority	High
Frequency of Use	Low
Flow of Events	<ol style="list-style-type: none"> 1. User selects “Forgot Password” 2. User enters registered email 3. System sends reset link 4. User clicks the link and sets a new password 5. System confirms reset
Alternative Flows	<ol style="list-style-type: none"> 1. Email is not found, the system will prompt the user to try again
Exceptions	<ol style="list-style-type: none"> 1. Link has expired, the system shall allow resend
Includes	None
Special Requirements	None
Assumptions	None
Notes and Issues	None

VII. Manage Users

Use Case ID	07		
Use Case Name	Manage Users		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	Admin
Description	Admin can create, deactivate, delete accounts.
Preconditions	Admin is logged in.
Postconditions	User accounts updated.
Priority	High
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Admin views users 2. Admin creates, deactivates, or deletes accounts
Alternative Flows	None
Exceptions	<ol style="list-style-type: none"> 1. Upon unauthorized access, admin operations are denied
Includes	None
Special Requirements	None
Assumptions	None
Notes and Issues	None

VIII. Manage Comments

Use Case ID	08		
Use Case Name	Manage Comments		
Created By	Jarrel Ng	Last Updated By	
Date Created	1 September 2025	Last Date Updated	

Actor	Admin
-------	-------

Description	Process reported comments.
Preconditions	Comments have been reported.
Postconditions	Comments are resolved or deleted.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Admin reviews reported comments 2. Admin approves, hides, or deletes comments
Alternative Flows	None
Exceptions	<ol style="list-style-type: none"> 1. When there is a network or database failure, the system shall display an error message
Includes	Report Comment
Special Requirements	None
Assumptions	None
Notes and Issues	None

4. UI Mockups

<https://www.figma.com/design/FyroxjrArSNNbTutLCSPRa/Dream-Neighbourhood-UI?node-id=0-1&t=3p7yy1riR2sxvs8k-1>