Use Cases Description

Use Case ID:	1		
Use Case Name:	Verify Login Credentials		
Created By:	Heng Zeng Xi	Last Updated By:	Phan Huu An
Date Created:	29/08/2024	Data Last Updated:	06/09/2024

Actor:	User (Initiating), Database
Description:	This use case verifies the login credentials of a user attempting to access the system. The user enters their username and password, and the system checks if these credentials are stored in the database. If the credentials are valid, the user is granted access to the system. If the credentials are invalid, the user is prompted to re-enter their credentials, initiate a password recovery process or create a new account
Preconditions:	The user must have an existing account with valid credentials stored in the database
Postconditions:	1. The user is successfully authenticated and granted access to the system, proceeding to the main menu 2. The user is not authenticated, and the system displays an error message, prompting the user to try again or reset their password
Priority:	High
Frequency of Use:	1-2 times per day
Flow of Events:	1. User accesses the login page 2. User enters credentials 3. User submits the login form 4. System validates credentials with database 5. If valid, the system grants access and the user proceeds to the main menu. 6. If invalid, the system denies access and prompts for

	re-entry or password recovery.
Alternative Flows:	AF-S1: User enters incorrect credentials 1. The system detects that the username or password is incorrect 2. The system displays an error message: "Invalid username or password. Please try again." 3. The user is returned to the login page to re-enter their credentials. 4. Return to step 1 AF-S2: User leaves required fields empty 1. The system detects that one or more required fields are empty. 2. The system displays an error message: "Username and password fields cannot be empty." 3. The user is prompted to fill in the missing fields and resubmit the form. 4. Return to step 3 AF-S3: User selects 'Forgot Password' 1. User clicks the "Forgot Password" button. 2. System redirects the user to the password recovery process. 3. Return to the login process after password reset or when the user cancels the recovery process. AF-S4:User Selects 'Create New Account' 1. User attempts to log in but does not have an existing account. 2. System displays an error message: "No account found with the entered credentials." 3. System provides the option: "Don't have an account? Create a new one." 4. User selects "Create New Account." 5. System redirects the user to the Register Account process.
Exceptions:	EX-S1: System unable to validate credentials due to connectivity issues 1. The system displays an error message: "Unable to verify credentials at this time. Please try again later." 2. The user is advised to try logging in again after some time or contact support
Includes:	Register Account
Special Requirements:	N/A
Assumptions:	N/A

Notes & Issues:	N/A

Use Case ID:	2		
Use Case Name:	Register Account		
Created By:	Heng Zeng Xi	Last Updated By:	Tan Zhe Kai
Date Created:	06/09/2024	Data Last Updated:	06/09/2024

Actor:	User (Initiating), Database, System
Description:t	This use case describes the process by which a new user creates an account in the system. It includes entering personal information, setting up security details, and verifying email or mobile number.
Preconditions:	1. The user does not have an existing account. 2. The user has access to the registration page.
Postconditions:	1. A new account is created in the database. 2. The user receives a confirmation email or OTP, requiring verification to activate the account.
Priority:	High
Frequency of Use:	As needed when new users join.
Flow of Events:	1. User navigates to the registration page. 2. User fills in required information, including name, email, and password. 3. User submits the registration form. 4. System checks for duplicate usernames or emails. 5. System sends a verification email or OTP to the user. 6. User verifies their email or enters OTP. 7. System confirms the account creation and directs the user to the login page.
Alternative Flows:	AF-S1: User Decides to Cancel Registration 1. User navigates to the registration page. 2. At any point before submitting the form, the user decides not to

	continue. 3. User exits the registration page without saving any data. 4. System discards any input data and does not create an account. AF-S2: Email Already in Use 1. User fills out the registration form and enters an email address. 2. System checks the email and finds it is already associated with an existing account. 3. System displays an error message informing the user that the email is in use. 4. System offers options to the user: 5. To redirect to the login page for user authentication. 6. To navigate to the password recovery process if the user has forgotten their password.
Exceptions:	EX-S1: Failure to send verification email/SMS 1. The system displays an error message: "Unable to verify credentials at this time. Please try again later." 2. The user is advised to check their email address or try again later.
Includes:	N/A
Special Requirements:	N/A
Assumptions:	N/A
Notes & Issues:	N/A

Use Case ID:	3		
Use Case Name:	Navigating Dashboard		
Created By:	Heng Zeng Xi	Last Updated By:	Chong Zhi Hen
Date Created:	04/09/2024	Data Last Updated:	05/09/2024

Actor:	Users

Description:	This use case begins once the user is logged into the application. It allows the user to navigate through various features of the dashboard, such as viewing electricity usage, tracking monthly bills, and monitoring individual appliances.
Preconditions:	Users must be logged in and authenticated.
Postconditions:	Users can access different functionalities from the dashboard related to energy management.
Priority:	High
Frequency of Use:	Multiple use per week
Flow of Events:	1. User logs in and is directed to the dashboard 2. User views general electricity usage of all connected appliances. 3. User views the individual appliance's electricity usage details. 4. User track and view monthly electricity bills. 5. User navigates to device management 6. User navigates to other sections or logs out.
Alternative Flows:	AF-S1: User attempts to access a non-responsive feature. 1. User selects a feature from the dashboard. 2. System fails to load the selected feature due to a network or server error. 3. System displays an error message and prompts the user to retry or return to the main dashboard. 4. User retries or returns to the main dashboard. 5. Return to step 1. AF-S2: User changes mind about viewing details. 1. User is in a specific section accessed from the dashboard. 2. User decides not to make any changes or entries and wishes to return to the main dashboard. 3. User selects the option to return to the main dashboard without making any changes. 4. System navigates the user back to the main dashboard. 5. Return to step 1.
Exceptions:	EX-S1: User not logged in 1. The system detects that no user is currently logged in and displays an error message: "You are not logged in. Please log in to continue accessing the dashboard." 2. The system prompts the user to log in by redirecting them to the

	login page
Includes:	Changing Personal Information, Changing of settings, Manage Notification, Device Management
Special Requirements:	N/A
Assumptions:	Users are able to log in.
Notes & Issues:	N/A

Use Case ID:	4		
Use Case Name:	Changing Personal Information		
Created By:	Heng Zeng Xi	Last Updated By:	Phan Huu An
Date Created:	29/08/2024	Data Last Updated:	06/09/2024

Actor:	User (Initiating)
Description:	This use case begins when the user successfully logs into the application and loaded into the Dashboard page. It specifically covers the functionalities that allow users to manage and interact with their account settings and update personal information
Preconditions:	Users must be logged in and authenticated.
Postconditions:	Users can update personal details, change passwords, and set security preferences.
Priority:	High
Frequency of Use:	1 to 2 times
Flow of Events:	1. User navigates to the account settings in the dashboard after successful login. 2. Users update personal details such as name, address, or contact information.

	3. User changes the account password. 4. User adjusts security settings or preferences.(2FA) 5. User saves changes and logs out or continues using the app.
Alternative Flows:	AF-S2: User decides not to update personal details 1. User navigates to update personal details but decides not to make any changes. 2. User selects the option to go back or exit without saving. 3. System does not make any changes and returns the user to the main account settings menu. 4. Return to step 1. AF-S3: User decides against changing the password 1. User selects the change password option but decides not to proceed. 2. User chooses to cancel or go back. 3. System cancels the password change process and retains the current password. 4. Return to step 1. AF-S5: User navigates away before saving changes 1. While making changes in any account settings, the user navigates away from the page without saving. 2. System prompts the user to save or discard changes before exiting. 3. If the user chooses to discard, the system does not save changes and returns the user to the previous menu. 4. If the user chooses to save, the system applies changes and then returns to the previous menu. 5. Return to step 1.
Exceptions:	EX-S1: User not logged into their account 1. The system prompts the user to log in by redirecting them to the login page
Includes:	N/A
Special Requirements:	N/A
Assumptions:	User is able to log in
Notes & Issues:	N/A

Use Case ID:	5		
Use Case Name:	Manage Notification		
Created By:	Heng Zeng Xi	Last Updated By:	
Date Created:	04/09/2024	Data Last Updated:	

Actor:	User, System
Description:	This use case allows the user to customise notification settings within the app, deciding which events they receive notifications for, and how (e.g., via email, push notifications).
Preconditions:	1. The user must be logged into their account. 2. The user must have access to the notifications settings page.
Postconditions:	1. The user's notification settings are updated in the system. 2. The system starts sending notifications based on the new settings.
Priority:	Medium
Frequency of Use:	1-2 times per month
Flow of Events:	1. User logs in and accesses the settings menu. 2. User navigates to the notification settings section in the dashboard. 3. User selects or deselects the types of notifications they wish to receive and chooses the delivery method (e.g., email, mobile push). 4. User saves changes. The system confirms the update. 5. System applies the new settings immediately and schedules notifications as per the user’s preferences.
Alternative Flows:	AF-S1: User abandons changes 1. User navigates to notification settings, makes changes, but decides to leave the page without saving. 2. System prompts the user to save or discard changes. If discarded, no changes are applied. 3. Return to step 2

	AF-S2: System fails to save settings 1. System detects a technical error while trying to save the user's notification settings. 2. The system informs the user of the error and suggests trying again later 3. Users are given options to 'Retry' or 'Cancel'. 4. If the user chooses 'Retry', the system attempts to save again. 5. If the user chooses 'Cancel', or if the 'Retry' fails repeatedly, user will return to step 2
Exceptions:	EX-S1: System outage or connectivity issues 1. If the system cannot access the database to update settings or if there is a service disruption, the user is informed of the inability to process changes at this time.
Includes:	N/A
Special Requirements:	Security Considerations: 1. Any changes to notification settings should be securely transmitted and stored to prevent unauthorized access or leaks of user preferences.
Assumptions:	Stable Internet Connection: 1. Assumes that users have a stable internet connection to access and modify notification settings without interruptions.
Notes & Issues:	N/A

Use Case ID:	6		
Use Case Name:	Device Management		
Created By:	Heng Zeng Xi	Last Updated By:	Chong Zhi Hen
Date Created:	04/09/2024	Data Last Updated:	05/09/2024

Actor:	User
Description:	This use case is for users to manage their devices. Users will be able to see which appliances they have added, average power consumption and duration of use.
Preconditions:	1. User has logged in and pressed on manage my devices
Postconditions:	1. User can see, edit, add and remove the devices
Priority:	Medium
Frequency of Use:	1-2 times per month
Flow of Events:	1. User press on manage my devices 2. They are able to view the list of devices, along with the power usage and duration 3. They can see more of list by scrolling up and down 4. They can press save to exit device management menu
Alternative Flows:	AF-S1: User want to edit a device power usage 1. User press on manage my devices 2. They are able to view the list of devices, along with the power usage and duration 3. They can see more of list by scrolling up and down 4. When they want to edit the power usage of a device, they press on it and can change the value using the onscreen number pad 5. When they are okay with it they can press done on the number pad to remove the number pad 6. They can press save to exit device management menu AF-S2: User want to edit a device usage duration 1. User press on manage my devices 2. They are able to view the list of devices, along with the power usage and duration 3. They can see more of list by scrolling up and down 4. When they want to edit the usage duration of a device, a list of pre selected duration will drop down for user to select 5. When they find the value closest to desired value, they can press on the number and it will remove the drop down and changing to the new value 6. They can press save to exit device management menu
Exceptions:	EX-S1: Missing value 1. If the user tries to press save when there is missing value it will

	warn the user and stop the user from saving.
Includes:	Adding New Device, Electrical Consumption Calculation
Special Requirements:	Default power usage value 1. There should be a button for user to return a device power usage to the default value
Assumptions:	The devices all draw power from the power grid.
Notes & Issues:	N/A

Use Case ID:	7		
Use Case Name:	Adding New Device		
Created By:	Zhi Hen	Last Updated By:	Heng Zeng Xi
Date Created:	30/8/2024	Data Last Updated:	06/09/2024

Actor:	User, Database
Description:	This use case allows users to add a new device to their list of devices. The app already has predefined values for common devices, such as average power consumption, stored in the database. The user selects a device from a list (e.g., fridge, air conditioner, microwave), and the system automatically calculates the device's power usage based on these predefined values. The database will store the newly added device in the user's list.
Preconditions:	User is in the manage devices menu
Postconditions:	The device will be added to the list of devices
Priority:	Medium
Frequency of Use:	Everytime the user need to add a device
Flow of Events:	1. User presses on the + button the manage devices menu 2. A new menu pop out with Device, Power Usage and Duration of

	use 3. User first press on device and a drop down of common appliances will appear 4. User will press on the device they want to add 5. The drop down will close and the device user has selected will appear 6. The value of Power Usage will change to the default value of the appliance 7. User will then press on the duration of use 8. A drop down of preselected values will appear and user select the value closest to the duration they want 9. The drop down will close. 10. User will press ADD to add the new device 11. The system stores the device information in the database.
Alternative Flows:	AF-S1: User know the power usage value 1. User presses on the + button the manage devices menu 2. A new menu pop out with Device, Power Usage and Duration of use 3. User first press on device and a drop down of common appliances will appear 4. User will press on the device they want to add 5. The drop down will close and the device user has selected will appear 6. The value of Power Usage will change to the default value of the appliance 7. The User presses the box on power usage and an onscreen number pad will appear to allow them to key in the value. 8. When they are okay with it they can press done on the number pad to remove the number pad 9. User will then press on the duration of use 10. A drop down of preselected values will appear and user select the value closest to the duration they want 11. The drop down will close. 12. User will press ADD to add the new device AF-S2: User device is not on default devices list 1. User presses on the + button the manage devices menu 2. A new menu pop out with Device, Power Usage and Duration of use 3. User first press on device and a drop down of common appliances will appear 4. User will press on the others 5. The drop down will close and the keyboard will appear

	6. User will then key in the device name and presses done 7. The value of Power Usage will be empty 8. The User presses the box on power usage and an onscreen number pad will appear to allow them to key in the value. 9. When they are okay with it they can press done on the number pad to remove the number pad 10. User will then press on the duration of use 11. A drop down of preselected values will appear and user select the value closest to the duration they want 12. The drop down will close. 13. User will press ADD to add the new device
Exceptions:	EX-S1: The power usage entered is too large 1. The system displays an error message that Power usage exceeds the maximum limit of watts. 2. The user is advised to adjust the power usage value to a valid range and attempt to add the device again.
Includes:	Device Management
Special Requirements:	User is already in the manage my device menu
Assumptions:	We have default power consumption value using the database
Notes & Issues:	N/A

Use Case ID:	8		
Use Case Name:	Remove Device		
Created By:	Tan Zhe Kai	Last Updated By:	
Date Created:	20/09/2024	Data Last Updated:	

Actor:	User, Database
Description:	This use case allows users to remove an existing device from the list of devices.

Preconditions:	User is in the manage devices menu.
Postconditions:	The device will be removed from the list of devices
Priority:	Medium
Frequency of Use:	Anytime the user wishes to remove a device
Flow of Events:	1. User clicks on manage devices button. 2. User proceeds to edit the selected device to be removed. 3. User clicks on the 'Dustbin' icon. 4. A confirmation message pops up to confirm with the user to remove the device. 5. Users clicks on confirm and the device is removed from the list of devices and updates the database.
Alternative Flows:	AF-S1: User decides not to remove upon confirmation. 1. User clicks on the 'Dustbin' icon while editing device information. 2. The confirmation message pops up for the user to confirm removal of device. 3. User clicks on cancel to not remove device. 4. The device information remains in the device list and database.
Exceptions:	EX-S1: System error when removing device. 1. User confirms to remove device on confirmation message pop-up. 2. System displays an error message for failure to remove device. 3. User has to acknowledge the error message by clicking on "Ok". 4. System retains device information in the database.
Includes:	Device Management
Special Requirements:	User is already in the manage my device menu
Assumptions:	N/A
Notes & Issues:	N/A

Use Case ID:	9
Use Case Name:	Electrical Consumption Calculation

Created By:	Zhi Hen	Last Updated By:	Heng Zeng Xi
Date Created:	30/8/2024	Data Last Updated:	06/09/2024

Actor:	User, Database
Description:	Using the parameters the User has provided, the app will calculate and give the user an overview of their energy consumption. They will be able to see an estimate of average energy consumption, which appliances are drawing the most electricity, the estimated electricity bill and more.
Preconditions:	User must have keyed in all of the required parameters and is on the device management menu
Postconditions:	The calculation is saved and displayed on dashboard
Priority:	Medium
Frequency of Use:	Everytime calculation is required
Flow of Events:	1. Users navigate to the device management menu to review and adjust device parameters. 2. Press "Save" to start the energy consumption calculation. 3. System calculates total energy usage based on updated device parameters. 4. Upon calculation completion, the dashboard refreshes to display new values and graphs showing energy distribution. 5. Receive a confirmation notification and automatically return to the main dashboard.
Alternative Flows:	AF-S1: User modifies parameters before final calculation 1. User navigates to the device management menu. 2. User adjusts one or more parameters related to the devices 3. User presses "Calculate" to see the immediate impact of the changes on the estimated energy consumption. 4. The system recalculates the values based on the new inputs. 5. The dashboard updates to reflect the new calculation results. 6. If satisfied, the user presses "Save" to finalise the changes, or continues to adjust parameters if further refinement is needed. 7. Return to step 1
Exceptions:	EX-S1: System unable to calculate due to missing data

	1. User presses "Calculate" but has not completed all necessary input fields. 2. The system checks for completeness and finds missing data. 3. An error message is displayed, instructing the user to enter all required information. 4. User returns to the input fields to complete the missing data and then reinitiates the calculation. EX-S2: System error during calculation process 1. User presses "Calculate" after entering all required data. 2. A system error occurs during the calculation, unrelated to user input (e.g., server error, timeout). 3. The system displays a generic error message suggesting the user to try again later. 4. Users may attempt to recalculate immediately or return to the dashboard to try again at a later time.
Includes:	Device Management
Special Requirements:	N/A
Assumptions:	The power usage is calculated based on average consumption and does not account for when devices draw more power under load.
Notes & Issues:	N/A

Use Case ID:	10		
Use Case Name:	Generate and Export Reports		
Created By:	Zhi Hen	Last Updated By:	Heng Zeng Xi
Date Created:	05/09/2024	Data Last Updated:	06/09/2024

Actor:	User
Description:	This use case allows the user to generate and view reports based on their historical energy consumption. Reports can include detailed usage statistics, comparisons with previous periods, and graphical

	representations of data to aid in understanding consumption trends. Users can also share reports via email or social media, enabling efficient communication of energy data with others.
Preconditions:	1. User has logged into with their account 2. Historical data on energy consumption is available. 3. External services (e.g. email) are accessible.
Postconditions:	1. User will be able to change settings. 2. A report based on the selected parameters is generated and displayed to the user. 3. The user can export the report to external formats such as PDF or Excel. 4. The user can share the report via various platforms.
Priority:	Medium
Frequency of Use:	Monthly or as needed
Flow of Events:	1. User navigates to the "Reports" section via the main dashboard or menu. 2. Users select the type of report they wish to generate (e.g., daily, monthly, yearly consumption). 3. User selects additional parameters such as time frame, type of data visualisation (chart, graph, table), and comparison metrics. 4. The system processes the request and generates the report based on the specified parameters. 5. The report is displayed to the user with options to download or export. 6. Users can choose to generate another report or return to the dashboard. 7. User selects the platform to share the report, and the system generates the file for sharing.
Alternative Flows:	AF-S1: User wants to regenerate a previously generated report 1. User navigates to the "Reports" section. 2. Users access the "History" tab where previously generated reports are listed. 3. User selects a report from the list and chooses the option to regenerate. 4. The system fetches the data for the same parameters as the original report and displays the updated report. 5. Users can then download or share the regenerated report. AF-S2: User modifies the data visualization type

	1. User selects to generate a report as usual by setting the desired parameters. 2. Before finalising, the user selects the option to change the visualisation type (e.g., from bar chart to line graph). 3. The system updates the report preview with the new visualisation. 4. User reviews the updated visualisation and proceeds to generate the full report with the chosen settings. AF-S3: Export Report 1. After generating the report, the user selects the export format (PDF or Excel). 2. The user chooses a sharing platform 3. The system generates the file and shares it via the chosen platform.
Exceptions:	EX-S1: No data available for the selected period 1. User selects a date range for the report and proceeds to generate it. 2. The system attempts to fetch data for the specified period but finds no records. 3. The system displays an error message indicating no data is available for the chosen period. 4. Users are prompted to select a different date range or to try again later. EX-S2: System error during report generation 1. User sets the parameters for the report and initiates the generation process. 2. A system error occurs, preventing the report from being generated. 3. The system displays an error message detailing the issue and possibly suggesting steps to resolve it or try again. 4. Users may attempt to regenerate the report or contact support if the issue persists. EX-S4:Connection Failure Prevents Sharing 1. The user attempts to share the report via an external platform. 2. A connection failure occurs, preventing the sharing process. 3. The system displays an error message prompting the user to check their internet connection. EX-S5: Insufficient Storage for File Export 1. The user attempts to export the report. 2. The system notifies the user that there is insufficient storage on

	their device and suggests freeing up space.
Includes:	Navigating Dashboard
Special Requirements:	N/A
Assumptions:	The user has active accounts on the selected sharing platforms
Notes & Issues:	N/A

Use Case ID:	11		
Use Case Name:	Setting Up Energy Saving Goals		
Created By:	Sanjeev	Last Updated By:	Heng Zeng Xi
Date Created:	13/09/2024	Data Last Updated:	

Actor:	User
Description:	This use case allows users to set personalized energy-saving goals and monthly energy budgets, monitor progress toward these goals, and receive recommendations to optimize their energy consumption and stay within their budget.
Preconditions:	1. Users must be logged into the app. 2. Users have access to historical energy consumption data. 3. Users have set up energy-saving goals or a monthly energy budget.
Postconditions:	1. The user's energy-saving goal and energy budget are saved in the system. 2. The system tracks and displays progress toward both the goal and the budget. 3. The system provides alerts when the user is close to exceeding their budget or reaching their energy-saving goals. 4. The system provides actionable recommendations to reduce energy consumption or stay within the budget.

Priority:	Medium
Frequency of Use:	Once per goal or when the user adjusts the goal.
Flow of Events:	1. User navigates to the energy management section in the dashboard. 2. User sets a target energy-saving goal 3. User sets a monthly energy budget from the settings menu. 4. System saves both the goal and budget and starts monitoring the user's energy consumption. 5. System displays progress metrics for both the energy-saving goal and budget on the dashboard. 6. System alerts the user when they are close to exceeding their energy budget or approaching their energy-saving goal. 7. System provides recommendations to reduce energy consumption to stay within the budget and meet the energy-saving goal.
Alternative Flows:	AF-S1: User abandons the goal-setting process. 1. System discards the input and returns the user to the main dashboard. AF-S2: User updates their energy-saving goal or budget mid-period. 1. System recalculates the energy consumption and updates the progress metrics.
Exceptions:	EX-S1: System error prevents the goal from being saved. 1. System displays an error message: 'Unable to save goal. Please try again later.'
Includes:	Energy Consumption Monitoring Report Generation(View reports based on their historical energy consumption) Manage Notifications (to notify users when they're close to meeting their goals)
Special Requirements:	N/A
Assumptions:	1. Historical consumption data is available 2. Users will adjust their energy consumption to meet the set

	goals and stay within the budget.
Notes & Issues:	1. Users may require clear instructions on setting realistic goals. 2. Accuracy of goal tracking depends on timely data from the appliances.

Use Case ID:	12		
Use Case Name:	Real Time Appliance Health Monitoring		
Created By:	Sanjeev	Last Updated By:	Heng Zeng Xi
Date Created:	13/09/2024	Data Last Updated:	16/09/2024

Actor:	User, System, Database
Description:	This use case offers users real-time diagnostics on appliance health by analyzing power consumption patterns and usage duration. It helps identify potential issues, optimize performance, and extend the lifespan of appliances through data-driven insights.
Preconditions:	1. Users are logged in and appliances are registered in the system. 2. System has sufficient data on appliance usage.
Postconditions:	User receives alerts on potential appliance issues.
Priority:	High
Frequency of Use:	Continuously monitored.
Flow of Events:	1. System continuously monitors appliance usage patterns. 2. System detects irregular power consumption or excessive usage. 3. If an anomaly is detected, the system runs a diagnostic check to analyze the possible cause of the issue (e.g.,

	malfunction, wear and tear, or incorrect usage). 4. The system sends a notification to the user, alerting them of the detected irregularity and providing recommendations for action (e.g., maintenance, repair, or replacement). 5. The notification includes an overview of the issue, a comparison of current vs. normal usage patterns, and any relevant data or insights. 6. The user can access a detailed appliance health report through the system's dashboard, showing real-time diagnostics and historical data. 7. If the issue persists, the system continues to monitor and escalate notifications as needed, with options for scheduling maintenance or contacting support.
Alternative Flows:	AF-S1: User disables health monitoring alerts. 1. System no longer sends health alerts for specific appliances.
Exceptions:	EX-S1: False positive alert due to brief power surge. 1. System verifies the anomaly before sending alerts.
Includes:	N/A
Special Requirements:	Integration with appliance diagnostics data
Assumptions:	The appliances provide accurate consumption data.
Notes & Issues:	1. Monitoring highly energy-intensive appliances may require additional data from external sensors. 2. False positives due to power fluctuations could lead to unnecessary alerts.

Use Case ID:	13
Use Case Name:	Comparing Energy Usage with Similar Households

Created By:	Sanjeev	Last Updated By:	Heng Zeng Xi
Date Created:	13/09/2024	Data Last Updated:	17/09/2024

Actor:	User, Database
Description:	This use case allows users to compare their energy consumption with that of similar households (e.g., same number of residents, same size home).
Preconditions:	1. Users have registered their household details (e.g., number of residents, home size). 2. The app has access to anonymized consumption data from other users.
Postconditions:	A comparison of energy usage is displayed visually in the form of bar charts or pie charts on the dashboard.
Priority:	Medium
Frequency of Use:	Monthly or as needed.
Flow of Events:	1. User navigates to the comparison section. 2. User selects the criteria for comparison (e.g., household size, location). 3. The system retrieves data from the database based on the selected criteria and generates a visual comparison (e.g., bar chart, pie chart) on the dashboard. 4. The user views the visual comparison directly on the dashboard, making it easy to understand their energy usage relative to similar households
Alternative Flows:	AF-S1: No comparable data is available 1. If no data is available for the selected criteria, the system displays an error message: "No comparison data available. Please try different criteria (e.g no. of family members)."

Exceptions:	EX-S1: Database query fails. 1. System displays an error message and suggests the user try again later.
Includes:	1. Energy Consumption Monitoring 2. Data Comparison Module
Special Requirements:	Integration with a data source of similar households.
Assumptions:	The system has sufficient comparison data Users can easily interpret visual data representations like bar charts and pie charts.
Notes & Issues:	1. Availability of comparison data could be limited in certain regions. 2. Users may require an explanation of how the comparison is made to similar households.

Use Case ID:	14		
Use Case Name:	Smart Device Remote Control		
Created By:	Sanjeev	Last Updated By:	
Date Created:	13/09/2024	Data Last Updated:	

Actor:	User, System and Database

Description:	This use case allows users to remotely control connected smart appliances (e.g., turning off lights or adjusting thermostat settings) through the app.
Preconditions:	1. Users must be logged in. 2. The smart device must be connected to the app.
Postconditions:	The user successfully controls the smart device remotely.
Priority:	High
Frequency of Use:	As needed.
Flow of Events:	1. User accesses the list of connected smart devices from the dashboard. 2. User selects a device to control (e.g., air conditioner, lights). 3. User adjusts the device settings (e.g., power on/off, temperature). 4. System sends a control signal to the smart device. 5. The device responds to the user's command, and the app updates the device status.
Alternative Flows:	AF-S1: Device is unresponsive. 1. System displays an error message: "Device is unresponsive. Please try again later." 2. Users are prompted to reconnect or troubleshoot the device.
Exceptions:	EX-S1: Device is disconnected from the network. 1. System displays an error message and suggests checking the device's connection.
Includes:	1. Smart Device Integration

Special Requirements:	The app must support integration with various smart home platforms.
Assumptions:	The device is compatible with the app's control features.
Notes & Issues:	1. Connectivity issues between the app and smart devices could cause delays or failures in control. 2. The user should be notified when the system fails to execute a remote command.

Use Case ID:	15		
Use Case Name:	Multi-User Account Management		
Created By:	Sanjeev	Last Updated By:	Heng Zeng Xi
Date Created:	13/09/2024	Data Last Updated:	17/09/2024

Actor:	Primary User, Secondary Users
Description:	This use case allows households with multiple users to manage energy consumption through individual profiles under a shared account. Each user can manage their own appliances and view personalized energy tracking, while the primary user maintains control over overall account settings and user permissions.
Preconditions:	1. A primary user has created an account. 2. The primary user can invite secondary users to join the account.
Postconditions:	1. Multiple users within a household can manage their appliances and view personalised energy data.

	2. The primary user maintains administrative control, such as adding or removing users and managing permissions.
Priority:	Medium
Frequency of Use:	As needed.
Flow of Events:	Primary user navigates to the "Manage Users" section. Primary users invite additional users by entering their email addresses. Secondary users receive an invitation and register under the shared account. The primary user manages user roles and permissions, such as controlling who can add or modify appliances, or access household-wide energy data. Each user can log in, add appliances, and view their own energy data while sharing the main household account.
Alternative Flows:	AF-S1: Primary user removes a secondary user. 1. The system removes the user's profile and their appliances from the account. AF-S2: Reinviting a User 1. If a secondary user does not respond to the invitation or fails to register, the primary user can resend the invitation or update the email address.
Exceptions:	EX-S1: Invitation email fails to send. 1. System displays an error message and suggests retrying.
Includes:	1. User Account Management 2. Device Management
Special Requirements:	The system must differentiate between primary and secondary user permissions.
Assumptions:	Multiple users in a household will benefit from managing their own appliances.

Notes & Issues:	Clear user roles (admin vs. regular users) may be necessary to prevent unauthorised changes.

Use Case ID:	16		
Use Case Name:	Energy Provider Integration		
Created By:	Sanjeev	Last Updated By:	Heng Zeng Xi
Date Created:	13/09/2024	Data Last Updated:	17/09/2024

Actor:	User, Energy Provider, Database
Description:	This use case allows users to directly integrate their app with their local energy provider to access outage and service notifications and streamline the billing and payment integration process, enabling them to view and pay their energy bills directly within the app.
Preconditions:	1. The energy provider offers API access. 2. User has an active energy account with the provider.
Postconditions:	The user can view real-time billing data and energy rates from their provider within the app.
Priority:	High
Frequency of Use:	Continuously.
Flow of Events:	1. Users link their energy provider account with the app. 2. The system authenticates the user's credentials and establishes a connection with the energy provider's API. 3. The system retrieves real-time billing data and service/outage notifications from the provider.

	4. The user views real-time updates on energy billing, including due dates, current balance, and payment history, within the app 5. The system sends push notifications or in-app alerts for any service interruptions, planned maintenance, or outages reported by the energy provider. 6. The user can choose to receive notifications via email, SMS, or in-app alerts for billing and service updates. 7. The user can directly access payment options and make payments through the app.
Alternative Flows:	AF-S1: User switches energy providers. 1. System allows the user to relink to a new provider. AF-S2: User temporarily loses connection to the internet. 1. The system displays cached billing information and the last known service status. 2. The system retries to establish a connection periodically until it succeeds or informs the user of any reconnection issues.
Exceptions:	EX-S1: Connection to the energy provider fails. 1. System displays an error message and suggests retrying later. EX-S2: Provider API is down or temporarily unavailable. 1. The system displays an appropriate message indicating that the provider is experiencing technical issues. 2. Users are informed of any possible delays in receiving updates and can still view cached information until service is restored. EX-S3: User enters incorrect provider credentials 1. The system displays an error message and prompts the user to re-enter their credentials or troubleshoot potential login issues. 2. The system offers help options or redirects to the provider's

	support page for assistance.
Includes:	1. API Integration with Energy Providers 2. Manage Notification 3. Billing and Payment Integration
Special Requirements:	Secure API integration with multiple energy providers.
Assumptions:	The user's energy provider supports API integration.
Notes & Issues:	1. Some energy providers might not offer API access, limiting real-time data availability. 2. The system should have fallback options if real-time data isn't available (e.g., estimated billing).

Use Case ID:	17		
Use Case Name:	Backup and Restore User Data		
Created By:	Sanjeev	Last Updated By:	Heng Zeng Xi
Date Created:	13/09/2024	Data Last Updated:	17/09/2024

Actor:	User, Database

Description:	This use case allows users to back up their energy data and settings to protect against data loss and recover from accidental deletions, ensuring continuous access to important information without interruptions.
Preconditions:	1.Users must be logged in. 2.Database is available
Postconditions:	User energy data and settings are backed up to the local database and can be restored later.
Priority:	High
Frequency of Use:	As needed (e.g., after accidental data deletion or for data recovery).
Flow of Events:	1. The user navigates to the "Backup and Restore" section in the settings. 2. The user selects the option to either back up or restore their data. 3. The system securely saves energy data and settings to the local database or retrieves data from it to complete the backup or restore process. 4. The system confirms the success of the backup or restore.
Alternative Flows:	AF-S1: User cancels the backup/restore process. 1. System stops the operation and reverts to the previous state.
Exceptions:	EX-S1: Connection error prevents the backup/restore process. 1. System displays an error message: "Unable to complete the backup/restore process. Please check your connection and try again."
Includes:	1. Database Integration
Special Requirements:	Data encryption and integrity checks before saving to or restoring

	from the local database.
Assumptions:	Users may need to recover their data due to accidental deletions or system failures.
Notes & Issues:	1. The system should securely encrypt all user data before saving to the local database. 2. Users may experience data inconsistencies if they restore data on a different device with incompatible settings.

Use Case ID:	18		
Use Case Name:	Gamification and Rewards		
Created By:	Sanjeev	Last Updated By:	Heng Zeng Xi
Date Created:	13/09/2024	Data Last Updated:	18/09/2024

Actor:	User, System, Energy Provider
Description:	This use case gamifies the app experience by providing users with rewards or badges for meeting energy-saving goals or reducing their consumption over time.
Preconditions:	1. Users must be logged in. 2. Rewards system is enabled

Postconditions:	Users receive points, badges, or other rewards for meeting energy-saving milestones and achievements.
Priority:	Low
Frequency of Use:	As needed (e.g., monthly or weekly).
Flow of Events:	1. The system continuously monitors the user's energy consumption and compares it against pre-set goals (e.g., reducing energy use by 10% over a month). 2. Once a goal is achieved, the system awards points, badges, or rewards to the user. 3. The system updates the user's profile with the latest achievements and displays progress in a dedicated rewards section. 4. The user can view their rewards and track progress towards future goals in the rewards dashboard.
Alternative Flows:	AF-S1: User disables gamification features. 1. System stops tracking or awarding points.
Exceptions:	EX-S1: Error in tracking user's progress towards goals. 1. System displays an error message: "Unable to track progress. Please try again later." 2. The user's previous progress is preserved, and tracking resumes once the issue is resolved.
Includes:	1. Setting Up Energy Saving Goals 2. Rewards System
Special Requirements:	Integration with gamification systems (points, badges, rewards).

Assumptions:	1. Users are motivated by rewards or recognition for their energy-saving efforts. 2. Users understand the energy-saving goals and criteria required to earn rewards.
Notes & Issues:	1. Clear communication is essential to help users understand the criteria for earning rewards 2. Over-gamification may lead to user disengagement if rewards feel trivial or irrelevant.