**Software Requirements Specification**

**for**

PowerSaver

**Version 1.0 approved**

**Prepared by**

Chong Zhi Hen

Heng Zeng Xi

Kalaiselvan Sanjeev

Low Jun Keat

Phan Huu An

Tan Zhe Kai

**Group 4**

**15 Nov 2024**

**Table of Contents**

[**1. Introduction 1**](#_heading=h.1fob9te)

[1.1 Purpose 1](#_heading=h.3znysh7)

[**1.2 Document Conventions 1**](#_heading=)

[**1.3 Intended Audience and Reading Suggestions 1**](#_heading=)

[**1.4 Product Scope 2**](#_heading=)

[**1.5 References 2**](#_heading=)

[**2. Overall Description 3**](#_heading=)

[2.1 Product Perspective 3](#_heading=h.2s8eyo1)

[**2.2 Product Functions 4**](#_heading=)

[**2.3 User Classes and Characteristics 6**](#_heading=)

[**2.4 Operating Environment 8**](#_heading=)

[**2.5 Design and Implementation Constraints 10**](#_heading=)

[**2.6 User Documentation 10**](#_heading=)

[**2.7 Assumptions and Dependencies 11**](#_heading=)

[**3. External Interface Requirements 13**](#_heading=)

[3.1 User Interfaces 13](#_heading=h.2jxsxqh)

[**3.2 Hardware Interfaces 25**](#_heading=)

[**3.3 Software Interfaces 25**](#_heading=)

[**3.4 Communications Interfaces 26**](#_heading=)

[**4. System Features 27**](#_heading=)

[4.1 Login 27](#_heading=h.bqrsypmwgyay)

[4.2 Registration 27](#_heading=h.67xrcjoz9mud)

[4.3 Main Dashboard 28](#_heading=h.fpw9dix2s708)

[4.4 Account Management 31](#_heading=h.76cf8qho4hb1)

[4.5 Notifications Management 31](#_heading=h.cyg2t0ek1x50)

[4.6 Device Management 31](#_heading=h.xmtqprog9ual)

[4.7 Adding New Device 32](#_heading=h.pwumtot2505c)

[4.8 Electricity Consumption Calculator 33](#_heading=h.w7ltc3qc34ld)

[4.9 Export Generated Report 33](#_heading=h.bhekc4wjko3q)

[4.10 Admin Dashboard 34](#_heading=h.oz0cosh9acl3)

[**5. Other Nonfunctional Requirements 34**](#_heading=h.3whwml4)

[5.1 Performance Requirements 34](#_heading=h.2bn6wsx)

[**5.2 Usability Requirements 35**](#_heading=)

[**5.3 Security Requirements 35**](#_heading=)

[**5.4 Reliability Requirements 35**](#_heading=)

[**5.5 Business Rules 35**](#_heading=)

[**6. Other Requirements 36**](#_heading=)

[6.1 Data Dictionary 36](#_heading=h.67ooikv6ljwj)

[**7. Appendix A: Analysis Models 38**](#_heading=h.3ugstrqa0dkw)

[7.1 Use Case Diagram 38](#_heading=h.4fair1x0agg0)

[7.2 Key Boundary and Control Classes 39](#_heading=h.12bqys7jjgjs)

[7.3 Class Diagram 40](#_heading=h.g8iy7fnb18lm)

[7.4 Dialog Map 41](#_heading=h.2ll7xwmgwock)

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# 

# **Introduction**

## **Purpose**

This document outlines the software requirements for the *Android* application ***PowerSaver***. As part of the Singapore Green Plan to achieve net zero emissions by 2050, our team has developed an application to help household owners manage their energy usage.

PowerSaver is a mobile application designed to help users monitor and reduce their household energy consumption. It provides insights and recommendations based on Singapore’s historical energy usage to obtain the average household energy consumption. This allows users to compare their energy usage with similar household types. The app also provides custom recommendations based on their energy consumption, such as alerts for high energy usage. Users can use the built-in electricity consumption calculator to compare against their monthly utility bills.

## **Document Conventions**

Prioritisation of Requirements:The priority of higher-level requirements is assumed to be inherited by detailed requirements unless explicitly stated otherwise.

Main Header:Font Family: Times New Roman, Font Size: 18, Font Weight: Bold

Subsection Header:Font Family: Times New Roman, Font Size: 14, Font Weight: Bold

Content: Font Family: Arial, Font Size: 11

Tables & Figures:Font Family: Times New Roman, Font Size: 11

## **Intended Audience and Reading Suggestions**

This document is intended for users of the app, the software developers of the app, documentation writers, marketing staff, project managers and testers and all other potential business partners.

The document contains a description of each function in the app. For each function, a use case is presented for readers to understand the sequence of each process. The next section entails the requirements for the external interface such as the hardware and software interfaces. The functional requirements for each feature are stated and the non-functional requirements are located at the end of the document.

For users, documentation writers, marketing staff and business partners, there is no recommended sequence for reading the document. As for the software developers and project managers, the recommended sequence would be the overall description first, followed by each system function, and then the specific requirements, which include the external interface and non-functional requirements.

## **Product Scope**

PowerSaver is a mobile application that aims to promote the adoption of energy-saving habits by offering personalised tips for reducing energy consumption, such as using a fan instead of an air conditioner or switching to the eco-mode function available in the appliance. The potential benefits of the app are lower energy bills, enhanced appliance lifespan, and environmental protection.

**Objectives:**

* To provide users with a tool to compute their monthly energy consumption.
* To reduce users' carbon footprint.
* To provide personalised energy-saving tips for users based on their energy consumption.
* To offer detailed reports and visualisations for users to understand their energy consumption.

**Benefits:**

* **Cost Reduction:** Users can significantly lower their utility bills based on their projected monthly energy consumption.
* **Environmental Impact:** Users are encouraged to adopt eco-friendly practices by reducing their carbon footprint and aligning with sustainability goals.
* **Better Decision Making:** Users are provided with data and recommendations for ease of making energy-saving decisions.
* **Check for discrepancies in bills**: Users can calculate the projected cost of electricity consumption and check that it tallies with their actual utility bill.

## References

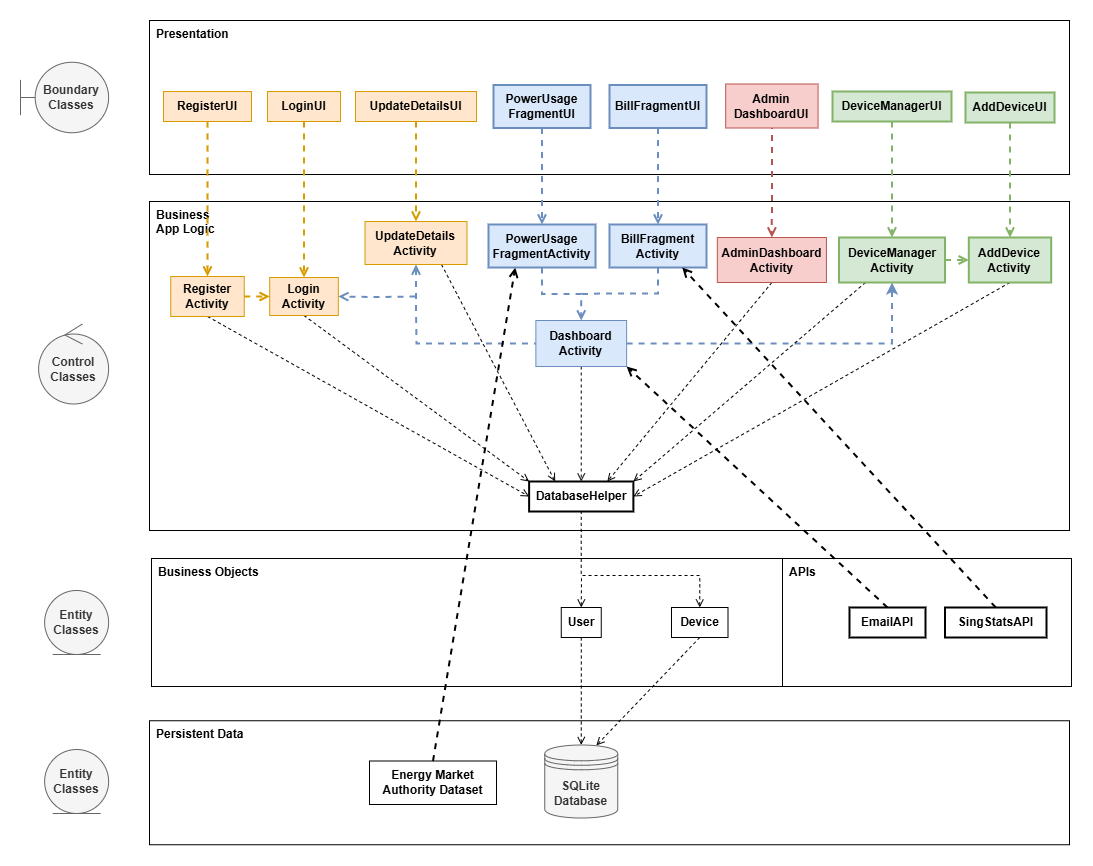
1. GitHub Repository - [github.com/softwarelab3/2006-TDDB-48](http://github.com/softwarelab3/2006-TDDB-48)
2. Android Studio - [developer.android.com/studio](http://developer.android.com/studio)
3. Energy Market Authority of Singapore (EMA) - [ema.gov.sg](http://ema.gov.sg)
4. Singapore’s Open Data Platform - [data.gov.sg](http://data.gov.sg)
5. SP Group - [spgroup.com.sg](http://spgroup.com.sg)
6. Singapore Department of Statistics (DOS) - [singstat.gov.sg](http://singstat.gov.sg)

# **Overall Description**

## **Product Perspective**

The product specified in this Software Requirements Specification (SRS) is a new, self-contained mobile application designed to assist users in monitoring and managing their household electricity consumption. This application does not follow any existing product family but was developed as a standalone solution tailored to meet the growing demand for energy efficiency and cost savings in the home. The product integrates with existing APIs, specifically the Gmail API for sending an email and the SingStats API for data retrieval. Other external packages are imported to meet the functional requirements, such as generating various graph types.

This can be presented in a system architecture diagram, which illustrates the interactions between the internal and external interfaces.



**Figure 2.1: System Architecture Diagram of *PowerSaver* App**

## **Product Functions**

PowerSaver includes the following functions:

### 2.2.1 User Account Management

**Account Registration**

* Allow users to create a new account.
* User credentials must be stored in the database.

**Account Access:**

* Allow users to log into their accounts.
* Enable access to update their user credentials.
* Interact with various functionalities, such as viewing electricity usage, tracking bills, adding/removing appliances, and logging out.

**Verify Login Credentials:**

* Authenticate user login attempts using stored credentials.
* Handle scenarios like incorrect credentials, empty fields, and password recovery.

### 2.2.2 Administrator Dashboard

* Administrators can view the list of registered user accounts.
* Administrators have access to update user credentials.
* Administrators can change the default device list and their corresponding device information.

### 2.2.3 Electricity Usage Monitoring

**View General Electricity Usage:**

* Display the overall electricity usage for all appliances.

**View Individual Appliance Usage:**

* Provide detailed electricity consumption data for each appliance.

**View Graphs and Statistics of Electricity Usage:**

* Display the total current electricity consumption based on devices in the device list.
* Display the average electrical consumption based on data imported from external sources.
* Display the historical tariffs.

**Track Monthly Electricity Bills:**

* Allow users to monitor and track their monthly electricity costs.

### 2.2.4 Appliance Management

**Entering Appliances Information:**

* Enable users to input information about their appliances, including average usage duration and power consumption.
* Assist users in selecting appliances and setting their usage parameters.

**Add/Remove Appliances:**

* Provide functionality to add new appliances to the monitoring list or remove existing ones.
* Appliances are specific to the user only. This means that other users would have a different list of appliances.

### 2.2.5 Electricity Consumption Calculation

**Calculate Electrical Consumption:**

* Compute and provide an overview of the user's energy consumption based on the inputted appliance information.

**Display Detailed Consumption Reports:**

* Allow users to view detailed breakdowns of their energy consumption.
* Identify which appliances are using the most electricity.

**Notification Alerts for Excessive Power Consumption**

* Notify users when the power consumption is above a threshold.

### 2.2.6 Generate Report

**Export Report**

* Allow users to export the dashboard in PDF format.
* Allow users to download the PDF to their devices or share it through other platforms.

**Email Report**

* The report can be sent to the user’s email address.

### 2.2.7 Error Handling and User Assistance

**Handle Incorrect Credentials:**

* Display appropriate error messages and prompts when users enter incorrect login credentials.

**Assist with Appliance Entry Errors:**

* Provide guidance and prompts if the user fails to enter the correct appliance information.

**Notify of Calculation Errors:**

* Alert users to any errors encountered during calculating energy consumption and guide them on how to correct these errors.

## **User Classes and Characteristics**

The users of the application are classified into the respective categories:

### 2.3.1 Regular Users

**Frequency of Use:** Moderate (1-2 times per day)

**Subset of Product Functions Used:**

1. Access Account
2. View electricity usage (general and individual appliances)
3. Track estimated monthly electricity bill
4. Enter appliance information
5. Calculate electrical consumption

**Technical Expertise:** Basic to Moderate

**Security or Privilege Levels:** Basic User Privileges (Authenticated user access)

**Educational Level/Experience:** General homeowners with basic knowledge of using mobile apps and an understanding of household appliances.

**Pertinent Characteristics:**

1. Regular users will most frequently interact with the app to monitor and manage their electricity usage.
2. They are the primary target audience and require a user-friendly interface with clear instructions.

**Importance**: High

**Comments**: This is the most important user class, as the product is designed primarily for them. The app must meet their needs for ease of use, reliability, and clear information presentation.

### 2.3.2 Administrators

**Frequency of Use**: Low (Occasional)

**Subset of Product Functions Used**:

1. Verify Login Credentials
2. Troubleshoot issues related to login, data entry, and calculation errors.
3. Modify the default device list and device information.
4. Manage user accounts.

**Technical Expertise:** High

**Security or Privilege Levels**: Elevated Privileges (Administrative access for troubleshooting)

**Educational Level/Experience:** Technical background, likely with experience in IT support or system administration.

**Pertinent Characteristics:**

1. This user class is responsible for addressing technical issues that regular users might face.
2. They require access to error logs, system diagnostics, and possibly the ability to reset or override user credentials.

**Importance:** Medium

**Comments:** While not as frequently engaged with the product as regular users, technical support users are crucial for maintaining the product's reliability and ensuring a smooth user experience.

### 2.3.3 Power Users

**Frequency of Use:** High (Daily)

**Subset of Product Functions Used:**

1. All functions, with a focus on detailed analysis and tracking.
2. Likely to engage more with advanced features like detailed breakdowns of energy consumption and historical data comparisons.

**Technical Expertise:** Moderate to High

**Security or Privilege Levels:** Standard User Privileges, but with more frequent and intensive usage patterns.

**Educational Level/Experience:** Likely more tech-savvy with a higher interest in energy management and efficiency.

**Pertinent Characteristics:**

1. Power users demand more from the app in terms of performance, data accuracy, and the availability of advanced features.
2. They may push the app's capabilities further and are likely to provide feedback on advanced functionality.

**Importance:** Medium

**Comments:** While not the primary target, satisfying this user class can lead to positive word-of-mouth and detailed user feedback that could drive future enhancements.

### 2.3.4 Infrequent Users

**Frequency of Use:** Low (Occasional)

**Subset of Product Functions Used:**

1. Mainly account access and possibly occasional checks on electricity usage and bills.

**Technical Expertise:** Low

**Security or Privilege Levels**: Basic User Privileges

**Educational Level/Experience:** General users who may not be very tech-savvy.

**Pertinent Characteristics:**

1. This group uses the app sporadically and may need reminders or prompts to engage with certain features.
2. The user interface must be intuitive enough to ensure that even infrequent users can easily navigate the app without needing a refresher each time.

**Importance:** Low

**Comments:** While important to consider, this user class is less critical compared to regular users and power users. However, their needs should still be met to ensure overall satisfaction.

In summary, the most important user class would be the regular users. They are the primary target audience of the application. Followed by the moderately important user classes, which are the administrators and power users. The least important user class would be the infrequent users, who have low active application usage. This categorization ensures that the product is designed and optimised primarily for those who will use it most frequently, while still addressing the needs of other user classes.

## **Operating Environment**

In this section, the operating environment of *PowerSaver* is defined in four components:

### 2.4.1 Hardware Platform

* The app will be designed to run on **mobile devices**, including both smartphones and tablets.
* Supported Devices:
  + Android Devices: A wide range of Android smartphones and tablets from various manufacturers.

### 2.4.2 Operating System and Versions

**Android OS:**

* The app will support devices running Android 9.0 (Pie) and later versions.
* Consideration will be given to the diverse range of Android devices, ensuring compatibility across different manufacturers, screen sizes, and device specifications.

### 2.4.**3 Software Components and Dependencies**

**Cross-Platform Development Framework:**

* The app may utilize a cross-platform framework such as React Native or Flutter to enable simultaneous development for both iOS and Android.

**API Integration:**

* The app will integrate with backend APIs to manage user accounts, retrieve electricity usage data, and perform calculations. These APIs must be compatible with both iOS and Android environments.
* In the latest version of the app, the APIs integrated are *Gmail API* and *SingStats API*.

**Database and Cloud Services:**

* The app will interact with cloud-based services for data storage, ensuring data is securely stored and accessible across both platforms.

**Security and Compliance:**

* The app will comply with the security requirements of the Google Play Store, including data encryption, user privacy, and secure login mechanisms.

**User Interface Consistency:**

* While the app will maintain a consistent user experience across both platforms, it will adhere to the design guidelines of Google's Material Design to ensure a native feel on each platform.

### 2.4.4 Coexistence with Other Applications

* The app must peacefully coexist with other apps on the user’s device, particularly those that might also access data related to electricity usage or home automation.
* It should not interfere with the operation of other apps and should efficiently manage its resources (e.g., battery consumption, memory usage) to avoid negatively impacting the device's overall performance.

The app will be built to operate smoothly on Android devices, covering a broad range of OS versions and ensuring compatibility with other software and hardware components on these devices. The development process will include thorough testing across different environments to ensure a consistent and reliable user experience.

This approach ensures that the app will meet the requirements for a seamless operation in the diverse mobile environment of the target users.

## **Design and Implementation Constraints**

### 2.5.1 Hardware Limitations

* The app is expected to run efficiently on devices and consumes little memory and processing power.
* The battery consumption of the app must be minimal to align to promote energy savings.

### 2.5.2 Regulatory Compliance

* The app must comply with the *Personal Data Protection Act (PDPA)*, which is the standard for the protection of personal data in Singapore. Personal data includes the user's name, password and email address.
* The development team must agree to the terms of use of data from external sources such as government websites and APIs.

### 2.5.3 Technology Stack

* The main Integrated Development Environment (IDE) will be Android Studio and the programming languages used are Java and Kotlin for Android development.
* SQLite is used for local data storage of user data.

### 2.5.4 Security Considerations

* Authentication mechanisms to ensure data privacy.
* Encryption of passwords so that they will be unreadable by hackers in case of a data breach.

### 2.5.5 Design Conventions

* The app should follow material design guidelines for the app’s user interface for consistency and user-friendly interface.
* Ensure the app is compatible with Android versions from API level 21 (Android 5.0) and above.

## **User Documentation**

The documentation is to improve the clarity of the application features for both the end-users and the developers. *PowerSaver’s* documentation is listed as follows:

### 2.6.1 README File

There is a README file in the GitHub repository and application files. This serves as a reference for developers to understand the latest document updates and what each file represents.

It is necessary for users and developers of the application to view the README file as it provides guidelines on the terms of application use. It includes all the prerequisites and advisories that are recommended for users to follow before getting started on running the project.

### 2.6.2 Demo Video

To enhance the user’s clarity of the application, a demo video of *PowerSaver* covers every key application feature. The video would be uploaded as an MP4 file and an unlisted video on YouTube.

### 2.6.3 Code Comments

During application development, the source code will be well-commented to provide an understanding of each code function. Comments will be included for each class, function, method and XML file where necessary. This clarifies the doubts of developers during implementation if any.

## **Assumptions and Dependencies**

### 2.7.1 Assumptions

* Users can access Android devices that run on OS version 5.0 (API level 21) or higher.
* Users will grant necessary permissions, such as Wi-Fi, storage, and email access, to enable the application features.
* Internet connectivity will be available for integration with third-party services, such as API functionalities.
* Users are familiar with basic app functionalities, such as registering a new user account, logging into the account and navigating to different menus.

### 2.7.2 Dependencies

*PowerSaver* is built on various dependencies that can be categorized into front-end and back-end libraries:

**Front-end Libraries**

| Library | Description |
| --- | --- |
| appcompat | Provides backwards-compatible versions of Android UI components for older Android devices. |
| material | Retrieves Google’s Material Design components for creating modern, visually appealing UI elements. |
| constraintlayout | Provides a flexible layout system for the complex UI interfaces without nesting multiple views. |
| mpandroidchart | The library for generating charts and graphs in Android applications. |
| navigation-fragment | Supports navigation between fragments, offering tools to manage back function and transitions. |
| navigation-ui | Provides UI components for app navigation, such as bottom navigation and action bar menus. |
| espresso-core | A testing library that allows developers to validate app interfaces by automating UI tests. |

**Back-end Libraries**

| Library | Description |
| --- | --- |
| activation / activation-android-activation / android-activation | Handles MME-type data in Java applications, and supports email attachments and other data processing functions. |
| google-api-client-android | Integration with Google APIs specifically for Android. |
| google-api-client-extensions-android-http | An extension of google-api-client with additional HTTP capabilities, enabling API communication. |
| google-api-client-gson | Integrates Google API client with Gson for JSON parsing and serialization. |
| google-api-services-gmail | Provides access to Gmail APIs for sending, receiving, or managing emails in Android Studio. |
| google-auth-library-oauth2-http | Manages authentication via OAuth 2.0, which supports secure access to Google APIs. |
| api-client-google-api-client | The core Google API client library for Java, enabling interaction with various Google services. |
| itext7-core | A library for creating and manipulating PDF files. |
| volley | A library for managing API requests, handling caching, and ensuring efficient data transfer. |
| play-services-auth | Provides authentication handling via Google accounts, sign-in functionality secure API access. |
| junit / ext-junit | Provides unit testing libraries for Java (JUnit) and Android extensions for testing logic in applications. |
| android-mail | For sending and managing emails on Android devices. |

**Data Storage**

*PowerSaver* operates using a local SQLite database as the primary data storage of all user and device information. Together with the front-end and back-end dependencies, the app relies on these dependencies for it to run.

# **External Interface Requirements**

## **User Interfaces**

### 3.1.1 General Characteristics

1. Design Standards:
   * Follow the Material Design Guidelines by Google for consistent, modern, and responsive interfaces.
   * Ensure compatibility with Android accessibility features (e.g. contrast settings)

1. Color Scheme:
   * Primary colours: Blue and green to represent sustainability and energy conservation.
   * Secondary colours: purple and white for contrast and clarity.

1. Typography:
   * Standard font across the application: Arial

1. Navigation:
   * Side menu for navigation between application functions (Manage Devices, Account Details, Logout, Close App).

### 

### 3.1.2 UI Mockup

The UI mockup below shows the different screens and interactions between the UI elements. The UI mockup provides an initial idea of the design template for developers to refer to during the implementation phase.

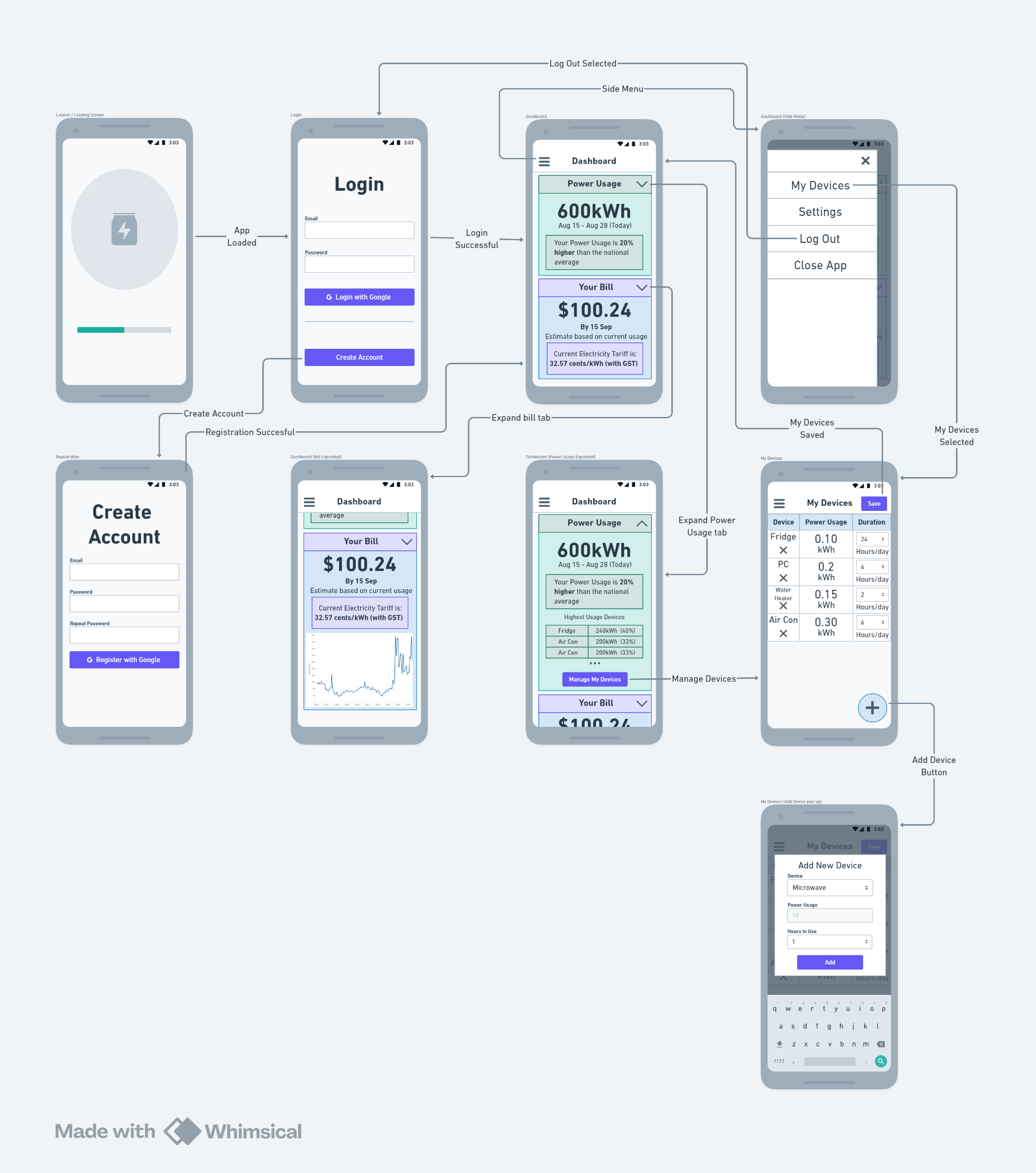
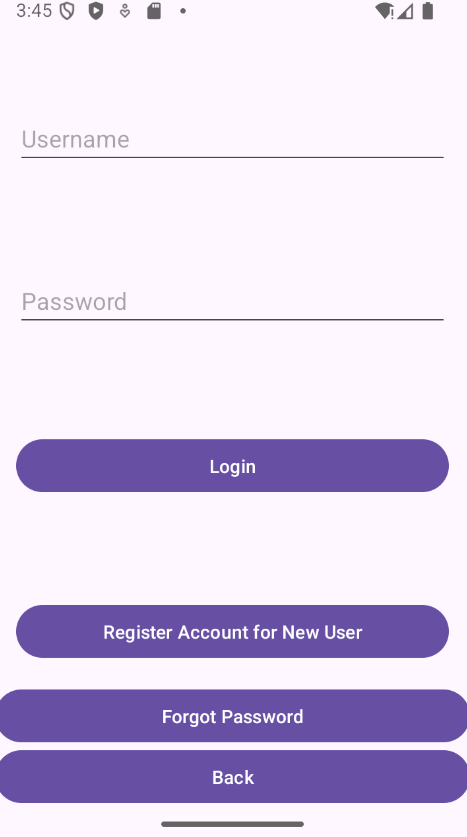
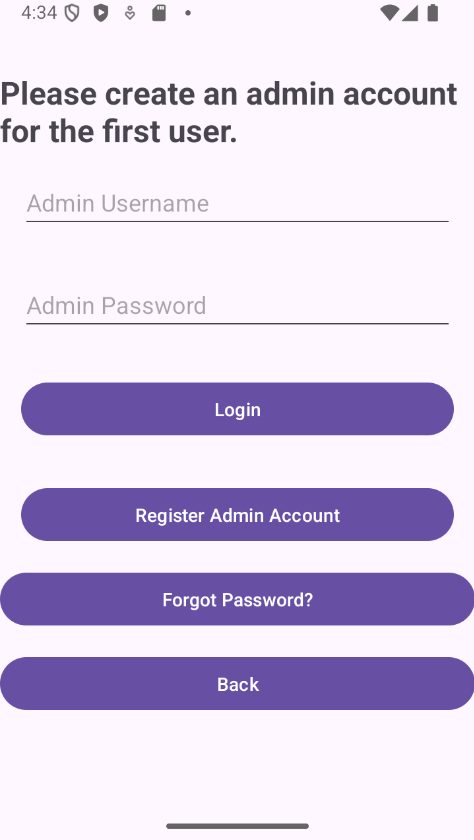


Figure 3.1.2: *PowerSaver* UI Mockup using *Whimsical*

### 3.1.3 *PowerSaver* Screenshots

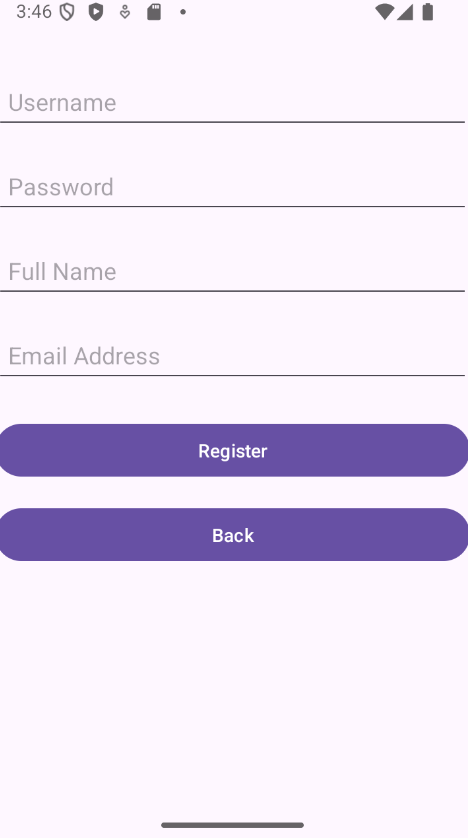
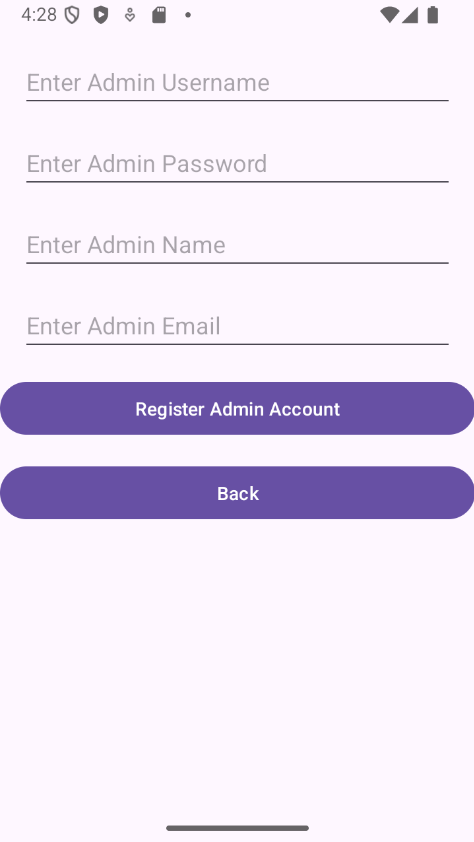
The following images are the actual screenshots of the *PowerSaver* app:

#### 3.1.3.1 User/Admin Login

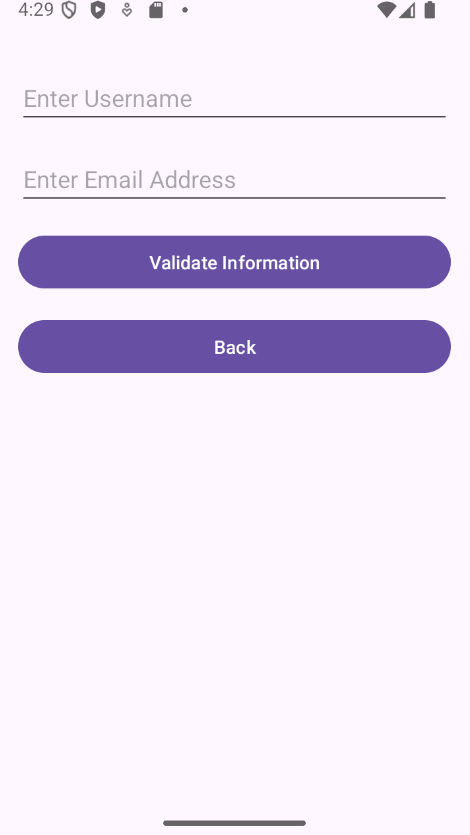
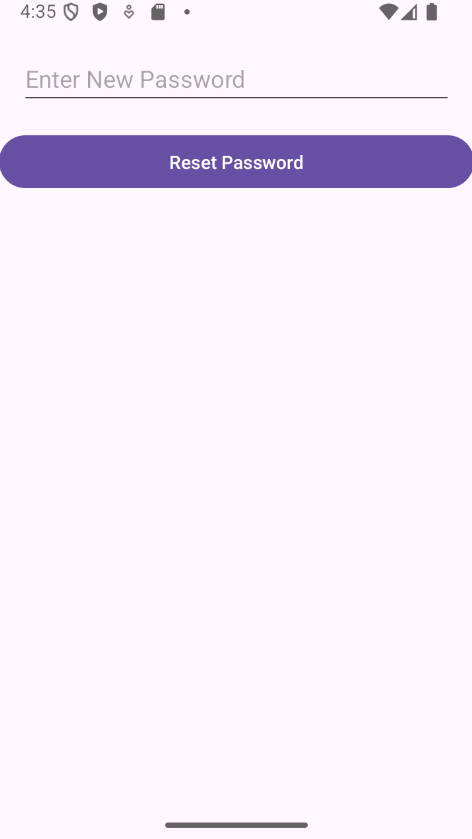
*The user and admin login pages.*

#### 3.1.3.2 Register User/Admin

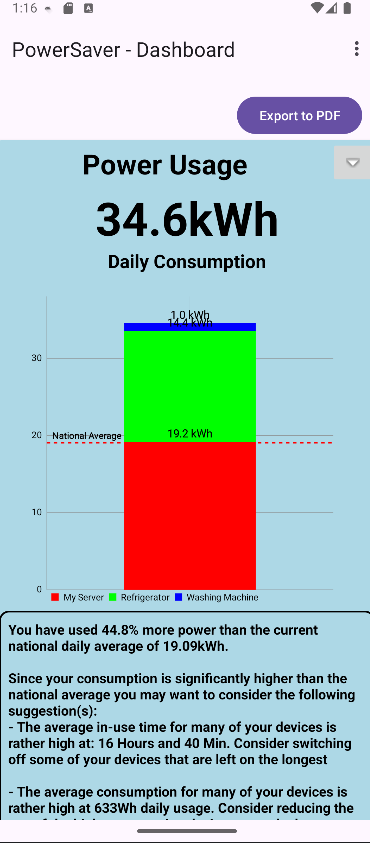
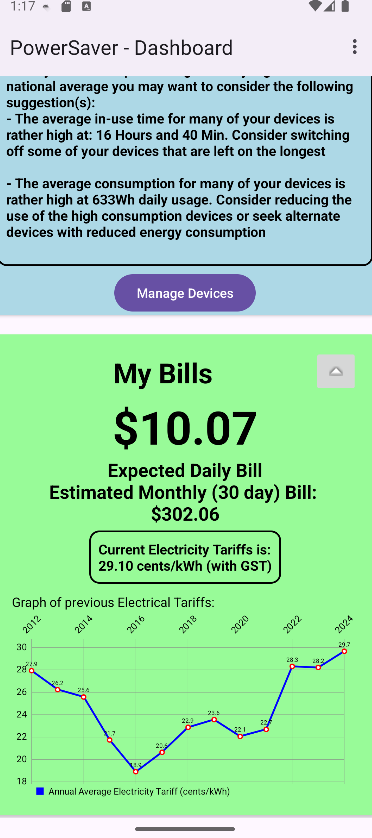
*Users and Admins are required to fill up the required fields to create an account.*

#### 3.1.3.3 Forgot Password

* *

*The user has to validate using their username and email address before creating a new password.*

#### 3.1.3.4 Main Dashboard

* *

*The main dashboard of PowerSaver includes the Power Usage Fragment and Bill Fragment.*

*Charts and graphs are updated in real time.*

#### 3.1.3.5 Device Manager

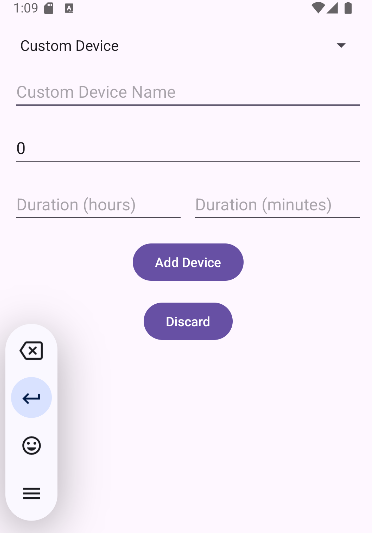
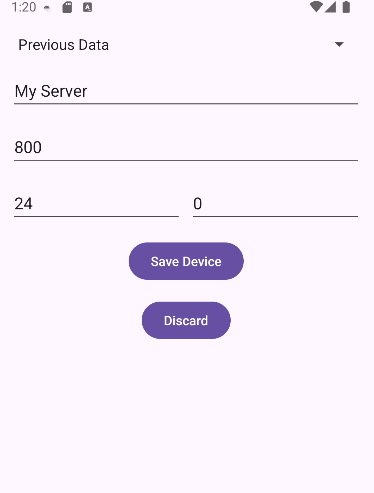
#### 

*The device manager lists the devices, their electricity usage and duration of use.*

*Users can edit the device information or remove the device from the device list.*

#### 

#### 3.1.3.6 Add/Edit Device

* *

*Adding and editing device information.*

#### 3.1.3.7 Export PDF of Generated Report



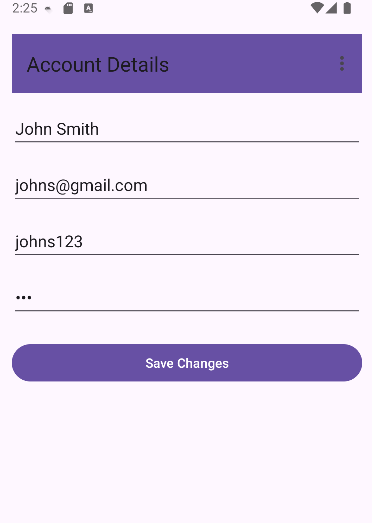
*Button to Export the dashboard to PDF format.*



*Success message to notify the user that the file is saved to the device’s Download directory.*

#### 

#### 3.1.3.8 Account Details

**

*User can update their account credentials*

#### 3.1.3.9 Admin Dashboard and Edit Default Appliance List

#### 

#### 

*The admin dashboard displays all accounts stored in the local database.*

*The admin can edit the default list of appliances for users to add to their device list.*

#### 3.1.3.10 Miscellaneous

#### 

*Sidebar menu for other features.*

**

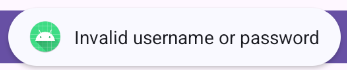
*Button to expand the Power Usage and Bill fragments in the dashboard.*

#### 

#### 3.1.3.11 Success/Error Messages

**

*Success message of the Electricity Consumption Calculator.*

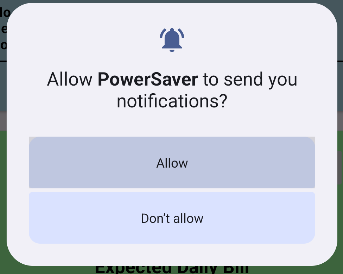
**

*Error message for invalid user credentials during login.*

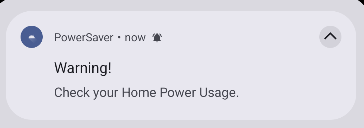
**

*Error message for invalid input of more than 24 hours for device usage.*

#### 3.1.3.12 Android Notifications



*PowerSaver requesting permission to send notifications.*



*A notification pop-up in device’s quick settings menu.*

## **Hardware Interfaces**

The app was primarily designed to be a pocket application running on any mobile device, such as smartphones and tablets.

### 3.2.1 Supported Device Types

* The app is compatible with any smartphone and tablet running Android OS 5.0 (API level 21) or higher.
* The devices must have sufficient memory (minimum 2GB RAM recommended) and processing power for smooth operation.
* The user must allow certain permissions for the app to function.

### 3.2.2 Logical Characteristics

* The app retrieves data from external sources to be displayed as charts and graphs.
* The app has access to email functionalities for sending the report to the user’s email address.
* The data type for electricity consumption is in kilowatts per hour (kWh).

### 3.2.3 Communication Protocols

* Wi-Fi is used for communication with external sources such as the APIs.

### 3.2.4 Control Interactions

* The app initiates data retrieval from the local database,
* The app may send notification alerts that will be displayed under the device’s quick settings menu if conditions are met.

## **Software Interfaces**

In this section, the connections between *PowerSaver* and other software components are described. This includes the databases, libraries, operating systems and tools it interacts with.

### 3.3.1 Operating System

The app runs exclusively on Android devices built-in with Android OS (version 5.0, API Level 21, and above). It utilizes Android APIs for communication, UI design, and system-level operations.

### 3.3.2 Database

The app uses SQLite as the local data storage for the device information and user configurations. The data that is retrieved includes the data from external APIs. The data that is going out of the app includes sending an email to the user’s email address and sharing of the generated report.

### 3.3.3 Libraries and Tools

1. **MPAndroidChart (Version 3.1.0)**: Generates charts for displaying household electricity consumption and historical electricity tariffs.

1. **Volley**: Handles API calls to fetch external data.

1. **Google APIs**: Google Auth Library for enabling OAuth 2.0 for secure authentication and Google API Client for interfacing with Google services.

1. **Material Design Library**: Provides modern UI components for consistent design and rendering of UI.

### 3.3.4 Integrated Commercial Components

**Third-party APIs:**

1. **SingStats API**: To retrieve the data of historical electricity tariffs for displaying in a chart.

1. **Gmail API**: Sending the report to the user’s email address.

### 3.3.5 Data Sharing

The SQLite database serves as the local data storage to transfer shared data across the application. It also maintains persistent data storage of the user's configurations and device information.

## **Communications Interfaces**

The app adopts various communication interfaces to maintain a responsive user experience. The communication requirements and protocols include data transfer mechanisms, message formatting and security considerations.

* **Hypertext Transfer Protocol Secure (HTTP/HTTPS)**: This is the main protocol used for communication between the app and third-party APIs. It ensures compatibility with most web services and secure data transmission using HTTPS.

* **Simple Mail Transfer Protocol (SMTP):** A standard protocol for the transmission of email over a network. This is used for sending emails via the Gmail server.

* **OAuth 2.0:** The standard protocol for secure authentication with Google services.

* **Message Formatting:** For transmitting incoming and outgoing messages such as API responses, which are JSON formatted data, and emails comprising SMTP messages with structured subject lines, HTML-formatted bodies and attachments.

* **Security and Encryption**: Secure data transmission of all external communications using HTTPS for encrypted data transfer. OAuth 2.0 for secure API access. AES for encrypting sensitive information, such as API keys and user credentials before storage.

* **Error Handling and Recovery:** To overcome communication failures, notifications are sent to users during API or device connectivity issues. The system is built with retry mechanisms for the successful retrieval of data. In offline mode, data is cached locally until the connection is restored for synchronization.

# **System Features**

In this section, the key system features of the application are listed below. For detailed documentation of the features, please refer to the Use Case Diagram and Use Case Description.

## Login

1. The system allows users to log in to their account using their username and password.
   1. The system must check if the information is correct by accessing the database
      1. If the input credentials are correct, the system redirects to the dashboard upon successful authentication.
      2. If the input credentials are invalid, an error message is displayed to inform the user.
      3. The user may choose to recover the password by hitting the 'Forgot Password' button.

## Registration

1. The user must be able to register a new account in the system.
   1. The system must display 4 required text fields for the user to enter registration information.
      1. The system must include a text field for Username (required).
      2. The system must include a text field for Password (required).
      3. The system must include a text field for Full Name (required).
      4. The system must include a text field for Email Address (required).
   2. The user must enter all the required fields before clicking the ‘Register’ button.
      1. The system must display an alert message if any required fields are empty.
   3. The system must verify the fields entered by the user.
      1. The system must check that the username is unique and does not exist in the database.
      2. The system must check that the email is in the correct format.
      3. The system must check that the email does not exist in the database.
      4. The system must check that the password has met the password requirements.
      5. The system must display an alert message for any fields that fail to meet the requirements.
   4. The system must create an account upon verification.
      1. The system must store the user’s account information in the database.
      2. The system must display a message upon a successful or unsuccessful account creation.

## Main Dashboard

1. The Main Dashboard will display the following components:
   1. Power Usage Fragment:
      1. The system will calculate the estimated daily consumption of the user based on the device list.
      2. The system must display the estimated daily consumption of the user devices.
      3. A stacked bar chart which displays the power usage of each device.
      4. A legend to display the name of the devices.
      5. Devices are colour-coded to distinguish between them easily.
      6. A threshold line that represents the average electricity consumption.
      7. A recommendation text must be displayed to inform the user of their current status of household electrical consumption.
      8. The system must display a “Manage Devices” button for the user to add or update device information.
   2. Expanded Power Usage Fragment:
      1. The system must display a pie chart of each device's consumption in percentage.
      2. A legend of the device is required to distinguish between the devices.
      3. The legend must include the name of the device and its usage.
      4. The system must display the device list in a table.
      5. The table must include the device name and its power consumption per day.
   3. Bill Fragment:
      1. The system must calculate the estimated cost of electrical consumption per day and per month.
      2. The system will display the estimated cost of electrical consumption for the day.
      3. The system will display the expected cost of electrical consumption for the month.
      4. The system will retrieve the latest tariff using SingStats API.
      5. The system will display the latest tariff in cents/kWh.
   4. Expanded Bill Fragment:
      1. The system will generate a line chart of the historical electrical tariffs.
      2. The system will retrieve the data using SingStats API.
      3. The system will display the annual average electricity tariff in cents/kWh.
2. The system must allow users to navigate to the device manager by selecting the “Manage Devices” option on the Side Menu.
   1. The device manager features include:
      1. Add Device
      2. Remove Device
      3. Edit Device
3. The system must allow users to change their account details and settings by selecting the “Account Details” option on the Side Menu.
   1. The personal detail include:
      1. Username
      2. Password
      3. Full name
      4. Email Address
   2. The system save the changes of the user credentials in the database.
4. The system must allow users to log out of the app by selecting the “Logout” option on the Side Menu.
   1. The user will be redirected to the login page.
5. The system must allow users to exit the app by selecting the “Close App” option on the Side Menu.
   1. The user will be logged out automatically and requires to log in to their account again.

## Account Management

1. The system must allow users to view and edit their account information (e.g. username, email, full name and password)
   1. The system must include an ‘Edit’ button for users to edit account information.
   2. The system must include input fields for entering the new account information.
2. The system must allow users to change or reset their password if they forget.
   1. Change Password:
      1. The system must provide input fields to change the password on the “Account Details” page.
   2. Forgot Password:
      1. The user must verify using their username and email to change their password.
      2. Upon successful verification, the user will be redirected to the “Reset Password” page.
      3. If the password is valid, the user will be redirected to the login page.
3. The system must verify that the user is logged in to access account information.

## Notifications Management

1. The user must permit *PowerSaver* to send notifications to the user's device.
   1. This setting can be toggled in the user’s device settings.
   2. The system will schedule notification and use the correct method accordingly.
   3. The notifications must be sent immediately with zero delay.

## Device Management

1. The system must allow users to view their device consumption in the Main Dashboard.
   1. The system must show the exact power usage of each device.
      1. The format for the power uasge is kWh.
   2. The system must allow the user to compare their total household power usage with the average household consumption.
   3. The system must display the information of the devices in the device list in a pie chart.
      1. The information includes:
         1. Name of the device
         2. Power Usage(in kWh)
         3. Partition of usage in total usage (in %)
2. The system must allow users to access their appliance list by selecting “Manage Devices” in the Side Bar button.
   1. The system allow user to see the information of each appliance:
      1. The data includes:
         1. Name of the device
         2. Average Power Usage (in kWh)
         3. Duration (in Hour/Day)
   2. The system allow user to edit the information of each appliance:
      1. The data includes:
         1. Average Power Usage
         2. Duration
      2. The system will save the data accordingly to the local database.
   3. The system must allow the user to remove an appliance by selecting the “Remove” button under the appliance name.

## Adding New Device

1. The system must allow the user to add a new device to the device list by hitting “+” in the device manager.
   1. The user must enter the name of device.
   2. The user must fill in the average power consumption of the device.
   3. The user must fill in the duration of device usage.
   4. Once all conditions are met, the system will add the appliance to the user’s device list.

## Electricity Consumption Calculator

1. The system must calculate the total electricity consumption based on the devices in the device manager.
   1. The system must validate that all required fields are entered in the device management menu before calculation.
      1. The system must display an error message if there are missing fields.
   2. The system must perform calculations using the correct algorithm and formula.
2. The system must calculate the expected cost for the total electricity consumption.
   1. The system must use the latest electricity tariffs for calculations.
   2. The system must calculate the total cost of electricity usage for each device.
   3. The system must determine the devices with the most electrical consumption.
   4. The system must calculate the total electricity bill for the month.
3. The system must display the average monthly household electricity consumption.
   1. The values must be retrieved from the EMA dataset.
   2. The system is not required to perform calculations for this metric.
4. The system must display a confirmation notification for updated calculation changes.

## Export Generated Report

1. The system must allow user to generate a report of their total electricity consumption.
   1. The layout of the report must be consistent across all users.
   2. The report must state the electricity consumption per appliance.
   3. The system must display an error message if the system fails to generate the report.
2. The system must display a graphical representation of the data.
   1. The system must provide a preview of the report with the charts.
3. The system must be logged into his account to generate or view report.
4. The system must allow users to download the report into different file formats.
   1. The recommended file format is PDF.
   2. The system must display a message for successful download of the report.

## Admin Dashboard

1. The system must display the list of users registered in the local database.
2. The system must allow the admin to edit the default appliance list.
3. The system must display input fields for entering the appliance information:
   * 1. Appliance name
     2. Average power usage of the appliance (in kWh)
     3. Duration (in Hours/Day)
4. The system must display buttons for the admin to edit or remove the appliance from the default appliance list.
5. The default appliance list must be stored in the local database.
6. The default list of appliances and their information must be displayed in the regular user’s device manager.

# **Other Nonfunctional Requirements**

## **Performance Requirements**

1. The system must be able to run with little downtime and it does not crash.

1. The system must return result for any interaction within 3 seconds.

## Usability Requirements

1. 90% of the users must be able to generate a report of their total electricity consumption within 30 seconds.
2. The notification messages must be clear, precise and helpful for user clarity.

## **Security Requirements**

1. Each account cannot access, or manipulate the private data of another account.

* 1. Private data include:

* + 1. Device list

* + 1. Account Details

1. The system must adopt two-factor authentication during user registration and user login.

1. The system must mask the password field to preserve user privacy.

1. The account password for each user stored in the database must be encrypted.

## Reliability Requirements

1. The recommendations for reducing electrical consumption must be sound and accurate.

1. The estimated total cost and total electrical consumption must have at least 95% accuracy.

1. During application downtime, it must not interfere with user processes and allow users to continue to perform application functions.

## **Business Rules**

1. The system must store the information in a database for scalability.

* 1. The information includes:

* + 1. Private data of each account

* + 1. Average power usage of each device

* + 1. Electrical Tariff

1. The system must maintain the data integrity.

# **Other Requirements**

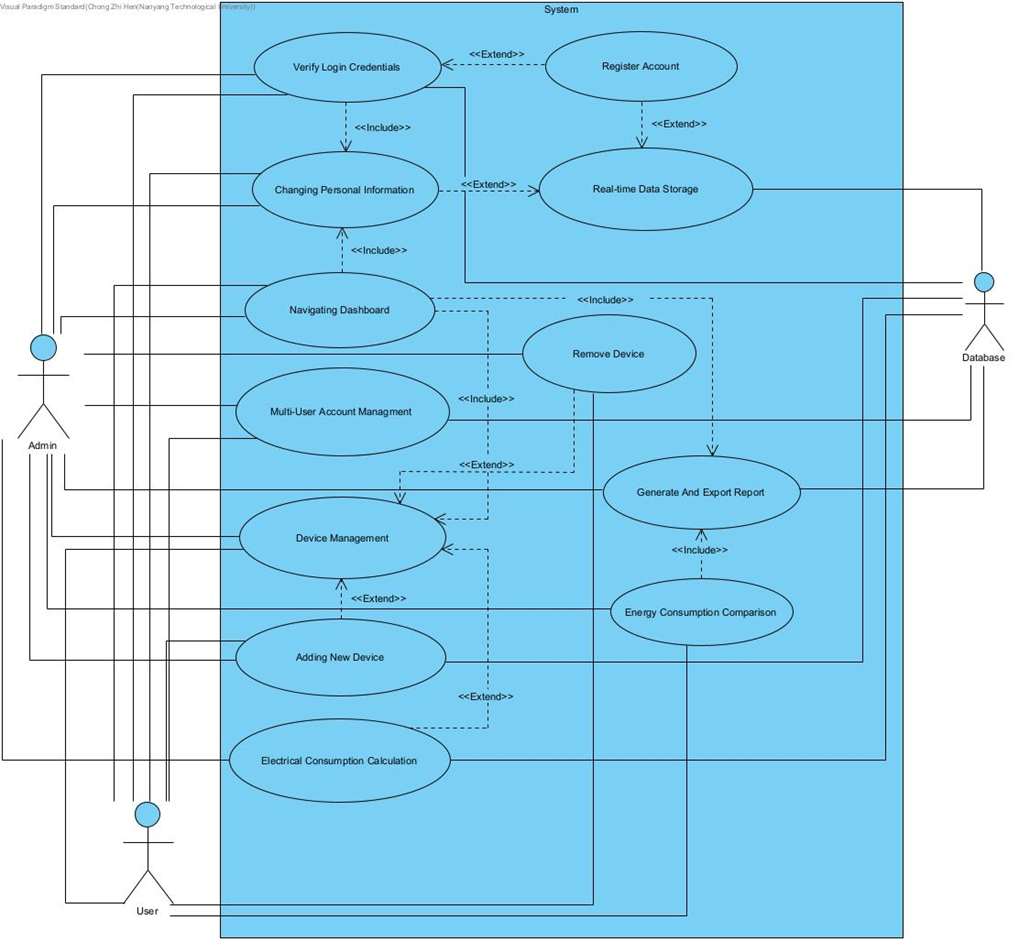
## Data Dictionary

The purpose of the data dictionary is to define the commonly used terms, phrases and abbreviations in the documentation and code implementation. This provides clarity and a reference source for other readers.

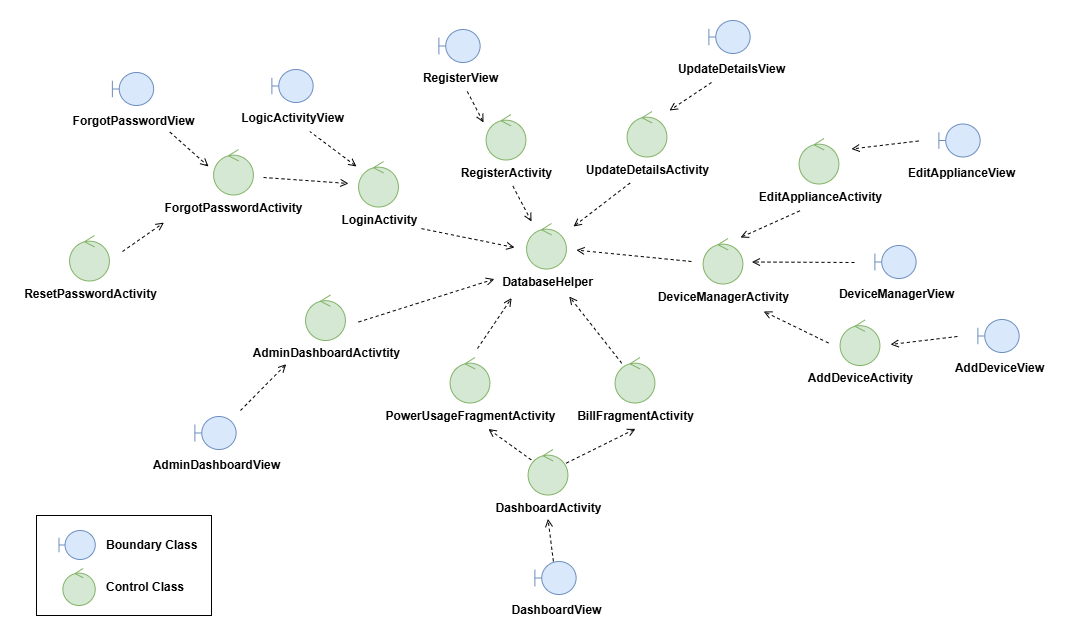
| **Term** | **Definition** |
| --- | --- |
| System | The system is the Power Saver application powered by the Android operating system. |
| Front-end | The application’s user-facing elements are in the User Interface (UI). |
| Back-end | The backbone of the application such as the code and database are hidden from the user. |
| IDE | Integrated Development Environment (IDE), which is the software application used for the development of the app. |
| XML | Extensible Markup Language (XML), is a markup language used for the front-end development of the app. |
| API | Application Programming Interface (API), is an application from an external source that is retrieved for certain functionalities within the app. |
| User | A user is a person who uses the application to optimise the electricity use in his/her household. |
| Side Menu | The side menu is the set of links that can be accessed by clicking “☰” button on the top right-hand corner of the dashboard screen. |
| Notification | Notification in this application informs the user of the status of their electricity consumption. It is displayed as a pop-up alert that can be viewed under the Android device’s quick settings menu. |
| Device/Appliance | An appliance is a household device that consumes electricity. |
| Usage | The usage of an appliance is power usage over a period. |
| Consumption data | Consumption data is the user's electricity consumption information, which includes the name, average power, and duration. |
| Device/Appliance List | A list of devices with the name, energy consumption and number of hours per device. |
| Bill | Bill is the estimated cost of electricity consumption for the month. It provides the expected amount on the consumer’s bill. |
| SingStat | Represents the *Singapore Department of Statistics (DOS)* website. |
| EMA | Abbreviated as the *Energy Market Authority (EMA)* *of Singapore*. |

# 7. Appendix A: Analysis Models

## 7.1 Use Case Diagram



## 7.2 Key Boundary and Control Classes

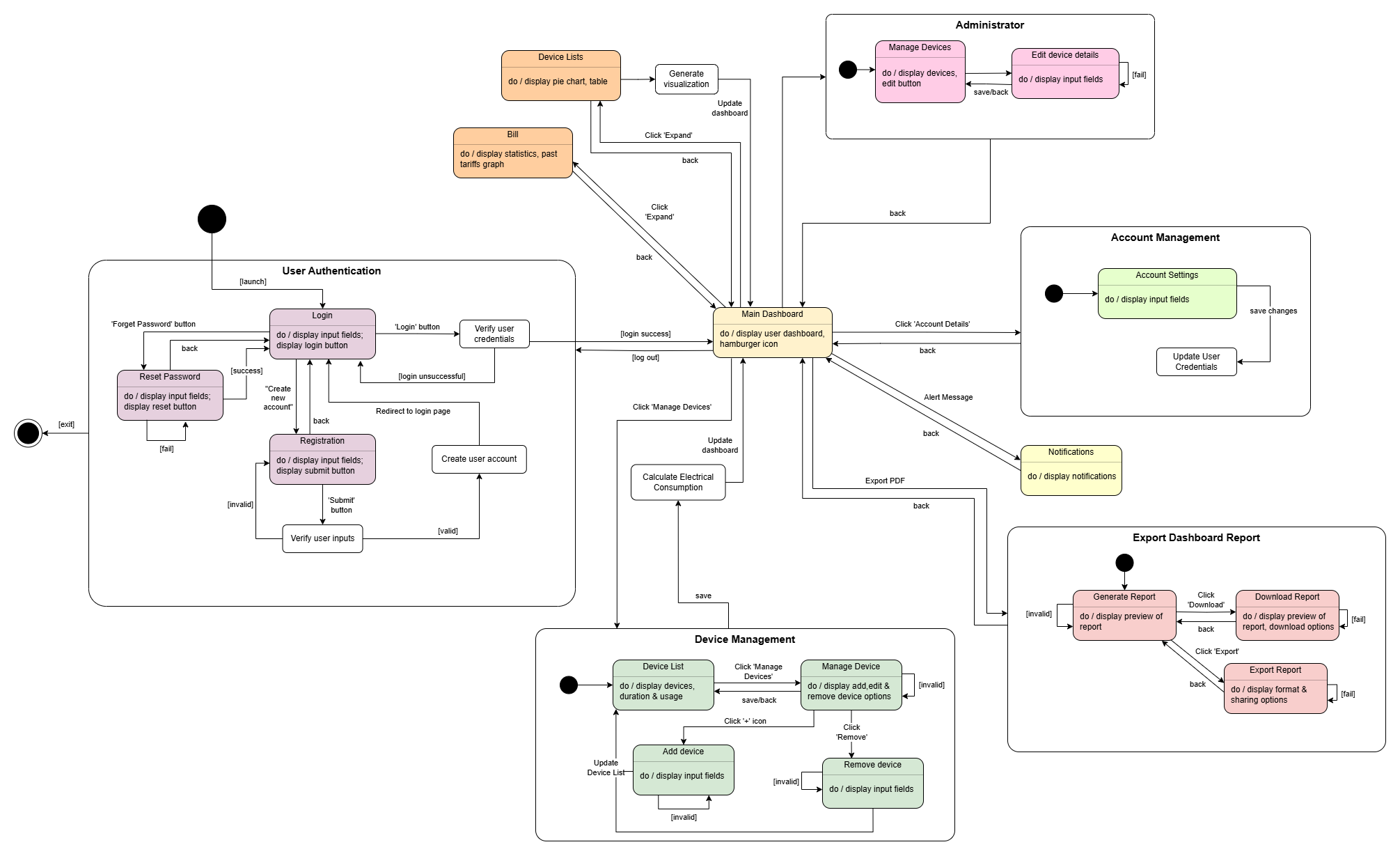


### 

## 7.3 Class Diagram

### 

## 7.4 Dialog Map



Source: http://www.frontiernet.net/~kwiegers/process\_assets/srs\_template.doc