

Use Case ID:	001		
Use Case Name:	Register		
Created By:	Tan Wu Ji	Last Updated By:	Boon Yi
Date Created:	11/2/2024	Date Last Updated:	23/2/2024

Actor:	App User, Database
Description:	App User can choose to register for an account which will be stored in the Database
Preconditions:	<ol style="list-style-type: none"> 1. Database must be up and online 2. App User must be connected to Internet
Postconditions:	<p>App User has successfully registered an account with a unique username and password.</p> <p>OR</p> <p>App User is notified of the unsuccessful registration of an account</p>
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> 1. At the home page, App User clicks on “Sign up” and is redirected to the registration page. 2. App User inputs a valid email, a unique username, a password with at least 8 characters, 1 upper-case letter and 1 lower-case letter and the repeated password in the respective fields in the registration page. 3. App User checks the checkbox of “I agree to the Terms of Use and Private Policy” and clicks on ” “Sign Up” to register and create an account. 4. The system verifies the username is unique and the password satisfies the constraints. 5. App User inputs a One Time Password (OTP) which is sent to his/her email inbox for further verification. 6. The system stores App User’s information in the Database. 7. App User is notified that the registration is successful.

<p>Alternative Flows:</p>	<p><u>AF-S2: App User left input field(s) blank.</u></p> <ol style="list-style-type: none"> 1. When the App User clicks on “Sign Up”, the system displays the message “Please fill in all input fields to register for an account!” above the registration page. 2. The system returns to Step 2 and waits for the App User inputs. <p><u>AF-S3: App User did not check the checkbox of “I agree to the Terms of Use and Privacy Policy”.</u></p> <ol style="list-style-type: none"> 1. When the App User clicks on “Sign Up”, the system displays the message “Please tick the checkbox for acknowledging the Terms of Use and Privacy Policy!” above the registration page. 2. The system returns to Step 3 and waits for the App User inputs. <p><u>AF-S4: App User inputs a taken username.</u></p> <ol style="list-style-type: none"> 1. The system displays the message “Username has been taken. Please input a new username!” above the registration page. 2. The system returns to Step 2 and waits for the App User inputs. <p><u>AF-S4: App User inputs a password that does not satisfy the given requirements.</u></p> <ol style="list-style-type: none"> 1. The system displays the message “Password must contain at least 8 characters, 1 upper-case letter and 1 lower-case letter!!!” above the registration page. 2. The system returns to Step 2 and waits for the App User inputs. <p><u>AF-S4: App User inputs mismatched passwords.</u></p> <ol style="list-style-type: none"> 1. The system displays the message “Passwords do not match! Please check again!” above the registration page. 2. The system returns to Step 2 and waits for the App User inputs. <p><u>AF-S5: App User inputs an incorrect OTP.</u></p> <ol style="list-style-type: none"> 1. The system displays the message “Incorrect OTP! Please try again!” above the registration page. 2. The system returns to Step 5 and waits for the App User inputs.
<p>Exceptions:</p>	<p><u>EX-1: App User did not receive the OTP in his/her email inbox.</u></p> <ol style="list-style-type: none"> 1. The “Resend another OTP” button is available after 60 seconds. 2. App User clicks on the “Resend OTP” button to resend another OTP to his/her email inbox.

	3. The system returns to Step 5 and waits for the App User inputs.
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	002		
Use Case Name:	Login		
Created By:	Tan Wu Ji	Last Updated By:	Boon Yi
Date Created:	11/2/2024	Date Last Updated:	23/2/2024

Actor:	App User, Database		
Description:	App User can login to his/her account with their username and password which are stored in the Database. Logging in allows them to save their preferences.		
Preconditions:	<ol style="list-style-type: none"> 1. The Database must be up and online. 2. App User must be connected to the Internet. 3. App User has a registered account. 		
Postconditions:	<p>App User has successfully logged into his/her account.</p> <p>OR</p> <p>App User is notified of the unsuccessful login to his/her account.</p>		
Priority:			
Frequency of Use:			
Flow of Events:	<ol style="list-style-type: none"> 1. At the home page, App User clicks on “Log in” and is redirected to the login page. 2. App User inputs his/her username and password. 3. App User clicks on “LOGIN”. 4. The system verifies the username and password with the Database. 5. Once the information is verified, the App User is redirected to the home page. 		
Alternative Flows:	<u>AF-S2: If the App User inputs an incorrect username or password.</u> <ol style="list-style-type: none"> 1. When the App User clicks on “LOGIN”, the system displays the message “Invalid username and/or password!” above the registration page. 2. The system returns to Step 2 and waits for the App User inputs. 		
Exceptions:	<u>EX-1: The App User did not register for an account.</u> <ol style="list-style-type: none"> 1. The system displays the message “Please register an account in order to log in”. 2. The system directs the App User to the registration page. 		
Includes:	Register		

Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	003		
Use Case Name:	SearchHome		
Created By:	Tan Wu Ji	Last Updated By:	Boon Yi
Date Created:	11/2/2024	Date Last Updated:	23/2/2024

Actor:	App User, API data
Description:	App User will be able to search for properties with keywords such as location or the properties' name. The system will retrieve information from the API data and display the result for the App User.
Preconditions:	<ol style="list-style-type: none"> 1. Database is up and online 2. App User is connected to the Internet.
Postconditions:	<p>App User obtained a list of searched properties based on the keywords.</p> <p>OR</p> <p>App User is unable to obtain a search result based on the keywords.</p>
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> 1. At the home page, the App User clicks on the search bar, types in keywords in the search bar and clicks on the "Search" icon/button. 2. The system retrieves a list of properties from the API data and searches for the relevant properties based on the keywords. 3. The system displays information of the relevant properties such as property name, price and address.
Alternative Flows:	<p><u>AF-S1: App User inputs nothing and clicks on the search icon.</u></p> <ol style="list-style-type: none"> 1. The system displays a list of 20 random properties. 2. The system returns to Step 1 and waits for the App User actions.
Exceptions:	<p><u>EX-1: The keyword input by the App User does not match any searched items.</u></p> <ol style="list-style-type: none"> 1. The system displays the message "No relevant properties are found!". 2. The system returns to Step 1 and waits for the App User actions.
Includes:	
Special Requirements:	
Assumptions:	

Notes and Issues:	
--------------------------	--

Use Case ID:	004		
Use Case Name:	Bookmark		
Created By:	Tan Wu Ji	Last Updated By:	Boon Yi
Date Created:	11/2/2024	Date Last Updated:	23/2/2024

Actor:	App User, Database
Description:	App User can bookmark their interested properties. The system saves the bookmarked properties in the database. App User is able to quickly find their properties when he/she uses the app later.
Preconditions:	<ol style="list-style-type: none"> 1. App User is connected to the Internet. 2. App User registered for an account with the Database. 3. App User has logged in to his/her account.
Postconditions:	<p>App User has successfully added the property into their bookmark and saved into the database.</p> <p>OR</p> <p>App User has successfully removed the property from their bookmark and updated the database.</p>
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> 1. When the App User clicks the property, the system directs App User to the detailed information page. 2. The “bookmark” icon located at the top right corner displayed in grey colour. 3. App User clicks the “bookmark” icon. 4. The system saves the property into the App User’s database. 5. The “bookmark” icon turns yellow, indicating that this property is successfully added into the bookmark database.
Alternative Flows:	<p><u>AF-S2: “Bookmark” icon is in yellow colour</u></p> <ol style="list-style-type: none"> 1. When the App User clicks the yellow colour bookmark, the system removes the property from the database. 2. The system changes the “bookmark” icon to grey colour, indicating that the property is successfully removed from the database. 3. The system returns to Step 1 and waits for the App User actions. <p><u>AF-S2: App User does not log into his/her account</u></p> <ol style="list-style-type: none"> 1. The system displays the message “Please log into your account to bookmark this property”. 2. The system directs the App User to the login page. 3. App User uses the LOGIN use case to log into his/her account. 4. The system returns to Step 1 and waits for the App User input.

Exceptions:	
Includes:	Login
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	005		
Use Case Name:	TransportOption		
Created By:	Tan Wu Ji	Last Updated By:	Boon Yi
Date Created:	11/2/2024	Date Last Updated:	23/2/2024

Actor:	App User
Description:	System prompts App User to choose the transport options such as MRT, CAR, BUS.
Preconditions:	<ol style="list-style-type: none"> App User is connected to the Internet. App User is in the detailed information page.
Postconditions:	<p>App User successfully input the transportation option.</p> <p>OR</p> <p>App User failed to input the transportation option.</p>
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> The system prompts App User to select the transportation option. App User chooses the corresponding option and clicks “Select”. The system saves the App User’s choice.
Alternative Flows:	<p><u>AF-S2: App User does not choose any option and clicks “Select”</u></p> <ol style="list-style-type: none"> The system displays the message “Please choose your transportation option”. The system returns to Step 1 and waits for the App User input.
Exceptions:	<p><u>EX-1: App User clicks “Cancel” button</u></p> <ol style="list-style-type: none"> The system returns to the detailed information page.
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	006		
Use Case Name:	ShowETA		
Created By:	Tan Wu Ji	Last Updated By:	Boon Yi
Date Created:	11/2/2024	Date Last Updated:	23/2/2024

Actor:	App User, API data
Description:	The system shows the Estimated Time of Arrival (ETA) based on the App User's input of the transportation option.
Preconditions:	<ol style="list-style-type: none"> 1. App User is connected to the Internet. 2. App User is in the detailed information page. 3. API data is up and connected.
Postconditions:	<p>The system successfully showed the ETA corresponding to the transport option.</p> <p>OR</p> <p>The system failed to show the ETA.</p>
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> 1. App User clicks "Show ETA". 2. App User uses the TransportOption use case to select the transportation option. 3. The system retrieves the property's address from the API data. 4. The system makes API request calls to calculate the ETA between the addresses. 5. The system displays the ETA.
Alternative Flows:	<p><u>AF-S4: The system failed to calculate the ETA</u></p> <ol style="list-style-type: none"> 1. The system displays the message "Calculating the ETA again..." 2. The system returns to Step 3.
Exceptions:	<p><u>EX-1: The system failed to calculate ETA in 30 seconds</u></p> <ol style="list-style-type: none"> 1. The system uses the FrequentAddress use case to wait for App User to input the address again.
Includes:	TransportOption
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	007		
Use Case Name:	FrequentAddress		
Created By:	Boon Yi	Last Updated By:	Boon Yi
Date Created:	12/02/2024	Date Last Updated:	23/02/2024

Actor:	App User, Database
Description:	App User can input the addresses of places they frequent. The system will show the ETA from their properties of interest to the places they frequent.
Preconditions:	<ol style="list-style-type: none"> 1. App User is connected to the internet 2. App User is in the detailed information page
Postconditions:	<p>The system successfully showed the ETA from their property of interest to their input address.</p> <p>OR</p> <p>The system failed to show the ETA from their property of interest to their input address.</p>
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> 1. App User click on the property of interest 2. System directs App User to the detailed information page 3. The “ShowETA” icon is located at the top right hand corner of the page 4. App User click on the “ShowETA” icon 5. System prompt the App User to input their address 6. System then calculate ETA using ShowETA function 7. System display ETA calculated
Alternative Flows:	<p><u>AF-S5: App User input invalid address</u></p> <ol style="list-style-type: none"> 1. System display “Unable to find address” 2. System return to step 5 and wait for App user input <p><u>AF-S6: Unable to calculate ETA with corresponding transport option</u></p> <ol style="list-style-type: none"> 1. System display “Transport Option not available” 2. System use TransportOption use case and wait for App user to select alternate transport option.
Exceptions:	<p><u>EX-1: App User clicks “Cancel” button</u></p> <ol style="list-style-type: none"> 2. The system returns to the detailed information page.
Includes:	ShowETA
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	008		
Use Case Name:	Show nearby amenities		
Created By:	Boon Yi	Last Updated By:	Boon Yi
Date Created:	12/02/2024	Date Last Updated:	23/02/2024

Actor:	App User, API data
Description:	The system will display nearby amenities from App User property of interest.
Preconditions:	<ol style="list-style-type: none"> 1. App User is connected to the internet 2. App User is in the detailed information page
Postconditions:	<p>The system successfully displays nearby amenities from App User property of interest.</p> <p>OR</p> <p>The system failed to display nearby amenities from App User property of interest.</p>
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> 1. App User click on the property of interest 2. System directs App User to the detailed information page 3. The “Amenities” icon is located at the top right hand corner of the page 4. App User click on the “Amenities” icon 5. System display nearby amenities from property of interest
Alternative Flows:	<p><u>AF-S5: No amenities nearby</u></p> <ol style="list-style-type: none"> 1. System display “Unable to find any amenities” 2. System return to step 2
Exceptions:	<p><u>EX-1: App User clicks “Cancel” button</u></p> <ol style="list-style-type: none"> 1. The system returns to the detailed information page.
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	