
Use Cases

for

REvaluate

Version 1.0 approved

Prepared by Group Z57

Nanyang Technological University

12 April 2023

Use Case ID:	1		
Use Case Name:	Search By Area		
Created By:	Rishabh	Last Updated By:	Rishabh
Date Created:	28/1/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer)
Description:	At the home page, users will toggle the buyer function and input specific perimeters. From there the app will generate an average housing price, details of property and trends of the property.
Preconditions:	The actor must know the address he/she wishes to query
Postconditions:	Upon selection, the map should display pins of all the available houses in that neighborhood
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	<ol style="list-style-type: none"> 1. The user launches the website 2. The user selects the Buyer Option and navigates to the drop-down box to select a district. 3. User selects the neighbourhood he/she wishes to query. 4. User selects flat type 5. User click on search button
Alternative Flows:	-
Exceptions:	E1: User did not fill in all the required fields Redirect user to information not found page
Includes:	Generate Search Data Form Validation
Special Requirements:	Users must be able to immediately view all options in the drop-down box and smoothly make a choice
Assumptions:	Assumes that user is aware of the location he is looking for.
Notes and Issues:	-

Use Case ID:	2		
Use Case Name:	Search By Postal Code		
Created By:	Rishabh	Last Updated By:	Rishabh
Date Created:	28/1/2023	Date Last Updated:	12/4/2023

Actor:	User (Seller)
Description:	On the home page, users will toggle the seller function and key in specific perimeters. From there the app will show prices of house sold near the area the user entered.
Preconditions:	The actor must know the postal code he/she wishes to query
Postconditions:	Upon selection, the map will display pins of all the available houses in that neighborhood and their details
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	<ol style="list-style-type: none"> 1. The user launches the website 2. The selects the Seller Option and navigates to the text Box 3. User inputs postal code he/she wishes to query. 4. User selects flat type 4. User click on search button
Alternative Flows:	-
Exceptions:	<p>E1: No nearby properties in the area No nearby property in the region for the actor to take reference from. The actor can then navigate the map to find nearer alternatives</p> <p>E2: Invalid postal code Redirect user to 'no information found' page</p>
Includes:	<p>Generate Search Data</p> <p>Form Validation</p>
Special Requirements:	Users must know the exact postal code location.
Assumptions:	Assumes that user is aware of the location he is looking for.
Notes and Issues:	-

Use Case ID:	3		
Use Case Name:	View House Details		
Created By:	Alexus	Alexus	Alexus
Date Created:	26/01/2023	26/01/2023	26/01/2023

Actor:	User (Buyer/Seller)
Description:	Users can interact with the google map and pins. After selecting a pin, the details of the house will be displayed.
Preconditions:	Buyer must have entered Housing District and selected Town and House Type. Seller must have entered Postal Code and House Type. User (Buyer & Seller) must click on an pin on the map to view the house details of the selected house.
Postconditions:	Houses must be displayed on the Google Maps House details must be displayed
Priority:	Very high priority, main feature of the website
Frequency of Use:	1 per Search Area (User may use this feature multiple times depending on the number of searches made)
Flow of Events:	1. User clicks on a pin on the map 2. User views house details
Alternative Flows:	-
Exceptions:	Pins on the map might not display properly or clicking on the pin might not present any housing details if the Google Maps API Network is down.
Includes:	View House Price
Special Requirements:	Users must be able to search for records and get results within 5 seconds.
Assumptions:	Assumes that user is searching for houses to view house details
Notes and Issues:	-

Use Case ID:	4		
Use Case Name:	Display Interactive Housing Prices Map		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	28/1/2023	Date Last Updated:	29/1/2023

Actor:	Google Maps API,
Description:	The platform should display an interactive map with pins to indicate past transactions details. Use case is triggered by a system event where a 'Get Price' form is submitted.
Preconditions:	There is a valid housing neighbourhood or postal code being sent to the Google Maps API.
Postconditions:	An interactive map, with ability to scroll and move, is generated with pins dropped on it
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	<ol style="list-style-type: none"> 1. Submission of 'Get Price' form triggers backend API (part of system). 2. Backend API to return all unique latitude and longitude according to past transactions. 3. Display map with pins drop according to the unique latitude and longitude received. 4. Send neighbourhood / postal code to Google Maps API so map can be centered at the neighbourhood / postal code 5. 'View House Details' use case when pin is selected
Alternative Flows:	-
Exceptions:	Backend API return error: Display "No Information Found"
Includes:	-
Special Requirements:	-
Assumptions:	Area/ postal code is valid
Notes and Issues:	-

Use Case ID:	4		
Use Case Name:	Display Interactive Housing Prices Map		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	28/1/2023	Date Last Updated:	12/4/2023

Actor:	Google Maps API
Description:	The platform should display an interactive map with pins to indicate past transactions details. Use case is triggered by a system event where a 'Get Price' form is submitted.
Preconditions:	There is a valid housing neighbourhood or postal code being sent to the Google Maps API.
Postconditions:	An interactive map, with ability to scroll and move, is generated with pins dropped on it
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	<ol style="list-style-type: none"> 1. Submission of 'Get Price' form triggers backend API (part of system). 2. Backend API take the neighbourhood / postal code and convert it to latitude and longitude 3. Backend API query against the database and get data points within latitude +/- 0.02 and longitude +/- 0.02 radius to it. 5. Send neighbourhood / postal code to Google Maps API so map can be centered at the neighbourhood / postal code 6. Display map with pins drop to indicate past transactions 7. 'View House Details' use case when pin is selected
Alternative Flows:	-
Exceptions:	<p>E1: Google Maps API is down Display "Interaction with Google Maps failed. Please try again later"</p> <p>E2: Backend API return error: Display "No Information Found"</p>
Includes:	-
Special Requirements:	-
Assumptions:	Area/ postal code is valid
Notes and Issues:	-

Use Case ID:	5		
Use Case Name:	Generate Neighbourhood Data Insights		
Created By:	Lavanya	Last Updated By:	Lavanya
Date Created:	27/1/2023	Date Last Updated:	27/1/2023

Actor:	User (Buyer / Seller)
Description:	View advanced data insights of price ranges for different house types
Preconditions:	User must have searched for a particular address or postal code and clicked on advanced data insights for further information
Postconditions:	Statistics for different house types in the chosen area is shown
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	1. API call to the backend to get data from the database 2. The data is then displayed in the form of charts and graphs
Alternative Flows:	-
Exceptions:	E1: No data found for the region Redirect to 'no information found' page
Includes:	-
Special Requirements:	The data must be logical and consistent and available to the user within a few seconds
Assumptions:	-
Notes and Issues:	-

Use Case ID:	6		
Use Case Name:	Form Validation		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	28/1/2023	Date Last Updated:	29/1/2023

Actor:	User (Both Buyers and Sellers)
Description:	When buyers and sellers submit the form to search for houses, we need to check that: - All fields are filled up - For sellers, the postal code submitted is valid
Preconditions:	Use case triggered when the user clicks on the 'Get Price' button on the landing page
Postconditions:	All field are valid, and user will be redirected to the details page
Priority:	High (will not affect core functionality if there is no malicious actors)
Frequency of Use:	1 (per form submitted)
Flow of Events:	1. User clicks on 'Get Price' button 2. Client calls backend API to validate form 3. If all the fields are filled up, system checks if the user is a seller 4. If user is not a seller, system returns success status and the sent data 5. If user is a selling, system validates postal code and return success status if all fields are field up and the latitude and longitude for the postal code
Alternative Flows:	AF1: Backend return status that is not success 1. Redirect user to 'no information found' page
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	7		
Use Case Name:	Filter by Price Range		
Created By:	Yau Xing Zhe	Last Updated By:	Yau Xing Zhe
Date Created:	30/1/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/Seller)
Description:	The actor can specify a price range to be able to view the houses listed within the price range, which will be highlighted from the others
Preconditions:	The actor must know the price range he/she wishes to query
Postconditions:	Upon selection, the map will display pins of all the available houses within the price range
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	1. The actor input the minimum and maximum price 2. The actor click on the filter button 3. Houses within the price range will be highlighted
Alternative Flows:	-
Exceptions:	E1: Minimum Price > Maximum Price Shows message that says price range is invalid
Includes:	-
Special Requirements:	-
Assumptions:	Assumes that user is searching for houses within a price range
Notes and Issues:	-

Use Case ID:	8		
Use Case Name:	Generate Housing Market Insights		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	29/1/2023	Date Last Updated:	29/1/2023

Actor:	User (Buyer, Seller)
Description:	Provides insights on Singapore housing market when the user clicks on 'Get Insights' on the navigation bar
Preconditions:	Use case triggered when user clicks on 'Get Insights' on the navigation bar
Postconditions:	Data charts are displayed
Priority:	Medium (secondary functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	1. User clicks on 'Get Insights' tab on the navigation bar 2. Page displays charts providing data insights on the Singapore housing market
Alternative Flows:	-
Exceptions:	Backend API return error: Display "No Information Found"
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	9		
Use Case Name:	Generate Estimated Pricing		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	29/1/2023	Date Last Updated:	30/1/2023

Actor:	- (triggered by system event)
Description:	When a buyer or seller submits the 'Get Price' form, the system needs to provide an estimated buying or selling price to display to the user.
Preconditions:	Use case triggered when the system generates the search data
Postconditions:	The details page displays an estimated buying or selling price
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	<ol style="list-style-type: none"> 1. System calls backend API to scan through all transactions with the same details as the searched details 2. Backend API returns the 25th, 50th and 75th percentile of the transactions 3. Set estimated buying or selling price as the 50th percentile price
Alternative Flows:	-
Exceptions:	Backend API return error: Display "No Information Found"
Includes:	-
Special Requirements:	-
Assumptions:	The neighbourhood or postal code, and flat type provided are valid
Notes and Issues:	-

Use Case ID:	10		
Use Case Name:	Generate Search Data		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	29/1/2023	Date Last Updated:	30/1/2023

Actor:	- (triggered by system event)
Description:	When the search details are submitted and the form is validated, the search data must be generated for display on the details page
Preconditions:	Use case triggered when search details are submitted and validated
Postconditions:	Pricing, data insights and map details populated on the details page
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	<ol style="list-style-type: none"> 1. When the user is redirected to the details page, this use case is triggered 2. System to generate pricing details (Case 9) 3. System to generate data insights (Case 5) 4. System to generate interactive housing map (Case 4)
Alternative Flows:	-
Exceptions:	Backend API return error: Display “No Information Found”
Includes:	-
Special Requirements:	-
Assumptions:	The neighbourhood or postal code, and flat type provided are valid
Notes and Issues:	-

Use Case ID:	11		
Use Case Name:	Learn about REvaluate		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	30/1/2023	Date Last Updated:	30/1/2023

Actor:	User (Both Buyers and Sellers)
Description:	When buyers and sellers click on the 'About Us' page, they should be directed to the 'About Us' page with the introduction of what REvaluate do
Preconditions:	Use case triggered when the user clicks on the 'About Us' button on the header
Postconditions:	User redirected to the 'About Us' page
Priority:	Low (does not affect core functionality of system)
Frequency of Use:	1 (per form submitted)
Flow of Events:	1. User clicks on 'About Us' button on the header 2. User redirected to the 'About Us' page with the write-up of what REvaluate is about
Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	12		
Use Case Name:	Log In		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	1/4/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/Seller)
Description:	User must be able to log in using Single-Sign On using an identity provider (Google/Microsoft)
Preconditions:	User has an account with the identity provider
Postconditions:	User will be directed to the REvaluate+ home page
Priority:	Very High (core functionality)
Frequency of Use:	1 per session
Flow of Events:	<ol style="list-style-type: none"> 1. User visits the REvaluate+ tab and click on one of the sign-in button 2. A new tab is opened for the user to sign in with the identity provider 3. Upon successful login, user is directed to the REvaluate+ home page
Alternative Flows:	AF1: Not successful login User will keep trying until they are able to log in through the identity provider
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	Identity provider authentication service will not fail
Notes and Issues:	-

Use Case ID:	13		
Use Case Name:	Log out		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	1/4/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/Seller)
Description:	User must be able to sign out of their account after they signed in
Preconditions:	User is signed in to REvaluate+
Postconditions:	User is signed out and redirected to home page
Priority:	Very High (core functionality)
Frequency of Use:	1 per session
Flow of Events:	1. User clicks on the log out button in REvaluate+ 2. User is signed out and redirected to the home page
Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	Identity provider authentication service will not fail
Notes and Issues:	-

Use Case ID:	14		
Use Case Name:	Compare Neighbourhoods		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	1/4/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/Seller)
Description:	User must be able to view charts and tables to compare statistics of two neighbourhoods
Preconditions:	User chooses two neighbourhoods
Postconditions:	Charts and tables displaying the statistics of the two neighbourhoods are shown
Priority:	Very High (core functionality)
Frequency of Use:	1 per minute
Flow of Events:	<ol style="list-style-type: none"> 1. User choose a neighbourhood from the dropdown box 2. Backend API retrieves statistics of the neighbourhood from the database 3. Statistics displayed using charts and tables 4. User chooses another neighbourhood from the dropdown box and steps 2-3 repeat
Alternative Flows:	-
Exceptions:	Backend API return error: Display "No Information Found"
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	15		
Use Case Name:	Generate Robo Advice		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	1/4/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/Seller)
Description:	User must be able to generate advice on a suitable flat type and neighbourhood based on the needs
Preconditions:	User clicks on the Robo Housing Advisor tab
Postconditions:	A side tab of a suitable flat type and neighbourhood is shown and a chart displaying the prices across time is also shown
Priority:	Very High (core functionality)
Frequency of Use:	1 (per form submitted)
Flow of Events:	<ol style="list-style-type: none"> 1. User chooses the minimum and maximum price using the slider 2. User chooses their workplace location 3. User chooses the number of people in their household 4. User clicks on the get advice button 5. Results side tab is displayed
Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	16		
Use Case Name:	Add Forum Comment		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	1/4/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/Seller)
Description:	User must be able to add a comment to an existing forum post
Preconditions:	User is in the forum post page
Postconditions:	Comment is seen on the forum post page
Priority:	Very High (core functionality)
Frequency of Use:	1 per minute
Flow of Events:	1. User types in the comment input space 2. User submits comment 3. Comment is seen on the forum post page
Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	17		
Use Case Name:	View Forum Post		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	1/4/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/Seller)
Description:	User must be able to view an existing forum post
Preconditions:	User clicks on the forum post
Postconditions:	User is directed to the forum post page
Priority:	Very High (core functionality)
Frequency of Use:	1 per minute
Flow of Events:	1. User clicks on a forum post 2. Information about the post is retrieved from the backend 3. User is directed to the page displaying the post and all its comments
Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-

Use Case ID:	18		
Use Case Name:	Add Forum Post		
Created By:	Augustine	Last Updated By:	Augustine
Date Created:	1/4/2023	Date Last Updated:	12/4/2023

Actor:	User (Buyer/ Seller)
Description:	User must be able to add a forum post
Preconditions:	User clicks on the add post button
Postconditions:	A new forum post is created and seen on the forum page
Priority:	Very High (core functionality)
Frequency of Use:	1 per hour
Flow of Events:	<ol style="list-style-type: none"> 1. User clicks on the add post button 2. User fills in the information of the post 3. User submits the post 4. User is directed back to the forum page and the new post is seen
Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	-
Assumptions:	-
Notes and Issues:	-