Cairo University
Faculty of Computers and Artificial intelligence
Department of Computer Sciences

SolarEase

SolarEase Assistant for Environmental Efficiency

Supervised by

Prof. Abeer El-Korany

Implemented by

IDs	Names
20200520	Mariam Ashraf Amin
20200521	Mariam Saeid Shawky
20201085	Sarah Ahmed Mahmoud
20200619	Hoda Shafek Ahmed
20201040	Eman Mustafa Fathy

Graduation Project
Academic Year 2023-2024
Final Documentation

Table of Contents

Table of Figures

Chapter 1: Introduction

1.1 Motivation

Our planet is experiencing rising temperatures, extreme weather events, melting ice caps, and a loss of biodiversity. These are not distant threats; they are happening now. Egypt possesses abundant solar energy potential to mitigate climate change however people are not aware enough to get the best use of this not exploited solar energy. The key challenge lies in increasing public awareness and promoting solar energy adoption which ultimately aids Egypt in its transition to a greener future.

This endeavor aligns with Egypt efforts to combat climate change as The Egyptian government is presently in the process of revising its strategy for sustainable development, aligning with the Vision 2030 initiative, and placing a strong emphasis on advancing renewable energy solutions. Our project addresses these challenges by guiding users to select the optimal solar system providing personalized solar system specifications, making it easier for users to embrace solar energy confidently.

1.2 Problem Definition

Awareness Challenges:

One of the primary motivations behind our solar energy project stemmed from the lack of public awareness regarding the significant environmental benefits of solar systems and their positive impact on climate change. People often underestimate the potential of solar energy in reducing greenhouse gas emissions and shifting away from fossil fuels. There's a critical need for clear and accessible information on solar system installation, as well as finding trustworthy suppliers and installers. Many individuals are also unaware of available government incentives and permits, hindering the broader adoption of solar energy. Additionally, escalating electricity costs can pose a financial burden on consumers, Solar panels generate electricity, allowing users to rely less on traditional grid power.

Facilitation Challenges:

Complex calculations and the difficulty in assessing whether solar energy suits specific user needs and circumstances pose significant hurdles in adopting solar technology. Calculating energy savings, payback periods, and system output can be intimidating for non-experts. Moreover, tailoring solar solutions to individual requirements is a complex task.

1.3 Project Objective (suggested solution)

The objective of our project is to mitigate climate change by automating the process of reducing carbon emissions from conventional electricity sources. The primary challenge is the limited public awareness regarding the environmental benefits of solar systems and their significant positive impact on mitigating climate change. Additionally, there is difficulty in assessing whether solar energy is suitable for users and provides long-term financial benefits.

To achieve this objective, we will implement a system that empowers individuals to make informed solar energy choices, contributing to a greener and more sustainable future.

by providing these services: -

- Assist users from the initial stages of solar adoption decision-making to the selection of certified installation companies. Our platform will provide comprehensive guidance to ensure users can easily navigate through the complex process of adopting solar energy.
- Provide personalized recommendations on system size, cost savings, and payback period based on individual needs. By analyzing specific user data and energy consumption patterns, we will offer customized advice.
- Predict solar system productivity using AI technology to optimize efficiency.
- Offer information and guidance through a user-friendly chatbot interface. The chatbot will serve as an interactive assistant and offer step-by-step support to users throughout their solar adoption journey.
- Establish an online marketplace for users to access the latest solar market prices, facilitate buying or selling of solar components by posting them in the marketplace, and allow users to save favorite products and posts for quick access and comparison.

The potential impact of the project includes reducing carbon emissions, preserving the environment, and contributing to a sustainable future by promoting solar energy adoption. By making solar energy more accessible and understandable, we aim to encourage more people to switch to renewable energy sources, thereby playing a crucial role in combating climate change and promoting environmental sustainability.

1.4 Gantt chart of project time plan

Start Date	End Date	Days
1-Oct	21-Oct	21
22-Oct	28-Oct	7
29-Oct	7-Nov	10
8-Nov	14-Nov	7
8-Nov	30-Nov	23
1-Dec	7-Dec	7
8-Dec	31-Dec	24
1-Jan	10-Mar	70
1-Feb	10-Feb	10
1-Mar	2-Mar	2
3-Mar	17-Mar	15
12-Mar	5-May	55
6-May	10-May	5
11-May	20-May	10
15-Jun	26-Jun	12
27-Jun	1-Jul	5

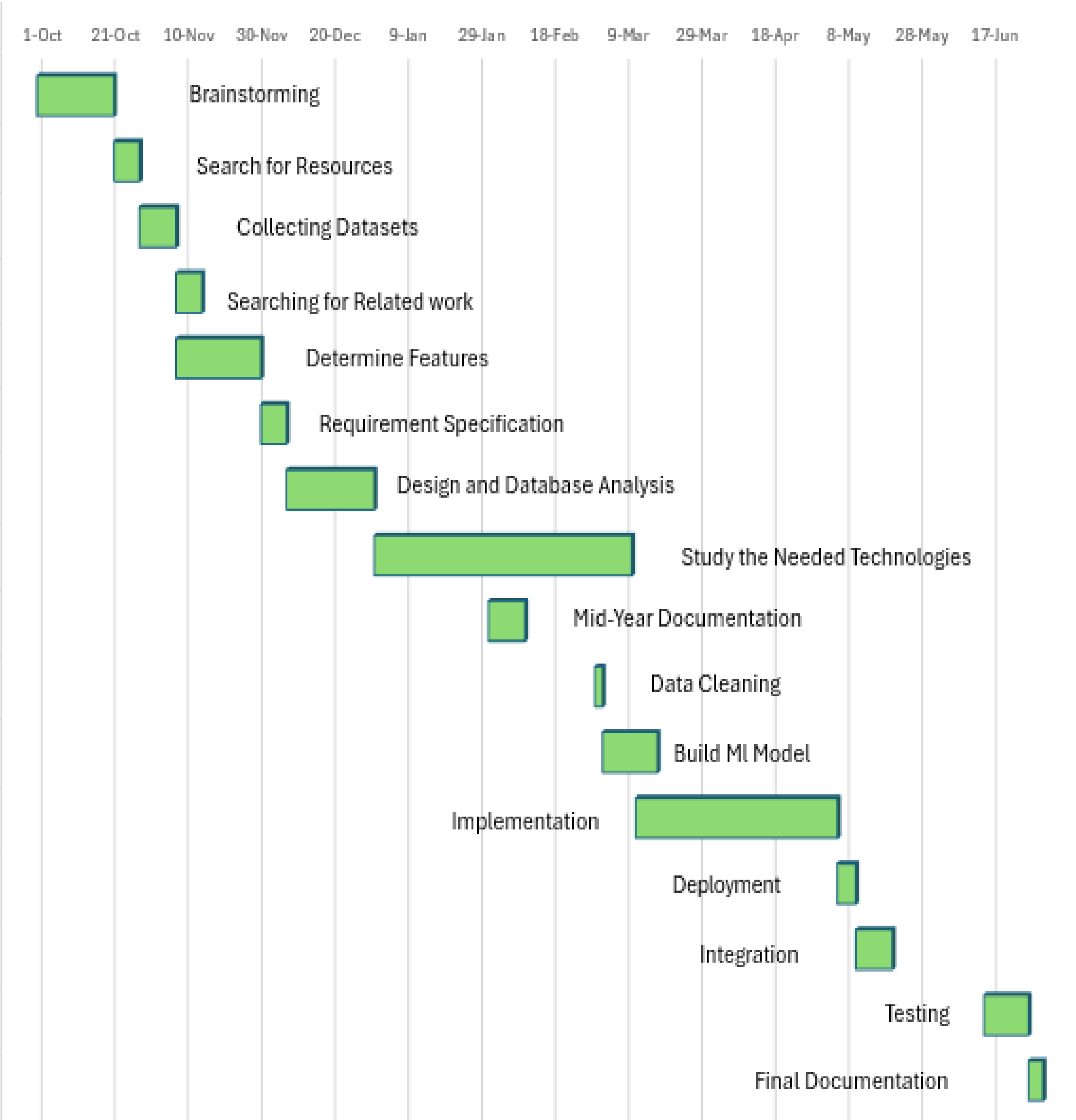


Figure 1 Gantt Chart

1.5 Project development methodology

For our graduation project, we have chosen the Waterfall methodology as the most suitable framework for developing our solar adoption assistance application. Waterfall is a linear and sequential software development approach that emphasizes a structured and disciplined process, where each phase must be completed before the next one begins. This methodology allows us to clearly define requirements and objectives from the start, ensuring that each stage of development is thoroughly planned and executed.

The Waterfall methodology fits well with our project as it allows us to meticulously plan and document each phase, ensuring a clear path from requirements gathering to system deployment. This structured approach involves the following phases:

1) Requirement Analysis

In this initial phase, we gathered all the detailed specifications for each service our application provides. For user assistance and personalized recommendations, we considered all scenarios of solar installation to determine suitability, we gathered peak hour data and the latest electricity tariff rates from January 2024 to ensure accurate and relevant recommendations for each user.

For productivity predictions, we gathered hourly and daily meteorological data sets from NASA power for all cities of Egypt to train our machine learning model. For the chatbot, we collected the most common questions and their appropriate answers to ensure comprehensive guidance. In the marketplace, we considered the latest solar market prices to facilitate informed buying and selling decisions. This thorough requirement analysis ensured that we had a comprehensive understanding of what the system needed to achieve.

2) System Design

Based on the requirements, we created detailed design documents outlining the architecture, components, classes, sequences for all scenarios, entities and interfaces of the application. This phase allowed us to plan the technical aspects and ensure all components would work together seamlessly.

3) Implementation

During this phase, we developed the actual code for each component of the application. The Waterfall methodology ensured that we had a clear plan to follow, making the development process more straightforward and organized. For example, we coded the machine learning models, the frontend, and the backend for each service in a sequential manner.

4) Integration and Testing

Once all components were implemented, we integrated them into a complete system. We first integrated the machine learning models with the backend, and then we integrated the backend with the frontend. Rigorous testing was conducted to identify and fix any issues, ensuring the application met the predefined requirements and functioned as expected.

5) Deployment and Maintenance

After successful testing, the application was deployed to the production environment making it available for use by intended users. This phase included tasks such as system configuration, server setup, database upload, and ensuring a smooth operation into the production environment.

By Adopting the Waterfall methodology, our goal was to achieve a high-quality solar adoption assistance application that meets the needs and expectations of our users. This approach enabled us to improve our software development skills and knowledge, as well as work in a structured and disciplined manner.

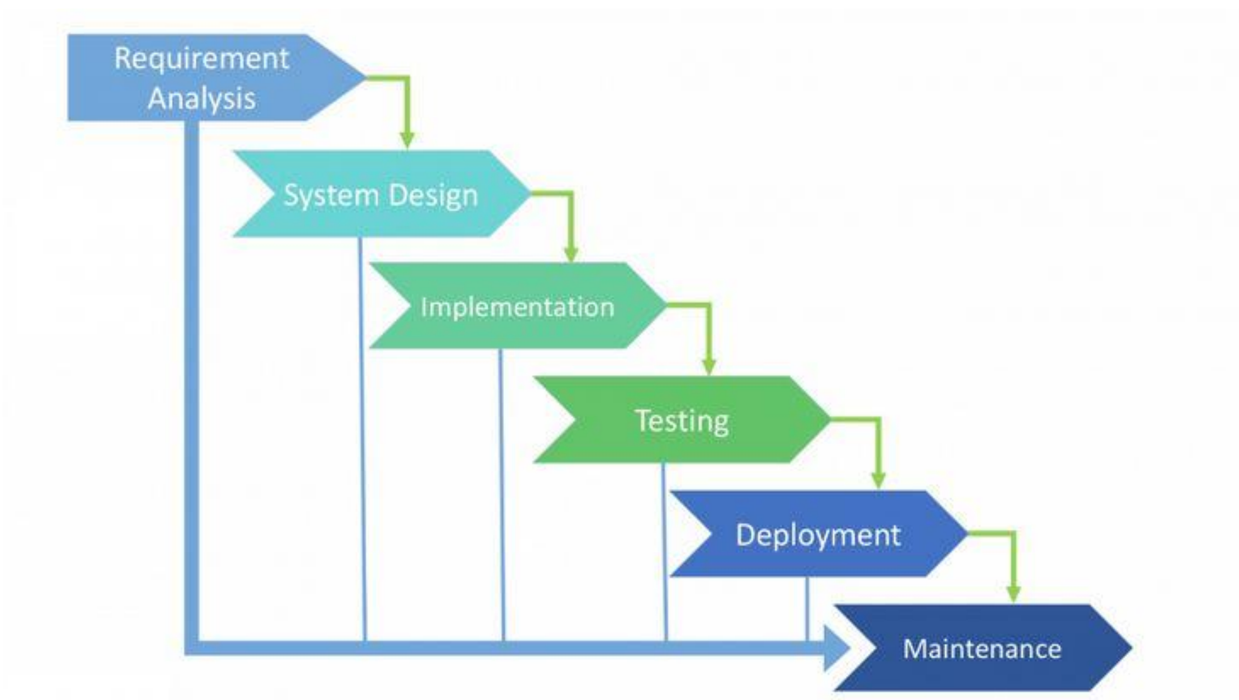


Figure 2 Waterfall Methodology

1.6 The used tools in the project (SW and HW)

1.6.1 Software Tools

Flutter

An open-source UI toolkit used for building natively compiled applications for mobile, web, and desktop from a single codebase. It was used to develop the frontend of our mobile app.

Visual Studio Code

A source-code editor made by Microsoft for Windows, Linux, and macOS. It was used as the primary IDE for developing the Flutter front end.

NET Web API Framework

A framework for building HTTP services that can be consumed by a broad range of clients including browsers and mobile devices. It was used to develop the backend of our application.

Microsoft Visual Studio

An integrated development environment (IDE) from Microsoft. It was used for developing the .NET Web API backend.

SQL Server

A relational database management system developed by Microsoft. It was used to manage and store the application's data.

Google Colab

An online platform that provides a Jupyter notebook environment. It was used for training and developing models.

Git/GitHub

Git is a distributed version control system, and GitHub is a web-based platform. They were used for source code management and collaboration among team members.

Draw.io

A web-based tool used to create all the diagrams in the system.

Canva

Online graphic design platform, used for creating graphical assets and project visual content.

Monster ASP.NET Cloud Platform

Utilized for hosting and deploying the backend of the application on a cloud computing platform.

NLTK

Used for text processing tasks such as tokenization and stemming in our chatbot. Includes libraries like nltk.tokenize and nltk.stem.PorterStemmer.

FastAPI

A modern, fast web framework for building APIs with Python, used for serving our machine learning models and our chatbot.

Huggingface.co Cloud Platform

A cloud-based platform offering tools and libraries. It was used to upload, manage, and integrate our machine learning models and our chatbot.

<u>Machine learning model Tools</u>

SVR (Support Vector Regressor)

A machine learning model used for predicting solar irradiance, using GridSearchCV for hyperparameter optimization.

Scikit-Learn

Python library for machine learning provides tools for model selection, preprocessing (scaling), and evaluation (metrics).

Feature_Engine

Python library for transforming cyclical features (time-related) using sin/cos transformations.

Joblib

Python library for saving and loading objects efficiently.

1.6.2 Hardware Tools

Personal Laptop

Used for development and testing.

Remote Servers

Utilized for deploying our application and our machine learning models. These servers host the ensure high availability, and scalability.

This comprehensive list includes both software and hardware tools used throughout the project, including those specifically utilized for the machine learning model described.

1.7 Report Organization (summary of the rest of the report)

The rest of the document discusses the different phases that describe and illustrate the characteristics of the project.

Chapter 2: Related work

Here, we compare the closest examples of the project to identify the main differences between them and our project, using references to others as necessary.

Chapter 3: System Analysis

This chapter includes

- 3.1 Project Specification
  - 3.1.1 Functional requirement
    - Detail the specific functions that the system must perform.
  - 3.1.2 Non-functional requirement
    - Indicate the performance and quality attributes the system must meet.
- 3.2 Use case Diagrams: Illustrate the interactions between stakeholders and the system and depict various use case scenarios.

Chapter 4: System Design

This chapter includes all diagrams that describe the relationships between classes and components in the system, as well as how data is stored and retrieved in the database.

- 4.1 System Component Diagram
- 4.2 System Class Diagrams
- 4.3 Sequence Diagrams
- 4.4 Project ERD
- 4.5 System GUI Design

Chapter 5: Implementation and Testing

This section includes reports on the implementation of hourly and daily solar irradiance prediction models. It will cover testing scenarios and results for both machine learning models and the backend.

References

This section lists all the sources cited in the document, including academic papers, datasets, and methodologies used in the project, ensuring credibility.

Chapter 2: Related work

Features \ Solutions		PV Output WebSite	Energy Sage WebSite (2009)	PVWatts WebSite (2012)	Solar Reviews WebSite (2012)	Solar Market Egypt WebSite (2018)	Know Solar App (2021)	Solar .com Website (2023)	SolarEase (Our App)
Decision Making	SolarCalculator Solar System Size & Cost Calculator		✓		✓			✓	✓
Decision Making	SolarSaving Cost Savings & Payback period		✓		✓			✓	✓
Decision Making	SolarEnergyPredictor Predict Solar Panels Production (Hourly – Daily)			✓			✓		✓
Installation	SolarInstallerFinder Solar Companies & nearby companies		✓		✓	✓		✓	✓
Installation	SolarMarketPrices Products Price & Favorite Products		✓		✓	✓		✓	✓
Installation	OnlineTradeMarketPlace Favorite Posts								✓
Awareness	SolarGreenImpact Environmental Savings	✓							✓
Awareness	SolarChatBot								✓

Figure 3 Related Work

Related Work Description

- **KnowSolar APP and PVWatts Website**: Estimate the solar energy that will be produced based on system size.
- **PV Output Website:** Collects solar power data from users and display the environmental savings that result from using solar energy.
- **Solar Market Egypt Website:** Provides information on the Egyptian solar market that include solar products prices and companies' locations in addition to nearby companies.
- **Solar Reviews, Energy stage and Solar.com Websites:** provide users with suitable system size based on their needs, the savings, nearby companies, products prices and information about solar energy.

Chapter 3: System Analysis

3.1 Project specification

3.1.1 Functional Requirements

1) Calculations:

Calculate System Size: Provides the user with the appropriate solar system size and roof space needed for this system, including solar panels, inverter, and battery capacity, based on user-provided monthly electricity consumption and location.

Calculate Cost: Provides the user with the total cost of the recommended solar system size. The cost calculation must consider the expenses associated with solar panels, inverters, batteries, installation, and any additional components.

Calculate Financial Savings: Provides the user with the savings resulting from the recommended solar system size, considering the user's monthly, yearly, and 25-year savings. It also provides the user with a payback period for the solar system.

Calculate Environmental Savings: Provides the user with the annual reduction in CO2 emissions that would be saved by installing a solar system, based on solar system size.

2) Prediction:

Predict solar energy: provide the user with values every 3 hours and daily, spanning a five-day predictions of panel output. This prediction should be based on the user's solar system size and user's location, as well as forecasted solar irradiance data.

3) Installation Guide:

Find Solar Installers: Provides the user with a list of certified solar installation companies, ordered by the nearest to the user's location.

Solar Market Prices: Provides the user with a price list of solar products including panels, inverters, and batteries from various brands and capacities. Users can add any products to their favorite products page for easy reference and comparison.

Online Trade Marketplace: Users can post solar products for sale, whether new or used. These products must be displayed to other users interested in purchasing them. The marketplace should allow sellers to provide their contact information, such as a phone number, within their product posts to allow potential buyers to communicate with them. Users should be able to search, view, and add any post to their favorite posts page for easy reference and comparison.

4) Chatbot:

Chatbot: The user can chat and interact with a rule-based chatbot that provides information on general solar-related topics, offers advice, and assists in facilitating the installation process.

5) Posts Filtration:

Filter Post: Admin can review and filter posts before publishing in the marketplace and reject the posts that violate the standards.

3.1.2 Non-functional Requirements

Usability

The user interface should be user-friendly, with clear navigation and a straightforward design, enabling users to interact with the application's features without confusion.

Performance

The chatbot is required to exhibit rapid and responsive behavior. Furthermore, the energy prediction output should be generated promptly, and the overall responsiveness of the application must be swift.

Scalability

The application should be capable of handling a growing user base without sacrificing its performance, ensuring its ability to adjust to evolving demands over time.

Reliability

The solar calculator and energy predictor must be reasonable and user-customizable, ensuring that they are both accurate. The application performs functions correctly. Users can rely on the software to work as expected without unexpected failures or disruptions.

Data Integrity

Measures should be in place to prevent data loss, corruption, or unauthorized alterations, with regular backups and reliable recovery mechanisms.

3.2 Use case Diagram

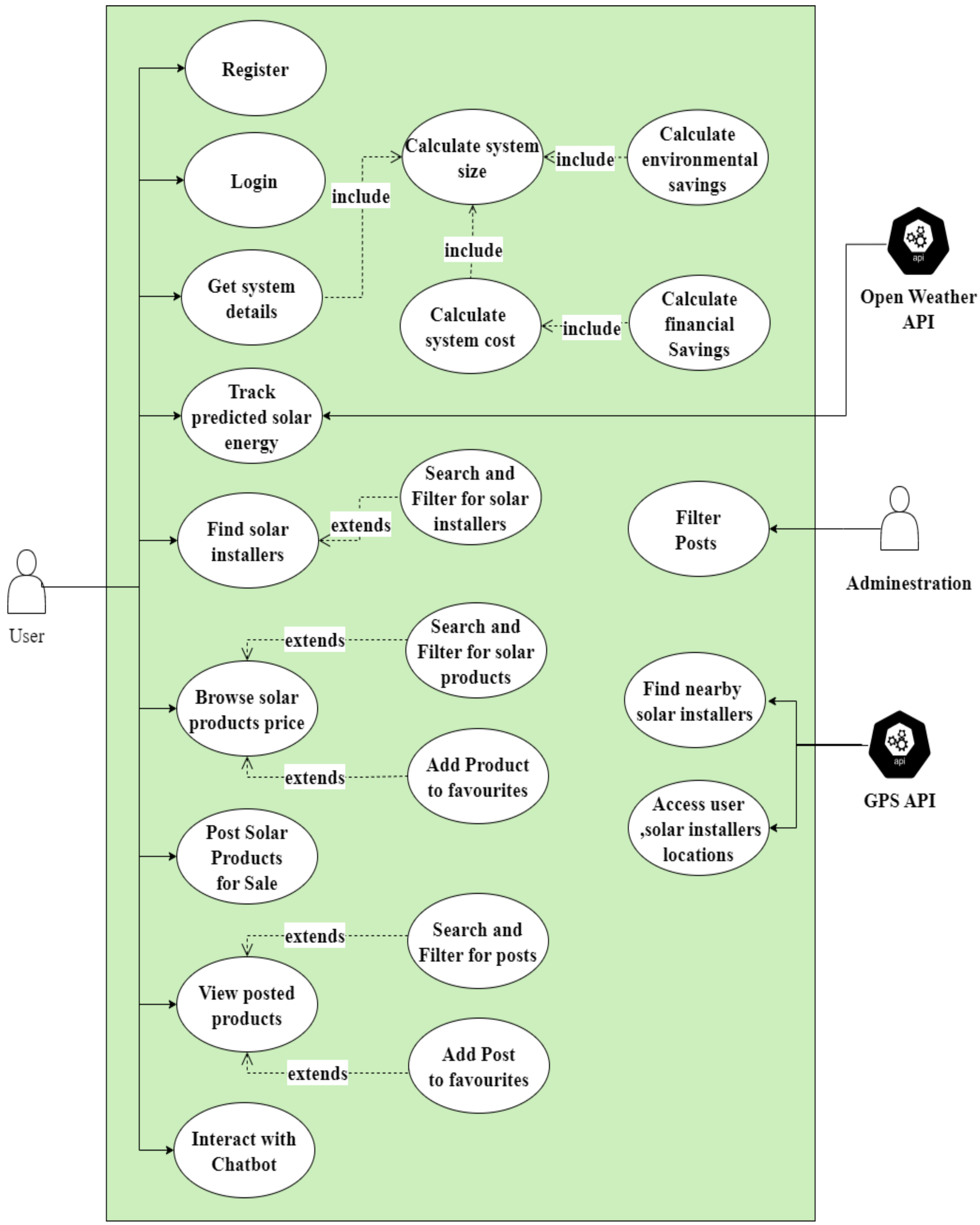


Figure 4 Use case Diagram

Chapter 4: System Design

4.1 System Component Diagram

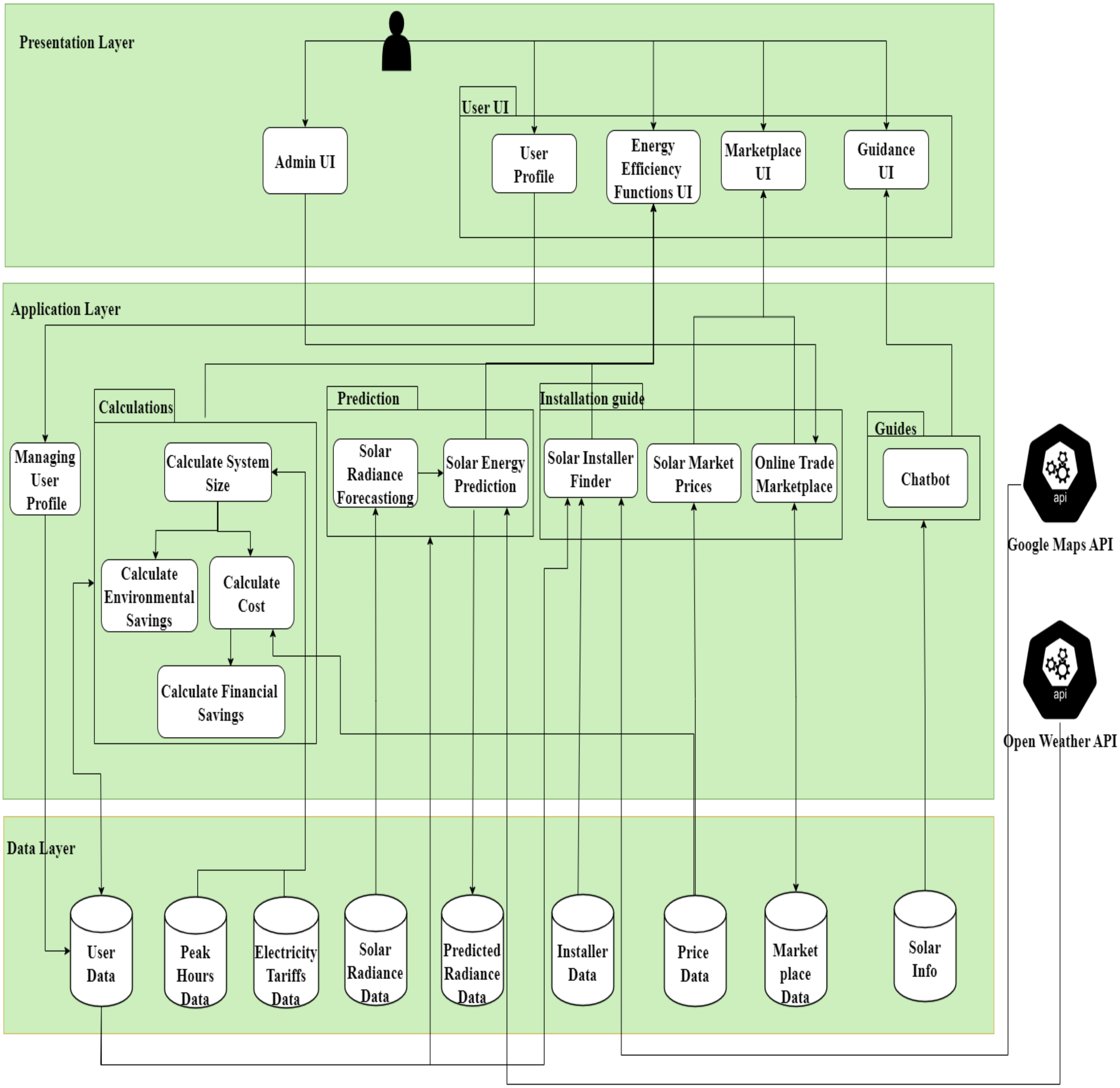


Figure 5 System Component Diagram

System Component Description

A 3-tier architecture diagram represents the system's structure by dividing it into three primary logical layers: the presentation, the application, and the data layer. This graphical representation illustrates how these layers interact together to create a cohesive system.

- At the pinnacle of the diagram stands the presentation layer, which is responsible for presenting the user and admin interface to end-users, it is divided into one module and one component (Admin UI). The module contains the User UI of main features (User Profile, Energy Efficiency Functions, Marketplace, and Guidance), it allows the user to modify his profile, post a product to sell in the marketplace, input needed info that helps to determine system size, cost, and saving benefits, track prediction of panels output, show the nearest installers, chat with chatbot.

- Positioned at the center of the architectural diagram, the application layer takes on a pivotal role. It is responsible for housing the system's fundamental business logic and functionalities, which include processing user requests, calculations, and managing data. **It consists of:**

  1. Managing the user profile component which contacts with user profile and user data.
  2. Calculate system size, system cost, financial savings, and environmental savings.
  3. Solar installer finder component that finds the nearest company based on user location.
  4. Solar system pricing component that shows the price of each solar component.
  5. Predict the solar panels' productivity based on system size and forecasting solar radiation.
  6. Online trade market that manages buying and selling solar products.
  7. Providing information for the user using chatbot.

- Situated at the bottom of the architecture diagram, the data layer plays the vital role of storing and managing the data essential for the system. It comprises the database server, where data is stored and retrieved, in addition to any related data storage components.

4.2 System Class Diagrams

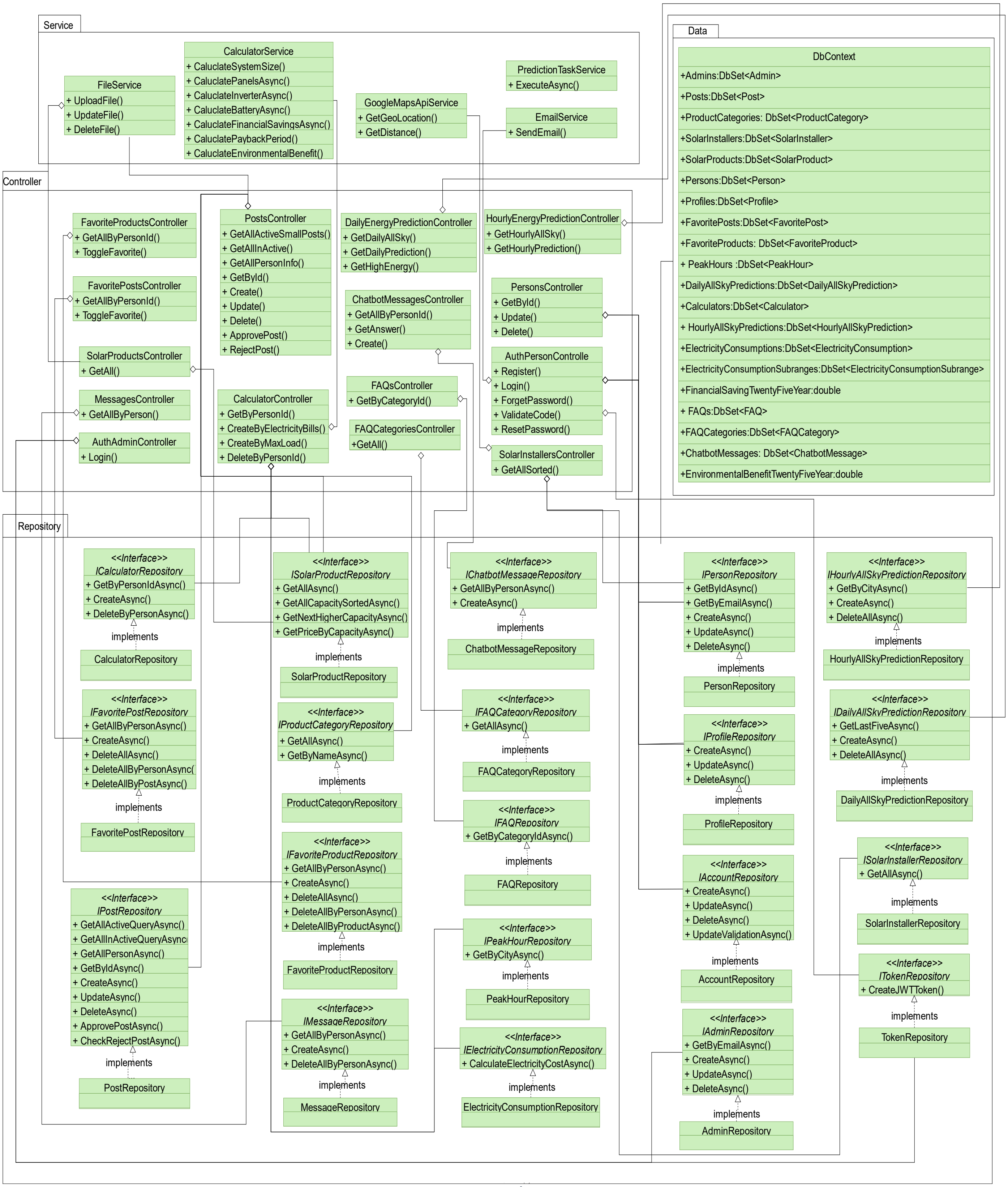


Figure 6 System Class Diagrams

4.3 Sequence Diagrams

4.3.1 Register Sequence

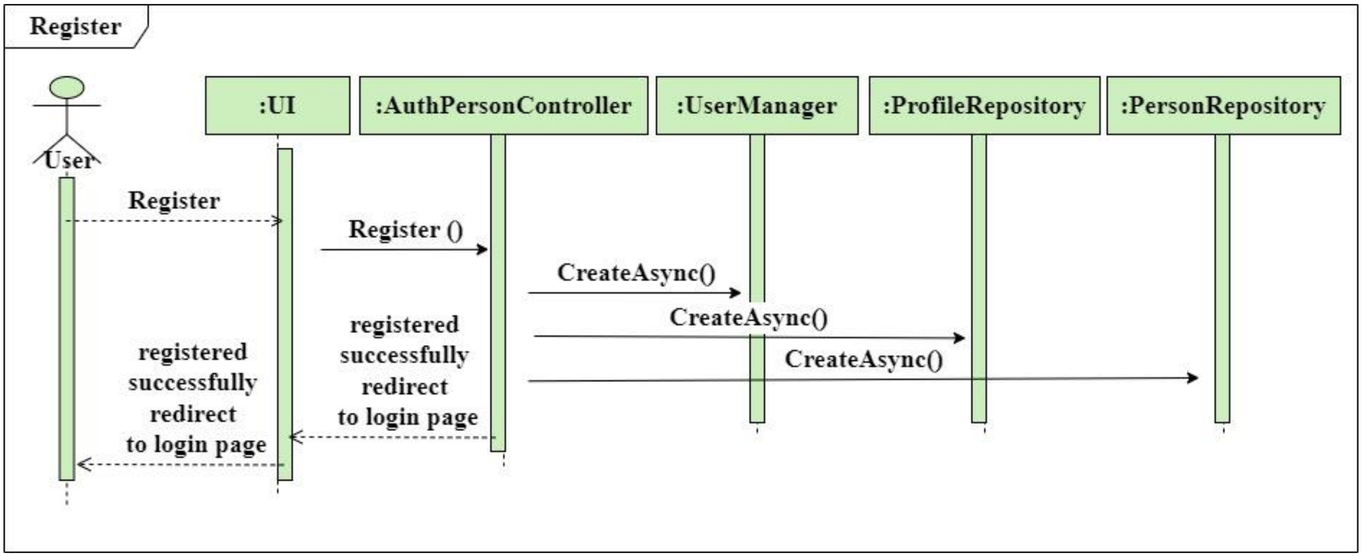


Figure 7 Register Sequence

4.3.2 Login Sequence

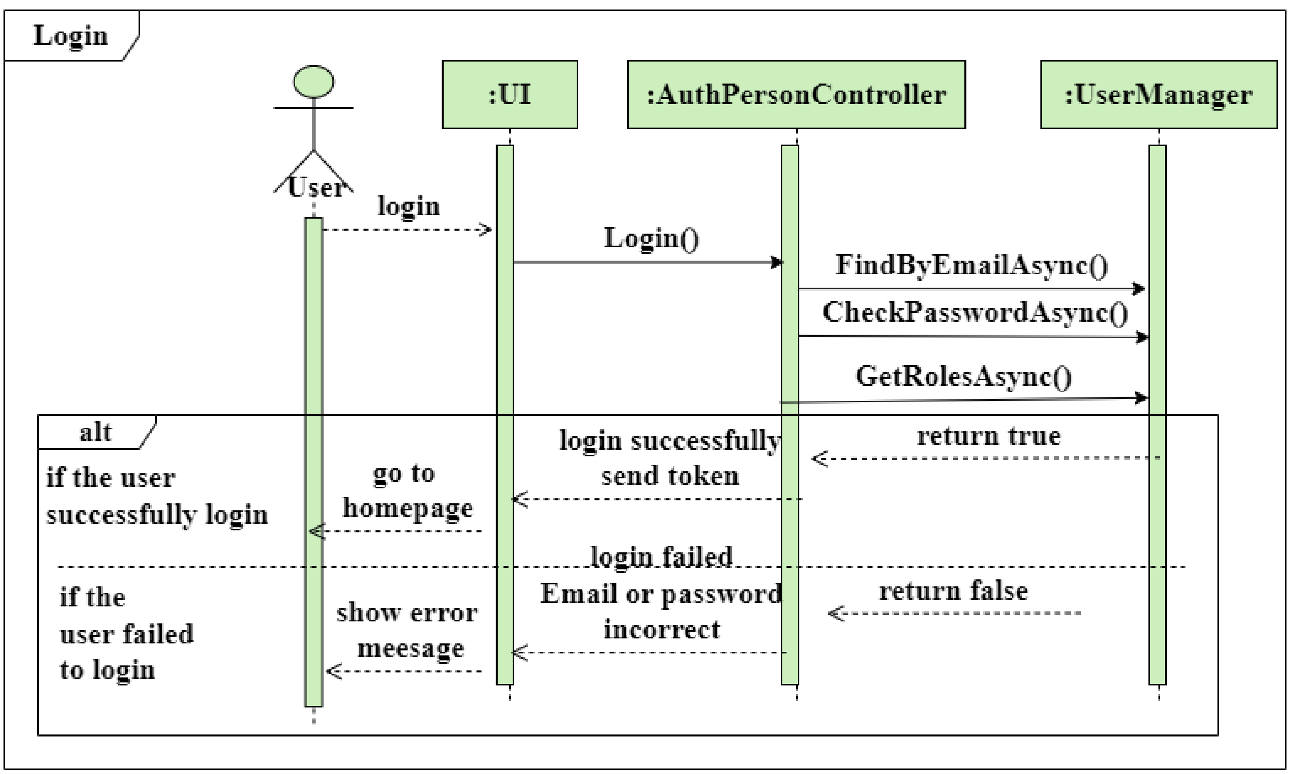


Figure 8 Login Sequence

4.3.3 Calculation Sequence

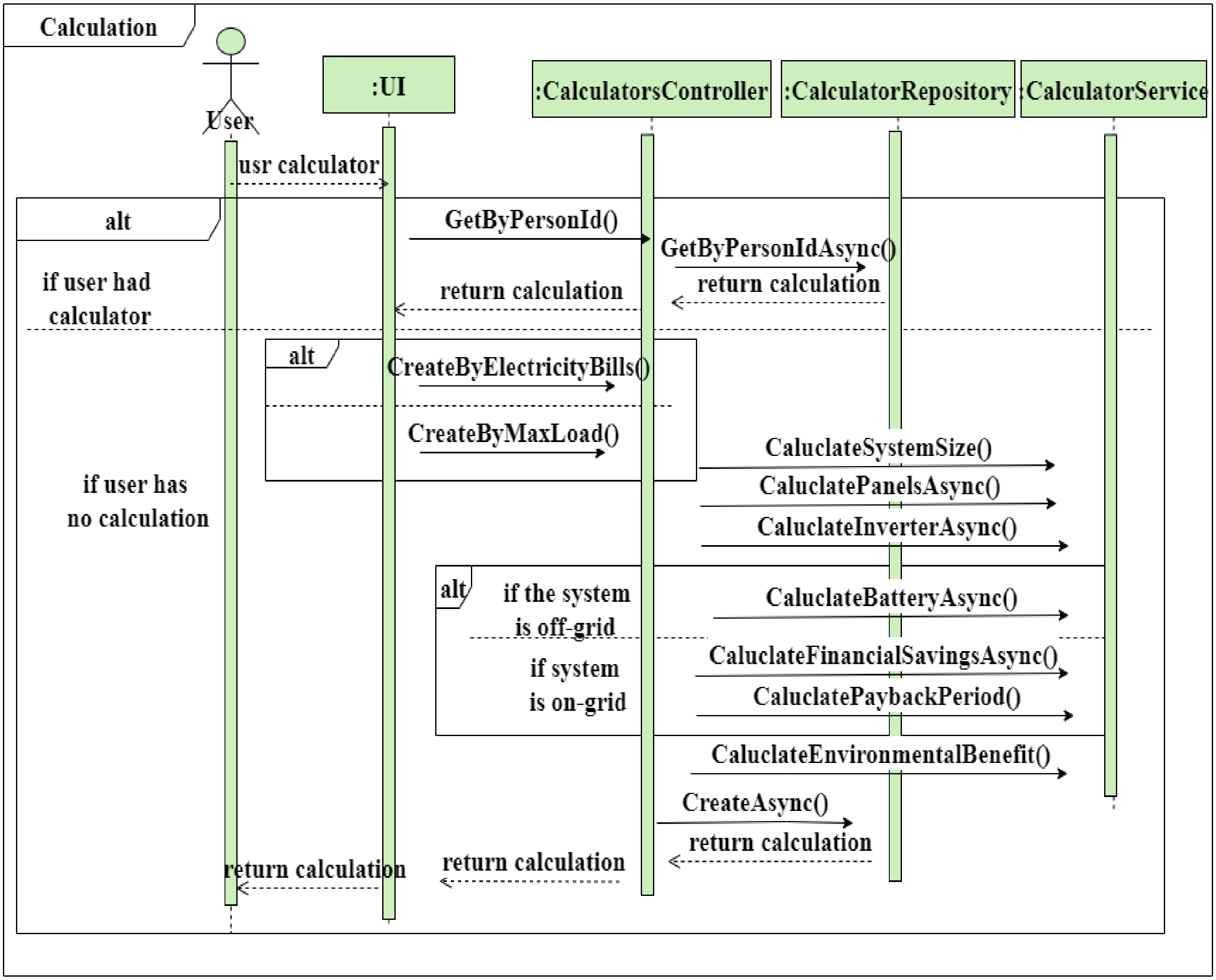


Figure 9 Calculation Sequence

4.3.4 Show Daily and Hourly Prediction Sequence

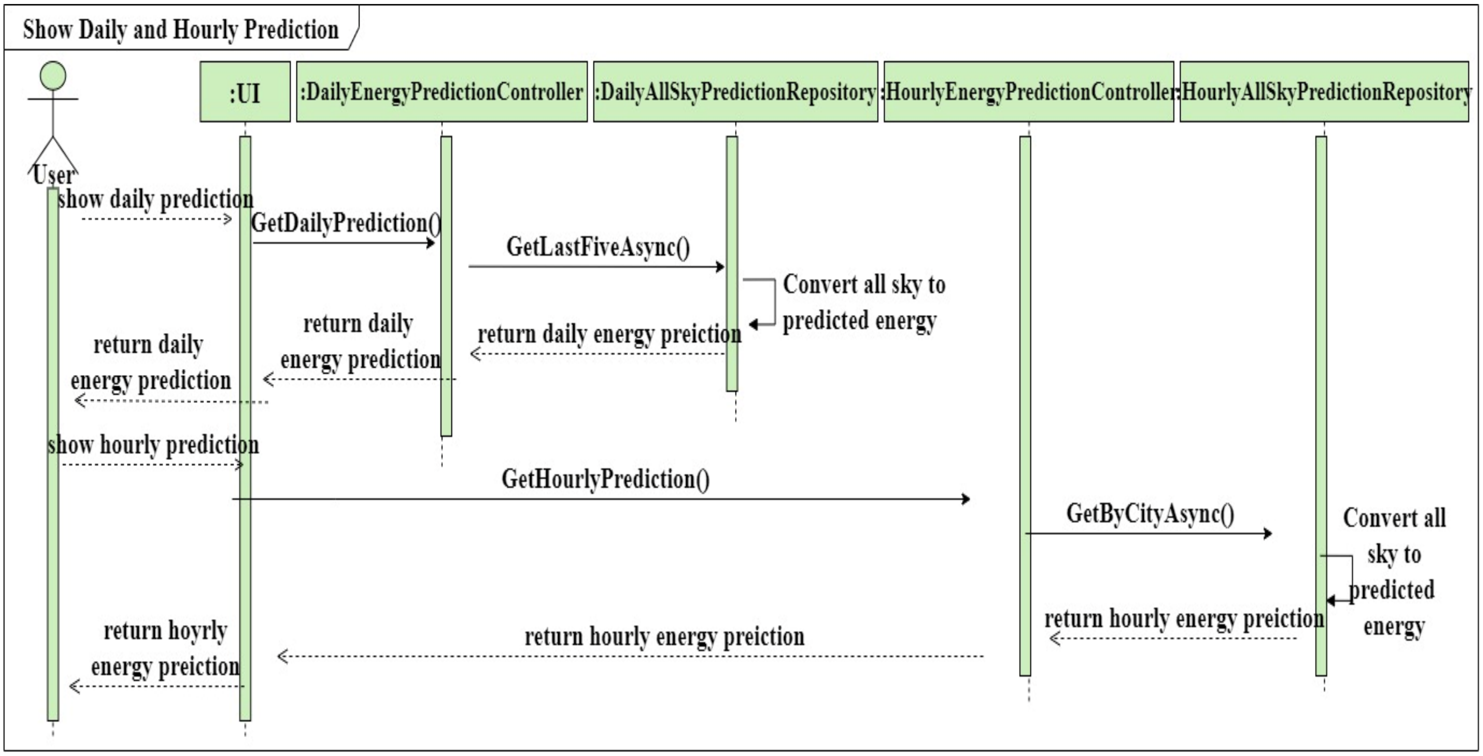


Figure 10 Show Daily and Hourly Prediction Sequence

4.3.5 Chatbot Sequence

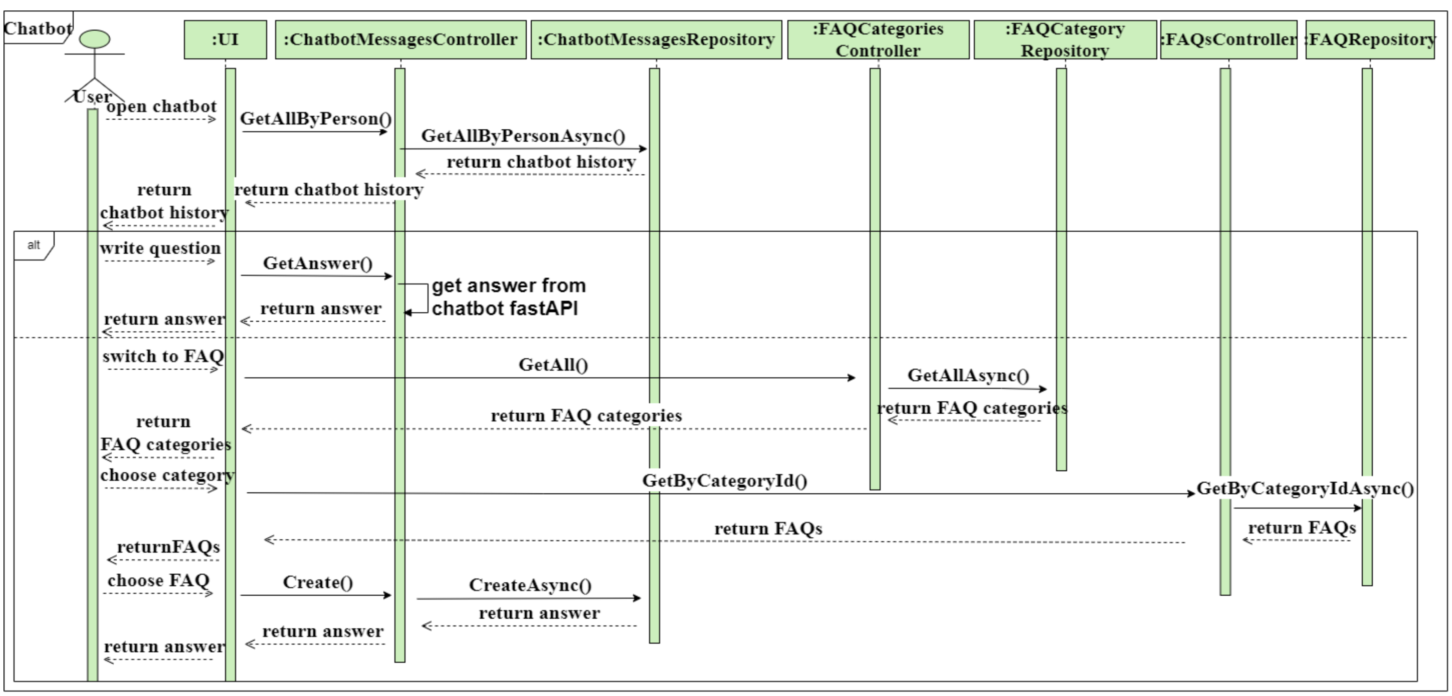


Figure 11 Chatbot Sequence

4.3.6 Show Products Sequence

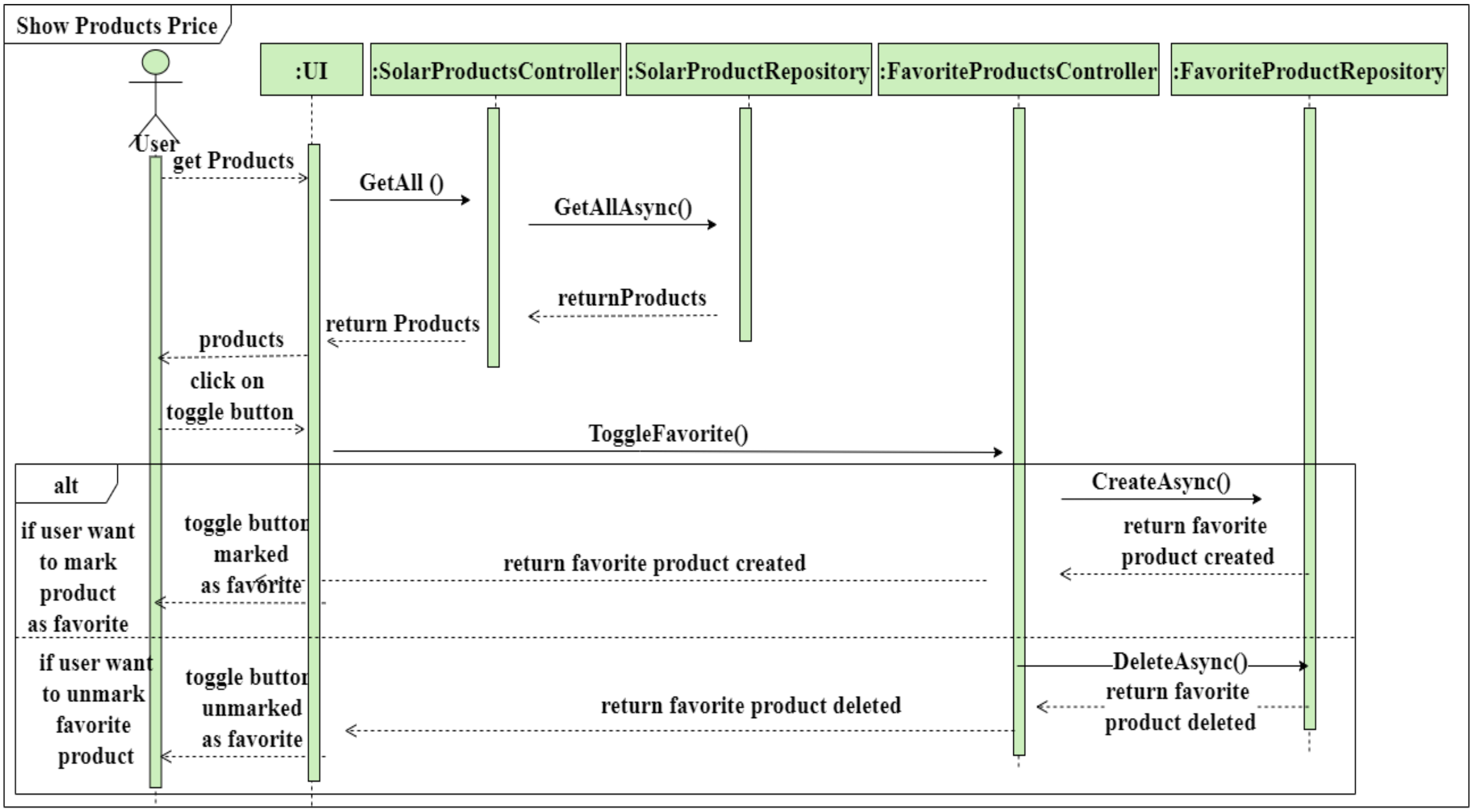


Figure 12 Show Products Sequence

4.3.7 Filter Posts Sequence

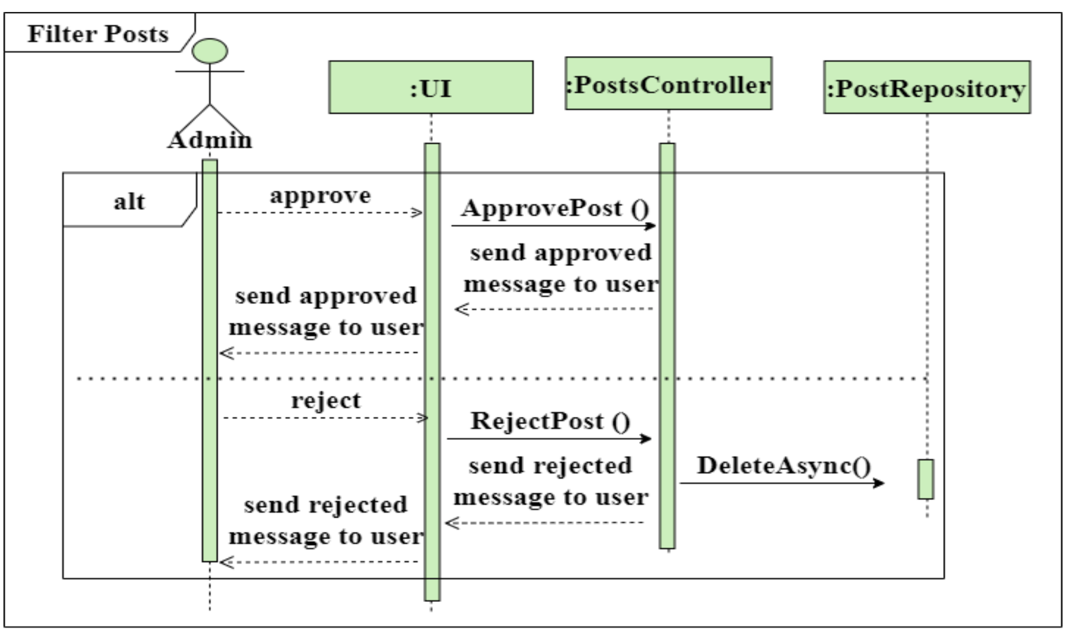


Figure 13 Filter Posts Sequence

4.3.8 Manage Posts Sequence

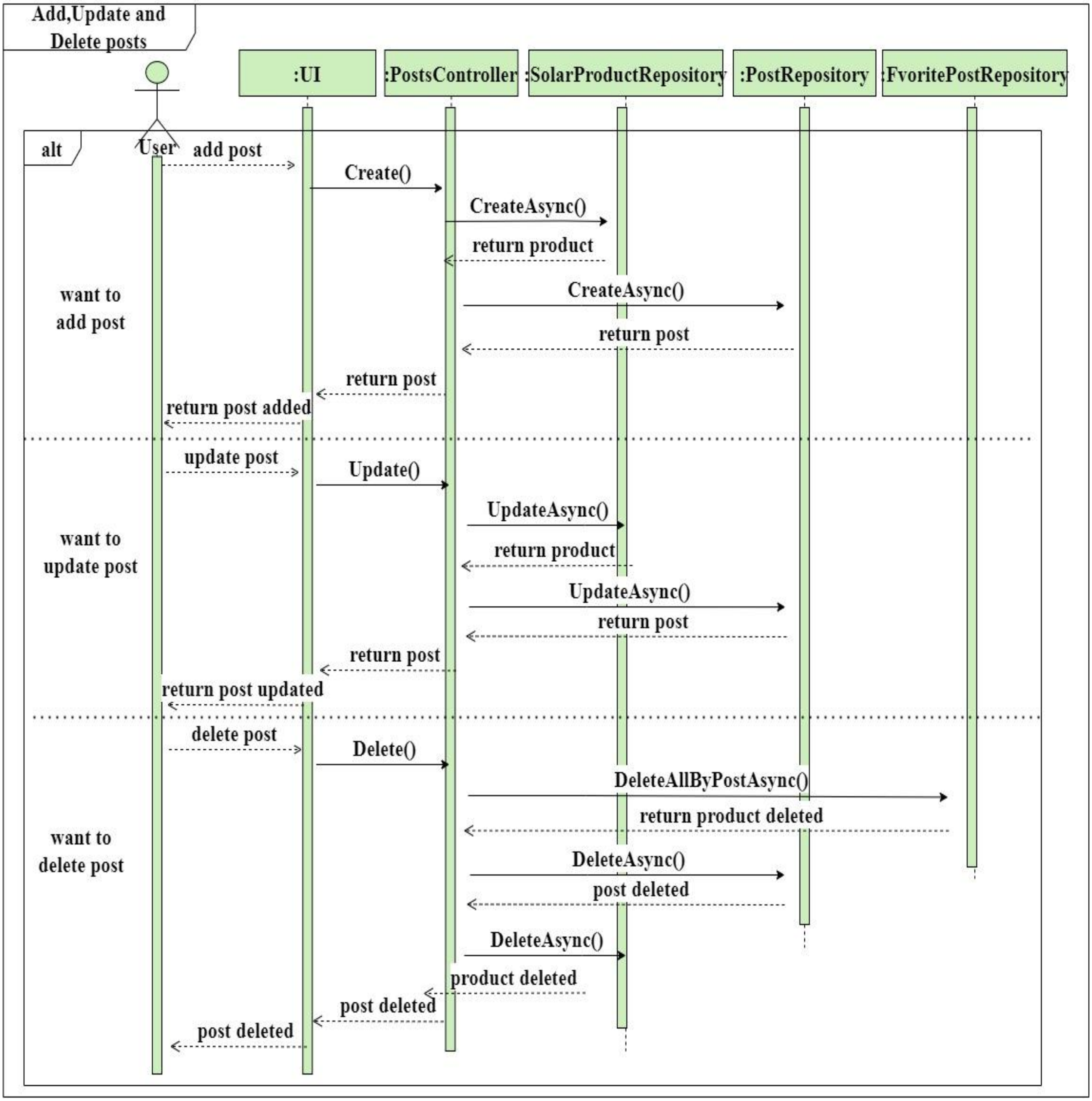


Figure 14 Manage Posts Sequence

4.3.9 Show Posts Sequence

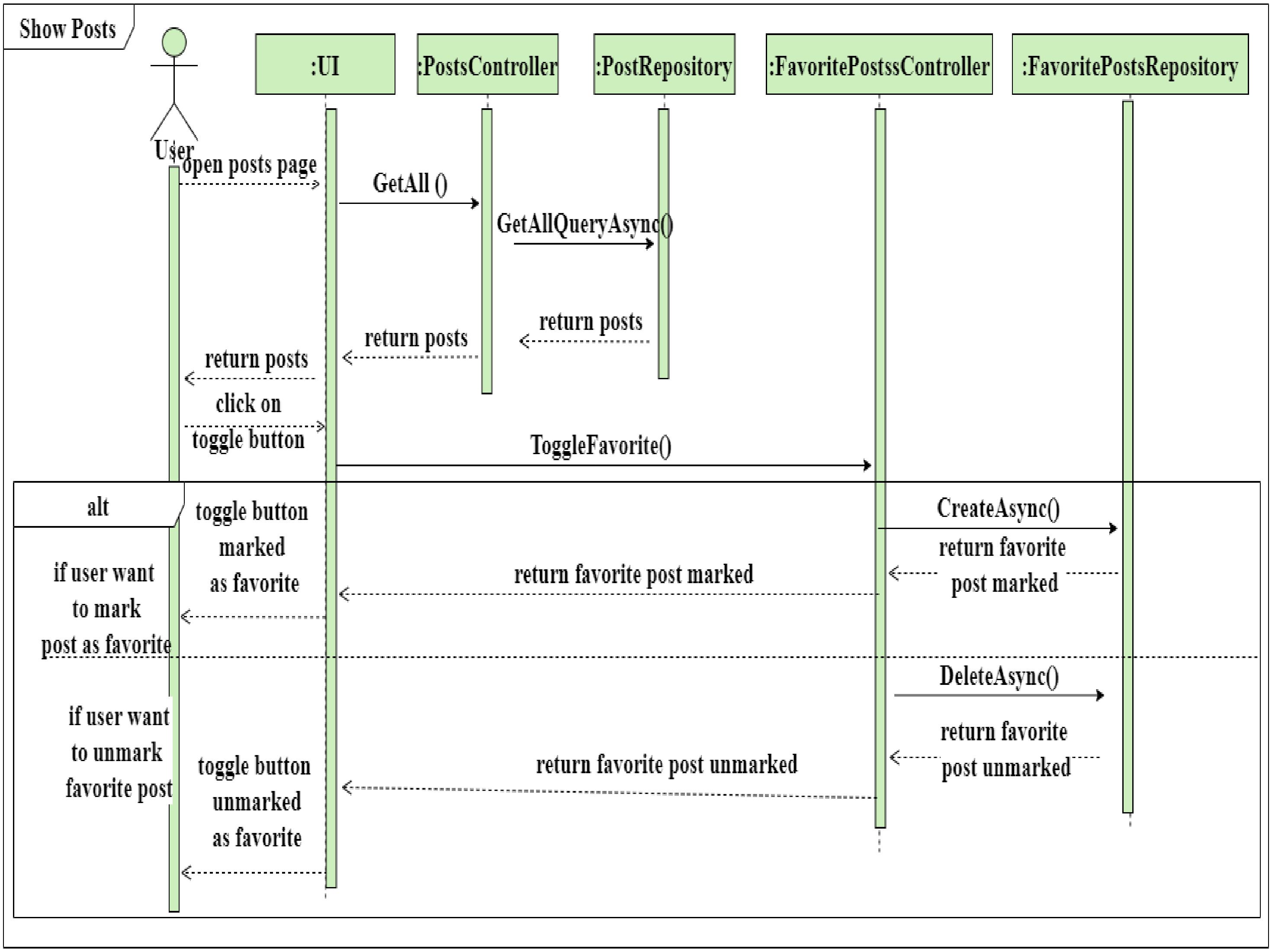


Figure 15 Show Posts Sequence

4.3.10 Find Solar Installer Sequence

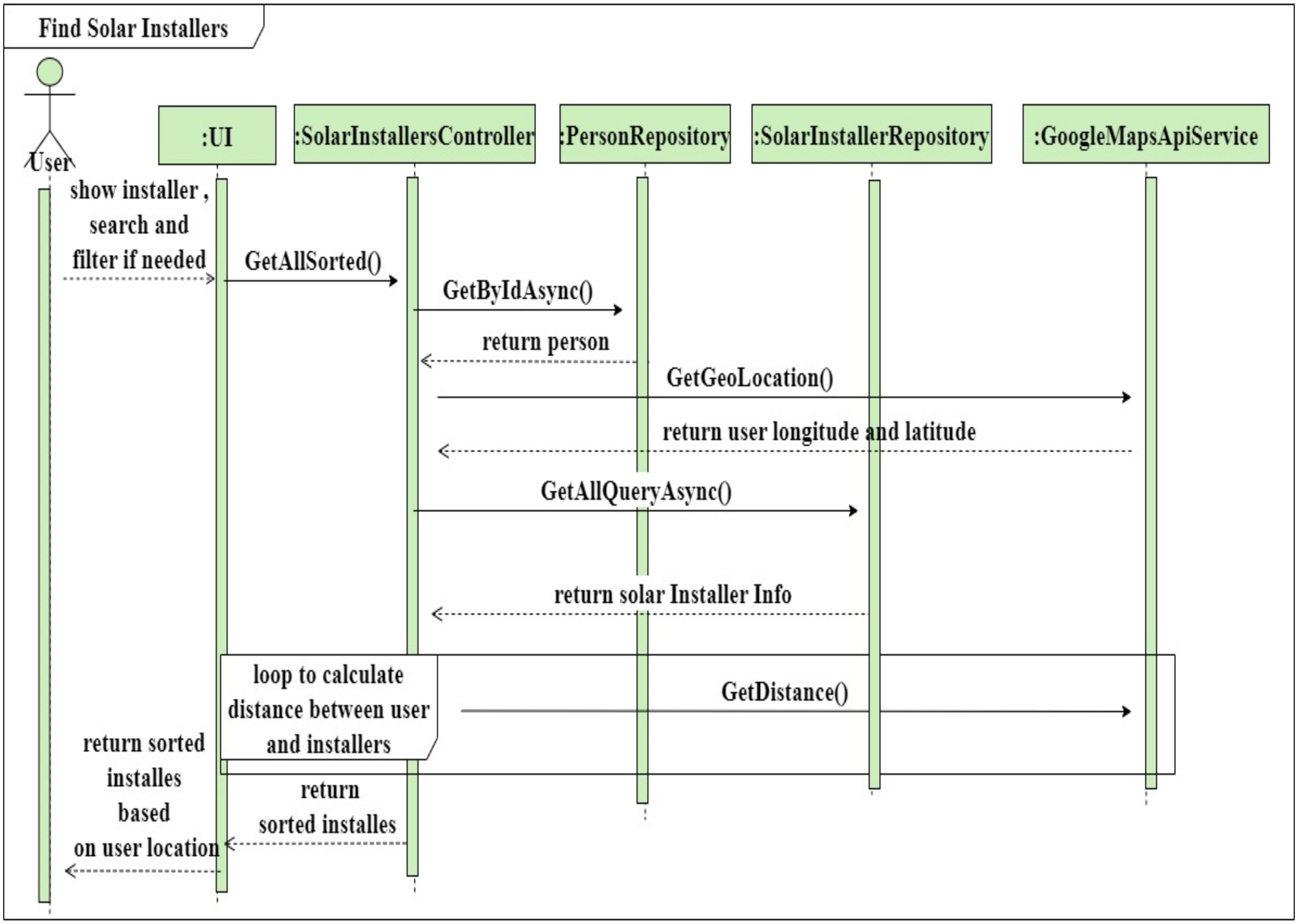


Figure 16 Find Solar Installer Sequence

4.4 Project ERD

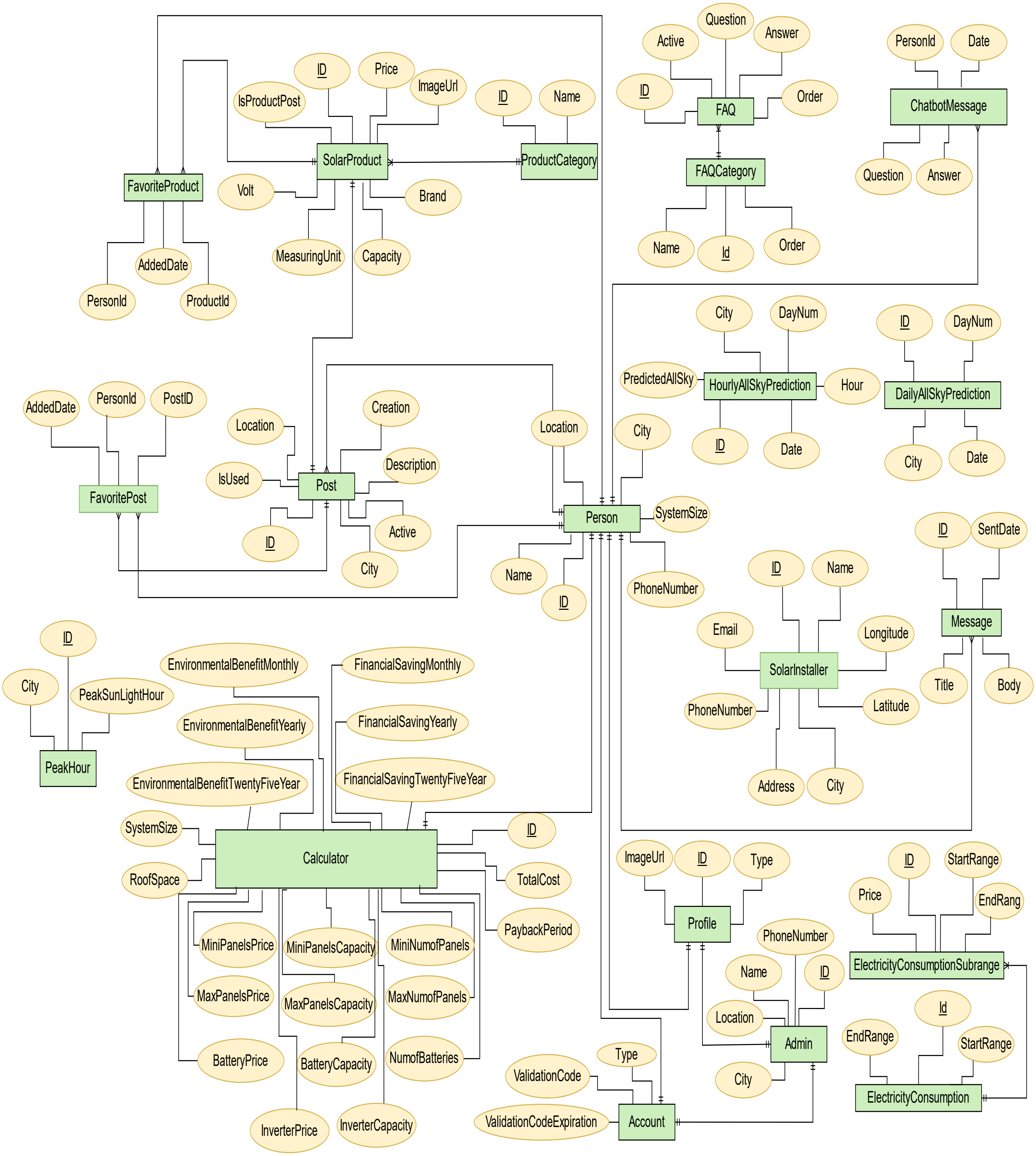
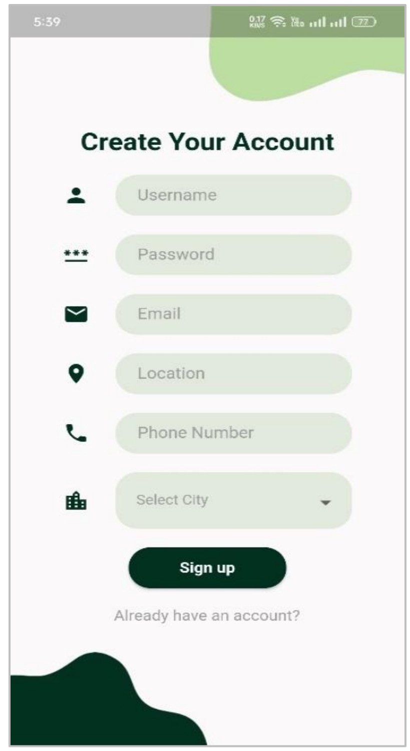


Figure 17 ERD

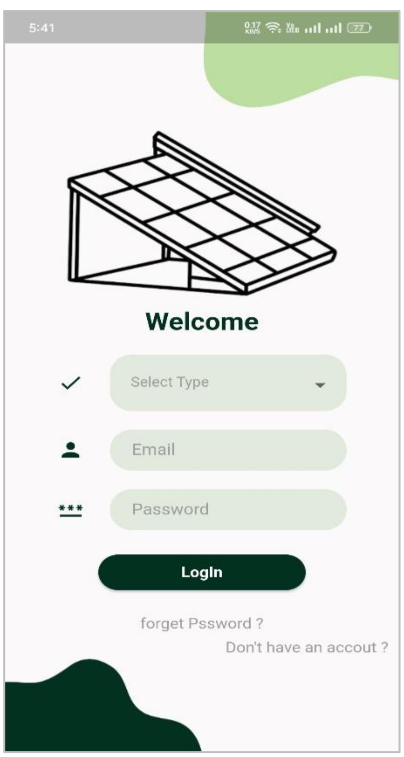
4.5 System GUI Design

4.5.1 Start Page

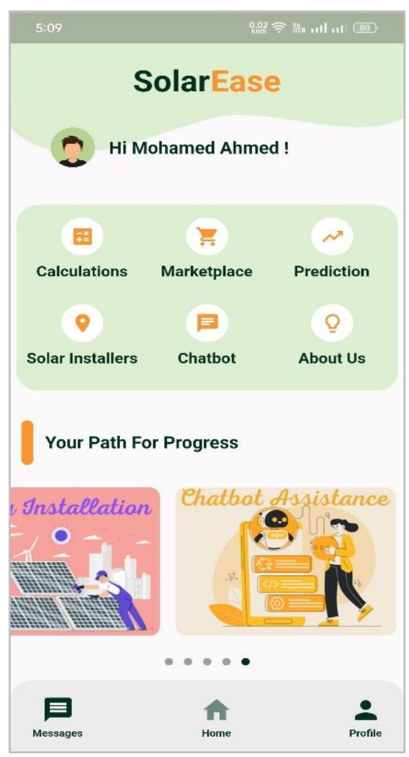
4.5.2 Sign up



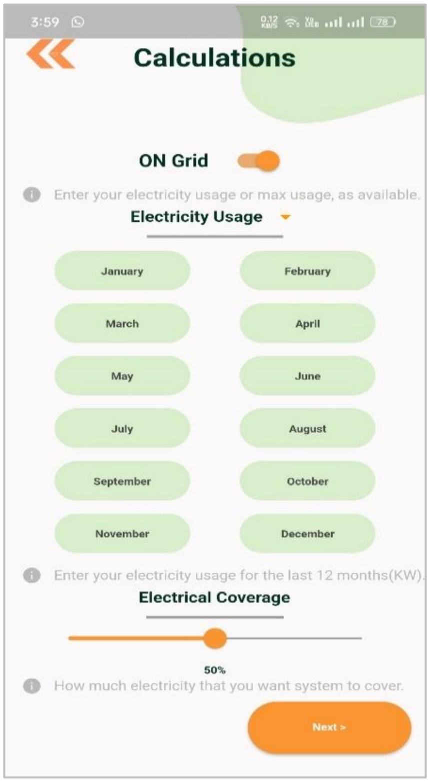
4.5.3 Login



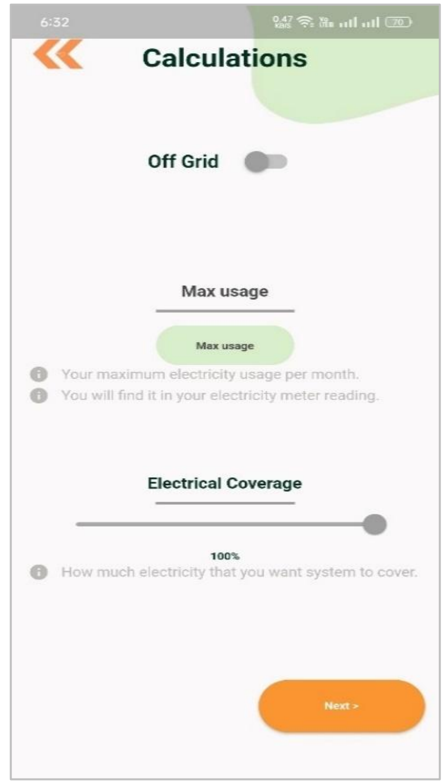
4.5.4 Home Page



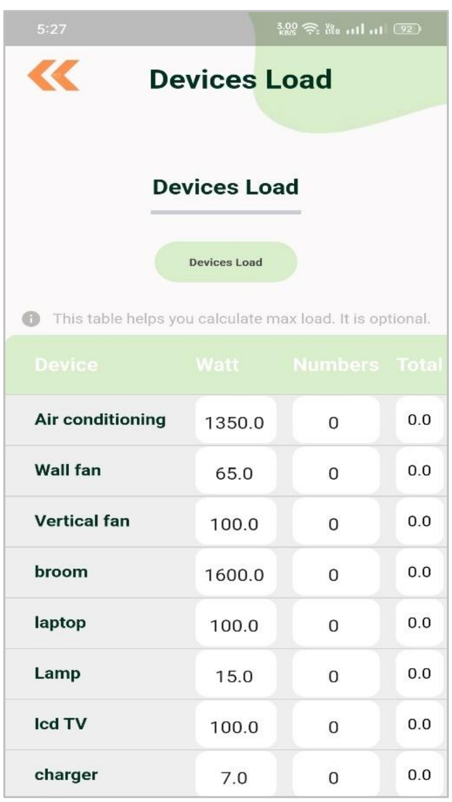
4.5.5 Electricity Usage



4.5.6 Max Usage

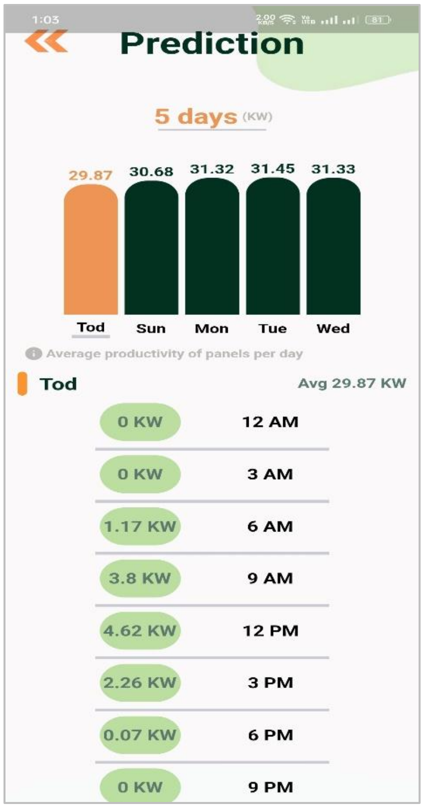


4.5.7 Devices load



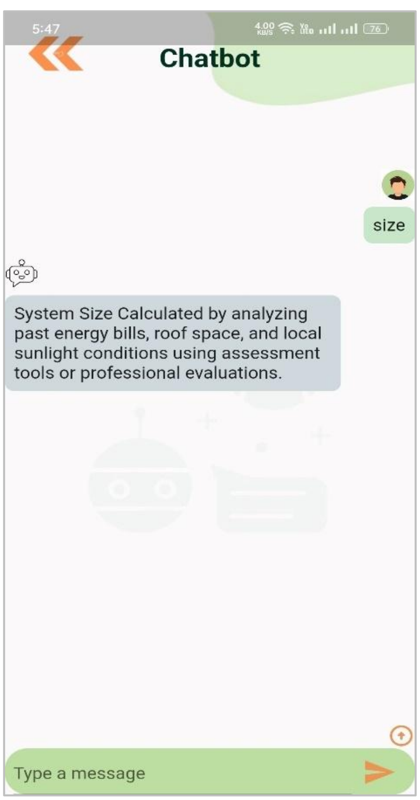
4.5.8 About Us

4.5.9 Prediction

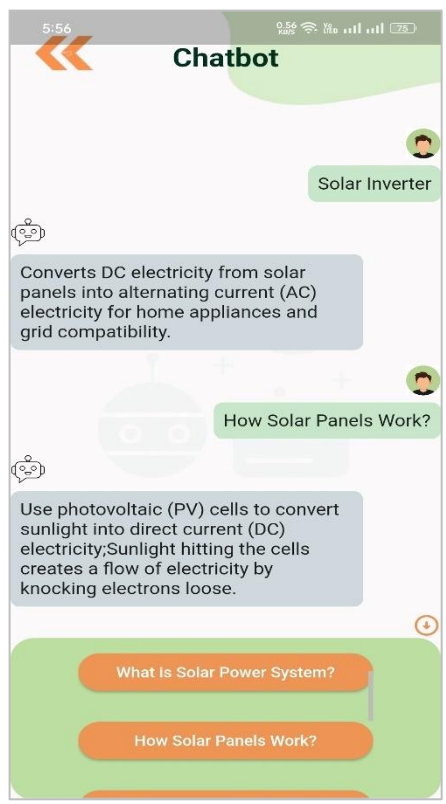


4.5.10 Solar Installers

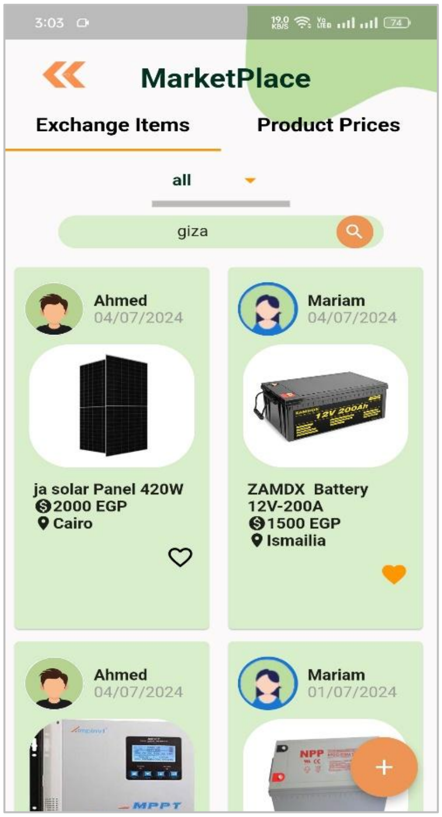
4.5.11 Interactive Chatbot



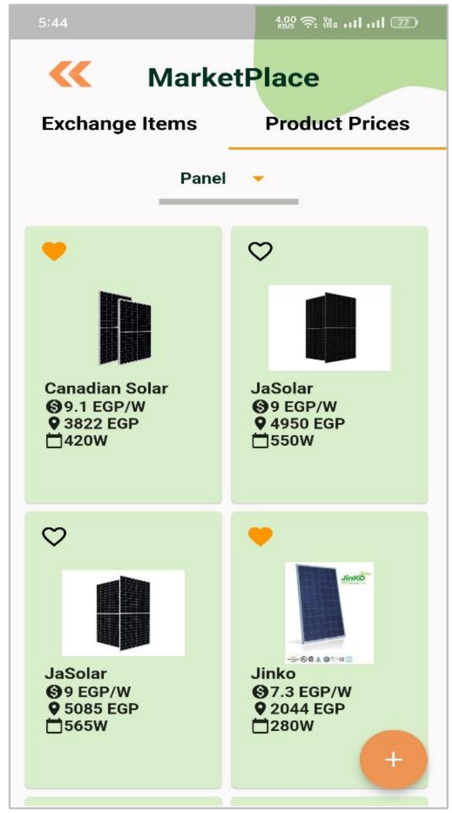
4.5.12 Category-Based Chatbot



4.5.13 Exchange Items



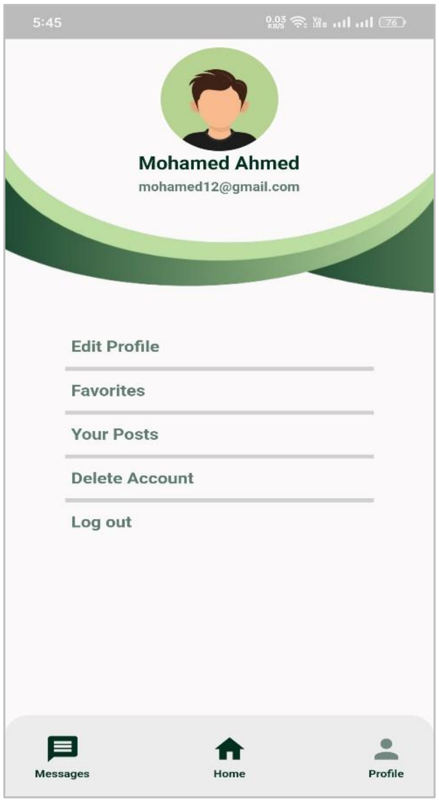
4.5.14 Product Prices



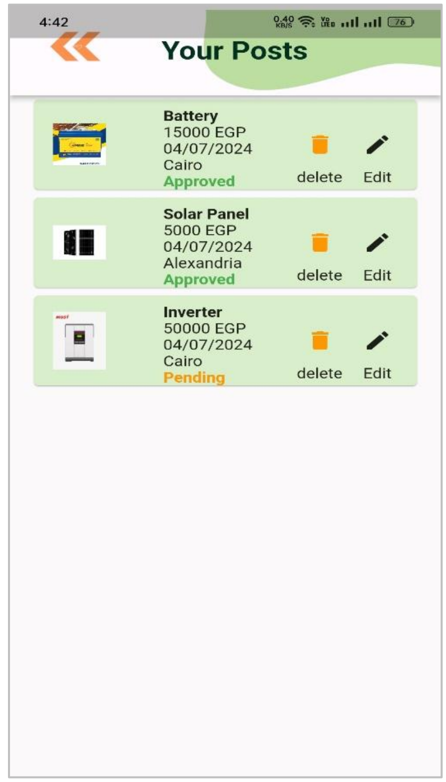
4.5.15 Detailed Post

4.5.16 Admin Filter Posts

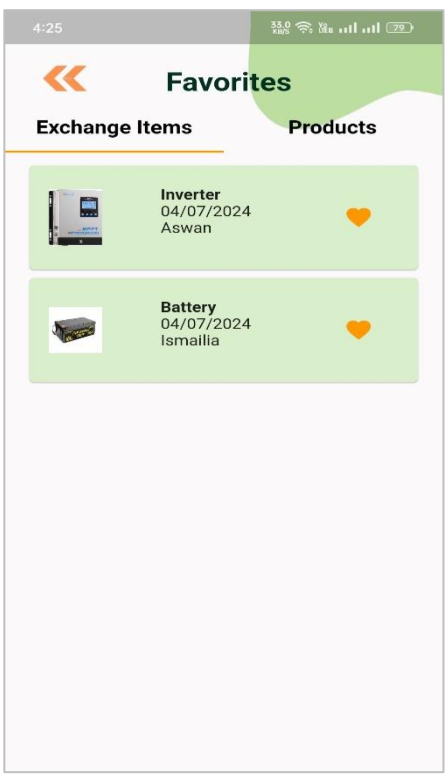
4.5.17 Profile



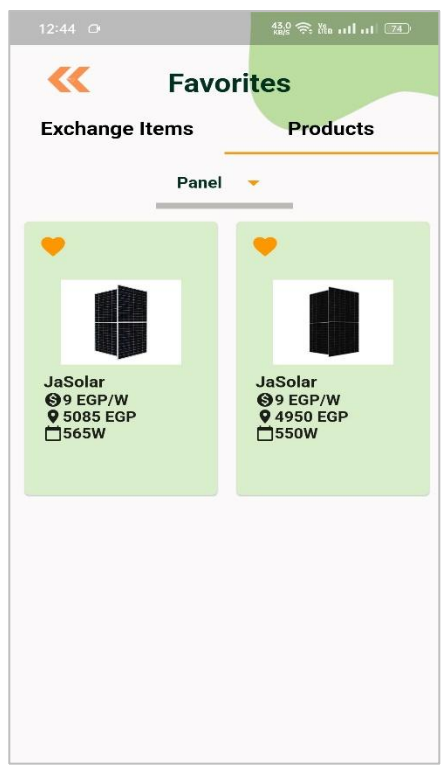
4.5.18 Your Posts



4.5.19 Favorites Exchange Items



4.5.20 Favorites Products



Chapter 5: Implementation and Testing

5.1 Implementation

5.1.1 Calculation

System Size and Roof Space Calculation

Firstly, calculating the required solar system size is fundamental. This calculation considers the user's daily average electricity consumption and the peak sunlight hours for their governorate. By using these inputs, we compute the optimal system size. Secondly, we calculate the required roof space to accommodate this system size. This calculation is crucial as it directly impacts the subsequent determinations of panels, inverters, and battery requirements.

Panels Calculation

The panels calculation is another significant aspect. Our calculator uses a comprehensive data set covering various solar panels, including their capacities and prices, to provide an estimation of the minimum and maximum number of panels needed and associated costs.

Inverter Calculation

In addition to panels, the We also determine the appropriate inverter capacity and cost. The inverter is a critical component of any solar system, converting the DC electricity generated by solar panels into AC electricity that can be used by household appliances. We evaluate the user's devices load to select an inverter that can handle the energy throughput efficiently. It also distinguishes between on-grid and off-grid systems, selecting the appropriate inverter type for each.

Battery Calculation

For users considering an off-grid solar system, the calculator calculates the necessary battery capacity and cost. Batteries are essential for storing excess energy during the day for use at during the absence of sunlight.

Financial Savings Calculation

Financial savings calculation is a key feature of the calculator, providing users with insights into the economic benefits of their solar investment. This includes computing the monthly, yearly, and 25-year savings by using electricity tariffs to determine the difference in cost before and after solar system installation, providing accurate financial savings estimations.

Payback Period Calculation

Additionally, the calculator determines the payback period by considering metrics such as the total system cost and the yearly financial savings achieved through reduced electricity bills, indicating how long it will take for the user to recover their investment in the solar system.

Environmental Benefits Calculation

Lastly, the calculator estimates the environmental benefits of installing a solar system by calculating the reduction in CO2 emissions monthly, yearly, and over 25 years.
Based on the fact that each kilowatt-hour (kWh) of solar energy results in an estimated reduction of 0.45-0.5 kg of CO2 emissions compared to conventional fossil fuel energy sources.

5.1.2 Solar Installers

Distance Calculation

We utilize both the user's location and each installer's coordinates to compute the distance between the user and every solar installer.
This critical step utilizes the Google Maps Distance Matrix API, which accurately measures distances in kilometers based on geographic coordinates.

Sorting Installers by Proximity

After calculating distances, we sort the list of solar installers. This sorting prioritizes the closest installers, ensuring they are prominently displayed to the user.

Filtering and Searching Installers

In addition to proximity sorting, users can filter solar installation companies by city and search for specific companies by name. This flexibility allows users to find and connect with certified solar installers that best meet their needs.

5.1.3 Prediction

In this project, we implemented two models for predicting solar irradiance: an hourly model and a daily model.

5.1.3.1 Daily Model

Introduction

Predicting daily solar irradiance is challenging due to weather variability, cyclic temporal features, and the necessity for precise meteorological data tailored to Egypt, impacting prediction reliability and accuracy.

We tested SVR and Feedforward Neural Networks (FNN) for their strong performance in handling complex, non-linear data relationships. SVR offers high accuracy with reduced overfitting, while FNNs capture intricate patterns effectively through their multi-layered structure.

Metrics / Model	SVR	FNN
R2	0.844	0.768
MAE	0.405	0.505

These metrics highlight SVR's superior performance in terms of both accuracy and reliability for our specific application in daily solar irradiance prediction.

In this project, we implemented a model to predict daily solar irradiance using Support Vector Regression (SVR). We evaluate our model using data from NASA POWER to ensure high-quality inputs and reliable predictions.

Literature Review

This section reviews previous research on daily solar irradiance prediction, comparing Support Vector Regression (SVR) with Artificial Neural Networks (ANN), Decision Tree (DT), Multilayer Perceptron (MLP), and other techniques. SVR shows superior performance.

- Muhammed A. Hassan et al. (2017) investigated tree-based ensemble methods for modeling daily solar radiation across five solar-meteorological stations in the MENA region. They compared SVR, MLP, and DT for daily solar radiation forecasting.

Metrics / Model	Paper SVR	Our SVR (Cairo)
R2	0.74	0.844

The table compares the R^2 metric for predicting daily solar radiation in Cairo between the SVR model from the research paper and our SVR model for Cairo.

- Meenal and Immanuel Selvakumar (2018) conducted a comparative study on predicting daily Global Solar Radiation (GSR) using SVM, ANN, and empirical models across eight locations in India utilizing parameters (month, latitude, longitude, bright sunshine hours, relative humidity, and temperature) data sourced from India Meteorolog Department. The study concluded that SVR outperformed ANN and empirical models due to its accuracy with smaller datasets, simplicity, and reliability. SVR effectively captures complex relationships within meteorological datasets, making it the best choice for solar irradiance prediction. The table compares the R^2 between the SVR model from the research paper and our SVR Cairo model.

Model / Metrics	Paper SVR	Our SVR (Cairo)
R2	0.911	0.844

Methodology

In this section, we describe our proposed approach to predict daily solar irradiance using Support Vector Regression (SVR) with various preprocessing techniques and transformations.

Data Collection

We gathered daily weather data from the NASA POWER project, which offers extensive datasets on solar and meteorological conditions including 3,652 records covering parameters like Temperature, Relative Humidity, Precipitation, Surface Pressure, and Wind Speed for all governorates of Egypt. The data spans from January 1, 2014, to December 31, 2023.

- **This is plot for visualizing solar irradiance patterns:**

The plot shows average solar irradiance (ALLSKY_SFC_SW_DWN) during the summer months (June to August) across different years, highlighting annual trends.

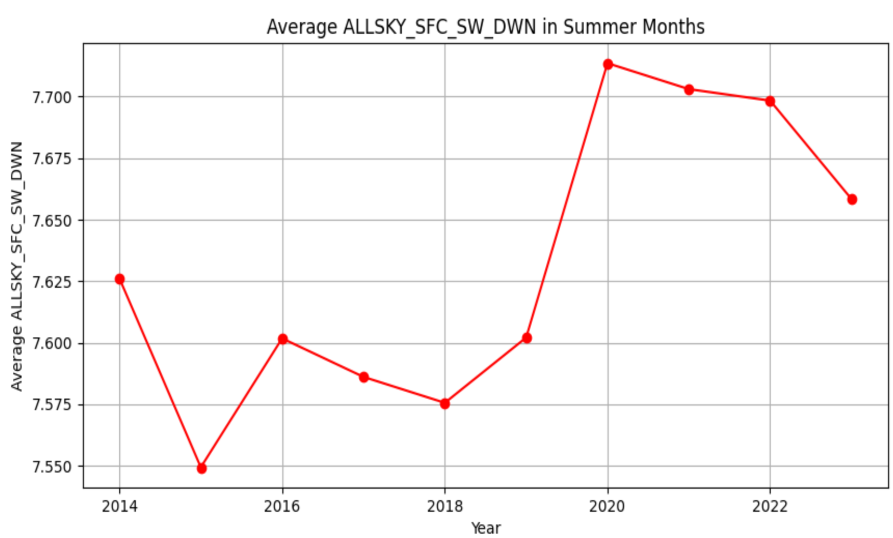


Figure 18 Line plot of average solar irradiance in summer in Cairo

Data Preprocessing

Data preprocessing focused on removing rows containing any value of -999, indicating missing data as per NASA POWER, ensuring reliability and accuracy.

Feature Engineering

Cyclical transformations were applied to time-related features (YEAR, MO, DY) by converting them into sine and cosine components to capture their periodic patterns effectively.

Feature Scaling

We normalized the meteorological features (T2M, RH2M, PRECTOTCORR, PS, WS10M) using MinMaxScaler to ensure each feature contributed equally to the model's learning process.

Model Architecture

Our approach involved implementing a Support Vector Regression (SVR) model to predict daily solar irradiance. Through extensive hyperparameter tuning using GridSearchCV, we optimized the models by selecting different parameter values for each model. Specifically, we experimented with various kernel functions (Radial Basis Function (rbf) and polynomial (poly)), regularization parameters (C), epsilon values, and gamma parameters.

This allowed us to tailor the SVR model to each dataset's specific characteristics, ensuring optimal performance and accurately capturing the complex relationships between meteorological parameters and solar irradiance for precise daily predictions.

Model Training and Hyperparameter Tuning

We employed Support Vector Regression (SVR) to forecast solar irradiance, leveraging GridSearchCV for thorough hyperparameter tuning which aimed to determine the optimal settings that enhance the predictive accuracy of our model.

Hyperparameter tuning involved evaluating diverse configurations including kernel types (rbf, poly, sigmoid), regularization parameters (C values of 0.1, 1, 10, 100), epsilon values (0.001, 0.01, 0.1, 1), gamma settings ('scale', 'auto', 0.001, 0.01, 0.1), polynomial degrees (2, 3, 4), and coef0 values (0.0, 0.1, 0.5, 1.0).

Model Evaluation

We evaluated the model using Mean Absolute Error (MAE) and R^2 score on the test set. These metrics helped us assess the performance and accuracy of our model.

Model Deployment

After finding the best SVR model via grid search, we saved the trained model and the preprocessing scalers using joblib. ensuring consistency in preprocessing steps for future use.

By following the previous systematic approach, we aimed to develop a robust and reliable model for predicting daily solar irradiance, addressing the challenges associated with variability in weather data and the need for accurate meteorological datasets.

Results and Discussion

In this section, we present the results of daily model evaluation for 27 governorates.

Accuracy

- **The evaluation accuracy of our daily models on the test sets for all 27 governorates:**
  - Alexandria: 86.9
  - Aswan: 88.9
  - Asyut: 82.9
  - Beheira: 85.2
  - Beni Suef: 85.4
  - Cairo: 86.5
  - Dakahlia: 92.1
  - Damietta: 87
  - Faiyum: 85.6
  - Gharbia: 86.4
  - Giza: 90.9
  - Ismailia: 92.5
  - Kafr el-Sheikh: 87.4
  - Luxor: 88.8
  - Marsa Matruh: 89.7
  - Minya: 82.4
  - Monufia: 86.4
  - New Valley: 89.5
  - North Sinai: 92.1
  - Port Said: 91.9
  - Qalyubia: 86.5
  - Qena: 91.5
  - Red Sea: 87.5
  - Sharqia: 86.4
  - Sohag: 90
  - South Sinai: 83.8
  - Suez: 91.4

These results show that our model performs handle the diverse meteorological conditions.

Testing

We conducted thorough testing of our model using NASA POWER datasets covering the period from January 1st to June 1st, 2024. These datasets are invaluable as they provide daily meteorological data crucial for our evaluation. Key variables such as surface shortwave downward irradiance (ALLSKY_SFC_SW_DWN), temperature at 2 meters (T2M), wind speed at 10 meters (WS10M), surface pressure (PS), corrected precipitation (PRECTOTCORR), and relative humidity at 2 meters (RH2M).

This comprehensive dataset enabled us to rigorously assess and validate our model's performance across a diverse range of meteorological conditions, extending beyond the constraints of our initial dataset.

- **Here are some examples of our finding:**

- **Location:** Asyut
- **Date:** 2024-01-31
- **Meteorological Data:**
    - T2M: 9.4, RH2M: 52.75, PRECTOTCORR: 5.27, PS: 100.4, WS10M: 3.74
- **Output** (Solar Irradiance)**:**
    - Predicted ALLSKY_SFC_SW_DWN: **4.05**
    - Actual ALLSKY_SFC_SW_DWN: **4.85**

- **Location:** Cairo
- **Date:** 2024-05-14
- **Meteorological Data:**
    - T2M: 22.1, RH2M: 47.94, PRECTOTCORR: 0.0, PS: 99.77, WS10M: 4.78
- **Output** (Solar Irradiance)**:**
    - Predicted ALLSKY_SFC_SW_DWN: **7.48**
    - Actual ALLSKY_SFC_SW_DWN: **7.28**

- **Location:** Luxor
- **Date:** 2024-06-01
- **Meteorological Data:**
    - T2M: 31.14, RH2M: 18.38, PRECTOTCORR: 0.0, PS: 98.34, WS10M: 5.89
- **Output** (Solar Irradiance)**:**
    - Predicted ALLSKY_SFC_SW_DWN: **8.09**
    - Actual ALLSKY_SFC_SW_DWN: **8.07**

- **Location:** Port Said
- **Date:** 2024-03-13
- **Meteorological Data:**
    - T2M: 18.85, RH2M: 68.25, PRECTOTCORR: 0.04, PS: 101.14, WS10M: 2.69
- **Output** (Solar Irradiance)**:**
    - Predicted ALLSKY_SFC_SW_DWN: **5.48**
    - Actual ALLSKY_SFC_SW_DWN: **5.42**

- **Here are the testing accuracies of our hourly SVR models on the test sets of the 27 governorates of Egypt are as follows:**

  - Alexandria: 78.1
  - Aswan: 72.6
  - Asyut: 81.8
  - Beheira: 83.9
  - Beni Suef: 70.1
  - Cairo: 84.4
  - Dakahlia: 83.8
  - Damietta: 80.9
  - Faiyum: 85.2
  - Gharbia: 84.8
  - Giza: 86.7
  - Ismailia: 87.9
  - Kafr el-Sheikh: 82
  - Luxor: 84.7
  - Marsa Matruh: 77.1
  - Minya: 83.6
  - Monufia: 84.1
  - New Valley: 66.5
  - North Sinai: 87.1
  - Port Said: 80.2
  - Qalyubia: 84.5
  - Qena: 87.6
  - Red Sea: 68.8
  - Sharqia: 85
  - Sohag: 84.7
  - South Sinai: 78
  - Suez: 87.5

Across the governorates of Egypt, daily SVR models exhibit diverse accuracies across 27 regions. Many governorates achieve accuracies above 80%, with notable performances in areas like Ismailia (87.9%), Qena (87.6%) and Giza (86.7%). Conversely, New Valley (66.5%) and Red Sea (68.8%) show lower accuracies below 70%. These findings highlight significant variability in predictive performance influenced by local conditions and the complexities of the models used.

API

Our FastAPI endpoints are designed to deliver daily predicted solar irradiance values for all governorates of Egypt. The process begins by fetching weather data from the OpenWeatherMap API every three hours. Daily forecasts involve additional preprocessing steps due to the need for summarizing daily weather conditions. Key summaries include average temperature, average humidity, maximum rain, average pressure, and maximum wind speed for each day. These daily summaries undergo cyclical transformations and scaling before being used by our models. The processed data are crucial inputs for the best SVR models saved during development, which leverage this information to predict accurate solar irradiance daily.

5.1.3.2 Hourly Model

Introduction

While daily predictions support long-term planning, hourly predictions are essential for optimizing energy capture efficiency and meeting real-time electricity demand.

In this project, we implemented a Support Vector Regression (SVR) model to predict hourly solar irradiance. By applying cyclical transformations to time-related features and using MinMaxScaler for normalization, our approach captures daily solar irradiance patterns. Evaluation with NASA POWER data ensures accurate hourly predictions of solar irradiance.

We chose SVR for daily predictions based on a comprehensive literature review, which highlighted its superior accuracy and stability. The success of SVR in daily predictions supported our decision to use SVR for hourly solar irradiance prediction as well. Our hourly model builds upon these findings to deliver precise and reliable solar irradiance forecasts.

Literature Review

This section reviews previous research on hourly solar irradiance prediction, comparing Support Vector Regression (SVR) with Artificial Neural Networks (ANN), Decision Trees (DT), Multilayer Perceptron (MLP). Highlighting the superior performance of SVR in domain.

- Muhammed A. Hassan et al. (2017) investigated tree-based ensemble methods for modeling hourly solar radiation across five solar-meteorological stations in the MENA region. They compared SVR, MLP, and DT for hourly solar radiation forecasting.

Model / Metrics	Paper SVR	Our SVR (Cairo)
R2	0.90	0. 949

The table compares the R^2 metric for predicting hourly solar radiation in Cairo between the SVR model from the research paper and our SVR model.

- Hui He, Nanyan Lu, Yongjun Jie, Bo Chen, and Runhai Jiao (2020) studied solar irradiance forecasting using SVR and other machine learning models on hourly data from 2006 to 2015 from MIDC. They partitioned data into training (2006–2014) and testing (2015), including variables like month, day, hour, dew point temperature, relative humidity, cloud cover, and wind speed. SVR performed robustly, competing with Random Forest and ELM, with LSTM ranking highest overall. SVR-RBF's accuracy highlights its reliability for solar irradiance prediction.

- Basaran, Özçift, and Kılınç (2019) investigated solar irradiance prediction in five Turkish cities using SVR, ANN, and DT models with 48,060 hourly observations spanning 2012 to 2016. Data were sourced from the Turkish State Meteorological Service and refined for quality. SVR consistently outperformed ANN and DT, achieving 79.20% accuracy compared to ANN's 75.13% and DT's 69.05%. The study highlighted SVR's efficacy in solar irradiance prediction, credited to rigorous data preprocessing, advanced feature engineering, and superior hyperparameter tuning.

Model / Metrics	Paper SVR	Our SVR (Cairo)
R2	0.792	0.949

Methodology

In this section, we outline our approach to predict hourly solar irradiance using Support Vector Regression (SVR) with various preprocessing techniques and transformations.

Data Collection

We collected hourly meteorological data from the NASA POWER datasets, covering all governorates of Egypt from January 1, 2014, to December 31, 2023. These datasets include 87,648 records detailing parameters like Temperature, Relative Humidity and Wind Speed.

➢ **This is plot for visualizing solar irradiance patterns:**

The plot is a heatmap showing average solar irradiance (ALLSKY_SFC_SW_DWN) by hour and year, providing a year-round overview.

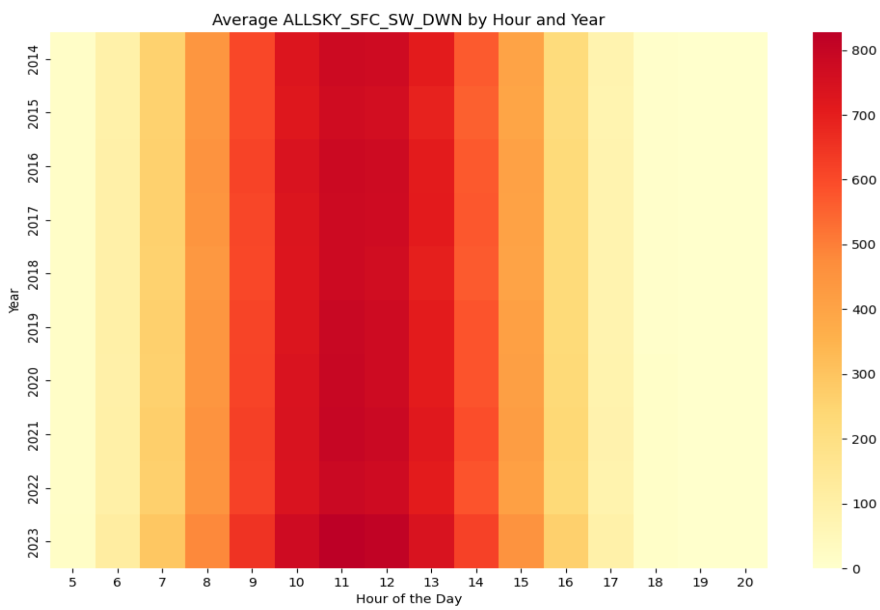


Figure 19 Heatmap of average solar irradiance per hour throughout the year in Cairo

Data Preprocessing

We began by removing rows containing any value of -999, indicating missing data as per NASA POWER, reducing the dataset to 86,184 rows. We then filtered the dataset to include only daytime hours between 5 AM and 8 PM, further refining the records to 57,456 rows, ensuring reliability.

Feature Engineering

We applied cyclical transformations to time-related features (YEAR, MO, DY, HR) by converting them into sine and cosine components, which improved the model's ability to capture periodic characteristics. This approach outperformed StandardScaler transformations, demonstrating its importance in enhancing the model's understanding of temporal patterns.

Feature Scaling

Meteorological features (T2M, RH2M, PRECTOTCORR, PS, WS10M) were normalized using StandardScaler to ensure each feature contributed equally to the model's learning process.

Model Architecture

We implemented a Support Vector Regression (SVR) model with a polynomial kernel ('poly') to predict hourly solar irradiance (ALLSKY_SFC_SW_DWN). Hyperparameter tuning using GridSearchCV optimized the model with parameters including C=100, epsilon=1, gamma='scale', degree=2, and coef0=0.0. The model's performance was enhanced by preprocessing steps that included cyclical transformations of time-related features and standardization of meteorological variables, ensuring accuracy in predicting solar irradiance.

Model Training and Hyperparameter Tuning

We used Support Vector Regression for solar irradiance prediction, employing GridSearchCV for comprehensive hyperparameter tuning to optimize model accuracy. Hyperparameter configurations evaluated included kernel types (rbf, poly, sigmoid), regularization parameters (C values of 0.1, 1, 10, 100), epsilon values (0.001, 0.01, 0.1, 1), gamma settings ('scale', 'auto', 0.001, 0.01, 0.1), polynomial degrees (2, 3, 4), and coef0 values (0.0, 0.1, 0.5, 1.0).

Model Evaluation

The SVR model's performance was evaluated on the test set using Mean Absolute Error (MAE) and R^2 score metrics to assess its accuracy in predicting solar radiation.

Model Deployment

After finding the best SVR model via grid search, we saved the trained model, StandardScaler and cyclical transformer, using joblib. ensuring consistency in preprocessing steps.

By following the previous structured methodology, we aimed to develop a robust predictive model for predicting hourly solar irradiance, leveraging preprocessing techniques and SVR optimization to handle the complexities of meteorological data effectively.

Results and Discussion

In this section, we present and discuss the results of hourly models evaluation for 27 governorates.

Accuracy

- **The evaluation accuracy of hourly models on the test sets for all 27 governorates of Egypt are as follows:**

- Alexandria: 95.3
- Aswan: 97.8
- Asyut: 95.8
- Beheira: 96.6
- Beni Suef: 96.8
- Cairo: 96.5
- Dakahlia: 96.1
- Damietta: 95.6
- Faiyum: 96.9
- Gharbia: 96.6
- Giza: 96.4
- Ismailia: 96.8
- Kafr el-Sheikh: 95.9
- Luxor: 97.7
- Marsa Matruh: 94.6
- Minya: 97.2
- Monufia: 96.5
- New Valley: 97.7
- North Sinai: 96.7
- Port Said: 95.4
- Qalyubia: 96.4
- Qena: 97.6
- Red Sea: 97.4
- Sharqia: 96.5
- Sohag: 97.7
- South Sinai: 97.1
- Suez: 96.9

These results show that our model performs well across all 27 governorates of Egypt and can effectively handle the diverse meteorological conditions and patterns found throughout the country.

Testing

We tested our model using NASA power datasets spanning from January 1st to June 1st, 2024. After preprocessing, which involved removing ALLSKY_SFC_SW_DWN values of -999, the dataset covered the period from January 1st to March 1st, 2024.

These datasets provide hourly meteorological data, including surface shortwave downward irradiance (ALLSKY_SFC_SW_DWN), temperature (T2M), wind speed at 10 meters (WS10M), pressure (PS), corrected precipitation (PRECTOTCORR), and relative humidity at 2 meters (RH2M). This comprehensive data allowed us to assess and validate our model's performance across a diverse range of meteorological conditions beyond our initial dataset.

➢ **Here are some examples of our findings:**

- **Location:** Asyut
- **Date and Time:** 2024-01-31 15:00
- **Meteorological Data:**
  - T2M: 15.99, RH2M: 29.88, PRECTOTCORR: 0.11, PS: 100.21, WS10M: 6.15
- **Output** (Solar Irradiance)**:**
  - Predicted ALLSKY_SFC_SW_DWN: **301.62**
  - Actual ALLSKY_SFC_SW_DWN: **328.79**

- **Location:** Cairo
- **Date and Time:** 2024-02-17 12:00
- **Meteorological Data:**
  - T2M: 18.68, RH2M: 46.44, PRECTOTCORR: 0.01, PS: 100.58, WS10M: 5.78
- **Output** (Solar Irradiance)**:**
  - Predicted ALLSKY_SFC_SW_DWN: **645.48**
  - Actual ALLSKY_SFC_SW_DWN: **668.74**

- **Location:** Luxor
- **Date and Time:** 2024-01-09 08:00
- **Meteorological Data:**
  - T2M: 9.6, RH2M: 35.5, PRECTOTCORR: 0, PS: 98.52, WS10M: 4.09
- **Output** (Solar Irradiance)**:**
  - Predicted ALLSKY_SFC_SW_DWN: **279.98**
  - Actual ALLSKY_SFC_SW_DWN: **270.2**

- **Location:** Port Said
- **Date and Time:** 2024-03-01 10:00
- **Meteorological Data:**
  - T2M: 26.05, RH2M: 23.94, PRECTOTCORR: 0.0, PS: 101.08, WS10M: 3.6
- **Output** (Solar Irradiance)**:**
  - Predicted ALLSKY_SFC_SW_DWN: **671.37**
  - Actual ALLSKY_SFC_SW_DWN: **662.14**

- **Here are the testing accuracies of our hourly SVR models on the test sets of the 27 governorates of Egypt are as follows:**
  - Alexandria: 93.8
  - Aswan: 97.3
  - Asyut: 93.1
  - Beheira: 94.9
  - Beni Suef: 84.3
  - Cairo: 94.9
  - Dakahlia: 94.2
  - Damietta: 91.9
  - Faiyum: 95.7
  - Gharbia: 95.9
  - Giza: 94.7
  - Ismailia: 95.9
  - Kafr el-Sheikh: 91.5
  - Luxor: 94.5
  - Marsa Matruh: 91.3
  - Minya: 96.1
  - Monufia: 94.4
  - New Valley: 75.5
  - North Sinai: 77.9
  - Port Said: 92.8
  - Qalyubia: 94.6
  - Qena: 97.1
  - Red Sea: 97.5
  - Sharqia: 95.5
  - Sohag: 90.6
  - South Sinai: 94.5
  - Suez: 96.2

Across the governorates of Egypt, the hourly SVR models show varied accuracies across the 27 regions. The majority of governorates achieve high accuracies exceeding 90%, with notable performances in regions such as Aswan (97.3%), Qena (97.1%), and Red Sea (97.5%). However, New Valley (75.5%) and North Sinai (77.9%) demonstrate lower accuracies below 80%. These results highlight significant variability in predictive performance influenced by local conditions and the complexities of the models utilized.

API

Our FastAPI endpoints are designed to deliver hourly predicted solar irradiance values for all governorates of Egypt. The process begins by fetching weather data from the OpenWeatherMap API every three hours. For hourly forecasts no preprocessing of the weather data summaries is required, the weather data are directly fed into our models. These models leverage cyclical transformations for date-time features and scaling using a pre-loaded scaler to predict accurate solar irradiance values on an hourly basis throughout the day. The data are crucial inputs for the best SVR models saved during development, which use the processed data to predict accurate solar irradiance every three hours across five days.

5.1.4 Chatbot

The solar energy chatbot has two interactive sections: text input queries and predefined category selection. By offering these two distinct sections, our solar energy chatbot ensures that users can efficiently access relevant information tailored to their preferences.

Interactive Chatbot

Description

In the text input section, users type their questions directly into a text field. The chatbot sends these questions to a FastAPI service, which processes them using NLTK's text processing techniques like tokenization, stemming, and Jaccard similarity. This mechanism finds the best matching response from predefined data segments to answer inquiries about solar.

Purpose

To provide users with personalized answers to specific queries about solar topics in real-time.

Literature Review

- Benjamin Chanakot and Charun Sanrach (February 2024) developed a Thai-language chatbot for analyzing mosquito-borne diseases. The chatbot uses Jaccard similarity to match user-provided symptoms with a database of disease symptoms, aiding in accurate disease identification. Implemented via the Line Messaging API with PHP and MySQL, the chatbot ensures effective user interaction by robustly handling variations in symptom descriptions. Jaccard similarity enhances its performance, making it a valuable tool for disease analysis.
- Suphakit Niwattanakul et al. (2013) introduced a method leveraging the Jaccard similarity coefficient to optimize information retrieval through search engines. Their research focused on enhancing search result accuracy and efficiency by measuring similarity between user-provided keywords and index terms. Implemented in Prolog programming, their approach demonstrated the effectiveness of the Jaccard method in matching search queries with relevant documents. Despite challenges with over-typed words, the study highlighted Jaccard's advantages over alternative similarity measurement techniques.

Category-Based Question Chatbot

Description

Users can select predefined categories related to solar energy, view related questions within each category, choose a specific question, and receive a corresponding answer generated from predefined responses by the chatbot.

Purpose

To guide users through structured information about solar energy.

5.1.5 Marketplace

5.1.5.1 Exchanged Items

In the Exchanged Items section of the Marketplace, users can buy and sell solar products to each other.

Browsing Posts

Users can browse through summarized posts of solar products. Moreover, they can access each post for more detailed information about specific products, enabling detailed evaluation. Users can filter posts based on product categories or search by city.

Additionally, users can mark posts as favorites for easy reference and comparison in the favorite page.

Adding New Posts

Users can add new post for selling solar product. Once post submitted, it is pending until admin approval to ensure quality control and adherence to marketplace standards.

Managing User Posts

Users can view their own solar product posts, including approved and pending ones, sorted from oldest to newest.

For existing posts, users can update them to modify product details. These updates undergo thorough validation by the admin to guarantee accuracy and consistency of data. Furthermore, users can remove their posts from the marketplace and associated favorites related to the post.

5.1.5.2 Product Prices

Solar Products Display and Interaction

The Solar Products component displays a variety of solar market products, featuring their latest prices, brands, and capacities. Products are categorized and presented in a user-friendly manner, enhancing navigation and understanding for users.

Additionally, the Solar Product component supports user interaction through product favoriting, enhancing engagement by allowing users to save and reference items for future use. Moreover, the Solar Product component enables users to filter products based on their category preferences.

5.1.6 Challenges and Solutions

For Both Hourly and Daily Models

- **Data Availability and Quality:**

Challenge: One of the significant challenges is the availability and quality of historical weather data. Incomplete or inaccurate data can lead to unreliable predictions. Additionally, the meteorological parameters across different areas of the same government are the same in most of the meteorological datasets, which limits the model's ability to make localized predictions.

Solution: We sourced our data from NASA POWER, which provides high-quality solar and meteorological datasets derived from NASA research to support renewable energy. This approach ensures access to reliable and datasets while acknowledging that it restricts us to predicting solar irradiance for entire governments in Egypt rather than specific localized areas.

- **Handling Temporal Features:**

Challenge: Effectively capturing the cyclical nature of time-related features (e.g., year, month, day) is crucial for accurate predictions.

Solution: Initially, we experimented with standard scalar transformations on temporal features. However, we found that applying cyclical transformations, converting these features into sine and cosine components, better preserved their periodic characteristics. This approach improved the model's ability to understand and predict temporal patterns.

For Hourly Model

- **Handling Nighttime Data for Solar Irradiance Predictions:**

Challenge: Hourly datasets include time intervals that extend beyond daylight hours, predicting solar irradiance during night hours, where solar irradiance is absent, poses a challenge as it introduces noise and potential biases into the model resulting in inaccurate predictions.

Solution: To address this challenge, we adopted a strategy inspired by research on solar irradiance forecasting. In the preprocessing step, we filtered out nighttime hours from our dataset by restricting our data to time intervals between 5 a.m. and 8 p.m., corresponding to daylight hours. This approach ensures that only relevant data, where solar irradiance is present, is used to train the model, thereby enhancing the performance and accuracy of solar irradiance predictions.

For OpenWeather API

- **Lack of Hourly Meteorological Data from OpenWeather API:**

Challenge: The OpenWeather API only provides meteorological data every 3 hours and not on an hourly basis, which restricts our ability to predict weather conditions on an hourly basis.

Solution: Even though the model is trained on hourly predictions, due to the limitations of the OpenWeather API, we will only provide users with predictions at 3-hour intervals: 0 AM, 3 AM, 6 AM, 9 AM, 12 PM, 3 PM, 6 PM, and 9 PM. This ensures that we align with the available data frequency while still delivering valuable and timely predictions to our users.

- **Lack of Daily Meteorological Data from OpenWeather API:**

Challenge: The OpenWeather API provides meteorological data only at 3-hour intervals, which limits our ability to directly predict daily weather conditions using their native data format.

Solution: To overcome this limitation, we implemented a preprocessing step to aggregate the 3-hourly meteorological data into daily averages. This approach involved calculating the mean values of relevant meteorological parameters (such as temperature, humidity, wind speed) across the 24-hour period to derive daily weather data points. By transforming the data in this manner, we ensured that our model could predict daily weather conditions accurately despite the API's limitation of not providing direct daily meteorological data.

5.1.7 Future Work

Our approach faces certain limitations and challenges that could be addressed in future work, including:

1) Dataset Diversity

Increasing the diversity of meteorological datasets by integrating data from multiple sources to ensure comprehensive coverage of detailed data specific to each locality within the governorates. Starting with Cairo and Giza allows us to establish a solid foundation before expanding to include additional governorates.

2) Advanced Machine Learning Algorithms

Exploring more advanced machine learning algorithms tailored for regression tasks on meteorological data, such as ensemble methods like Random Forests or Gradient Boosting.

3) Enhanced OpenWeather API Data Availability

Integrating the paid versions of OpenWeather API, which provides both hourly and daily meteorological data, to enhance data availability across all governorates of Egypt. This approach aims to improve the temporal resolution of the predicted solar irradiance, ensuring timely and accurate predictions.

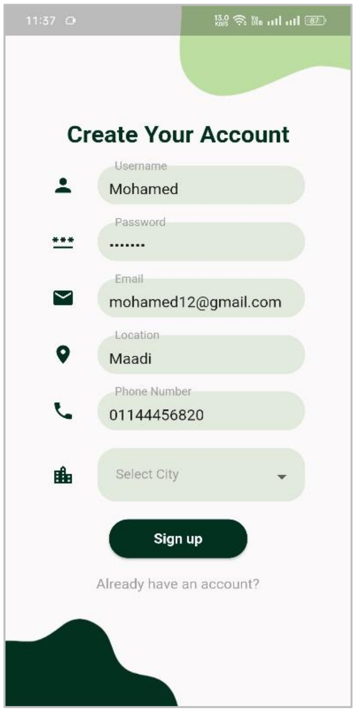
4) Integrated Online Payment in the Marketplace

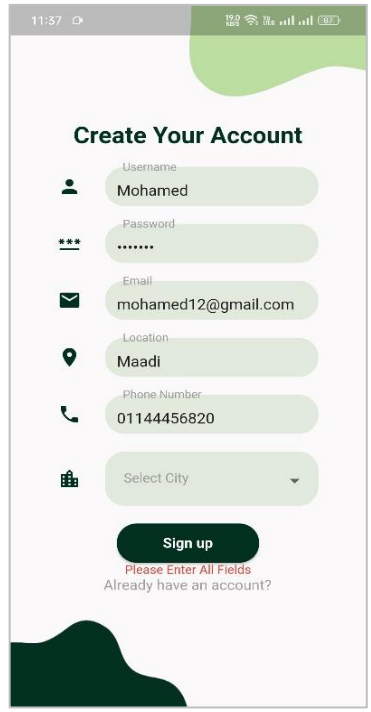
Facilitating the entire purchase process within the marketplace by incorporating online payment options for consumer-to-consumer (C2C) transactions. This enhancement will simplify the buying and selling of solar components, allowing users to complete their transactions directly through the app.

5.2 Testing

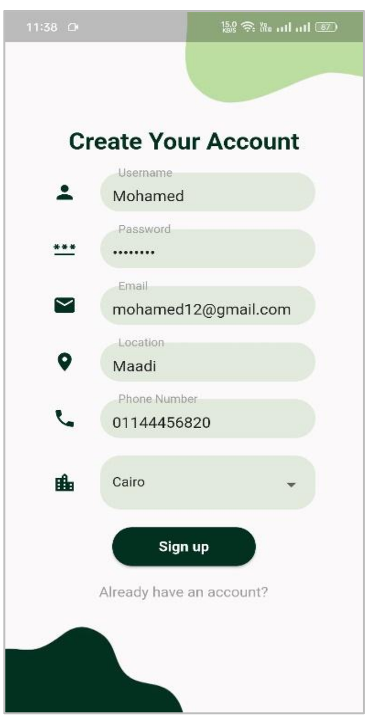
5.2.1 Resister

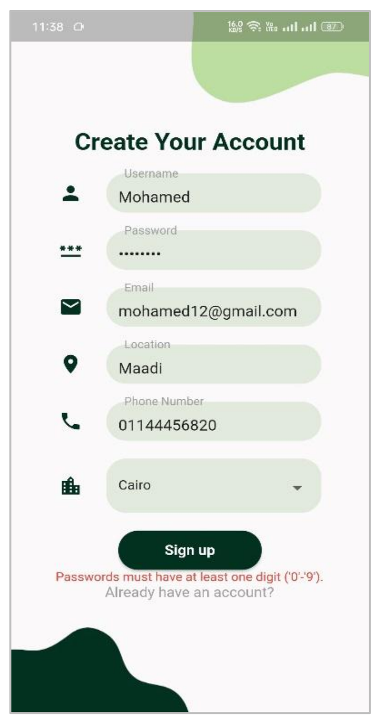
Test Case 1: This test case checks the registration process for when the user leaves the city field empty. An error message is displayed, and the registration fails (Enter all fields).



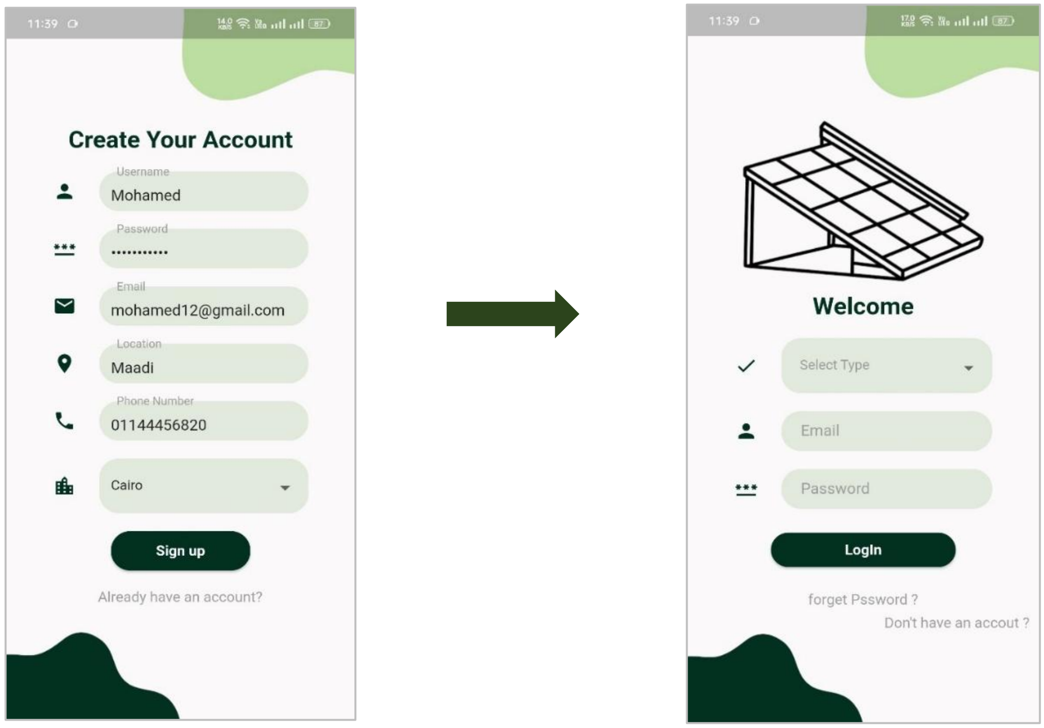


Test Case 2: This test case checks the registration process when the user enters a password without digits. An error message is displayed, and the registration fails.



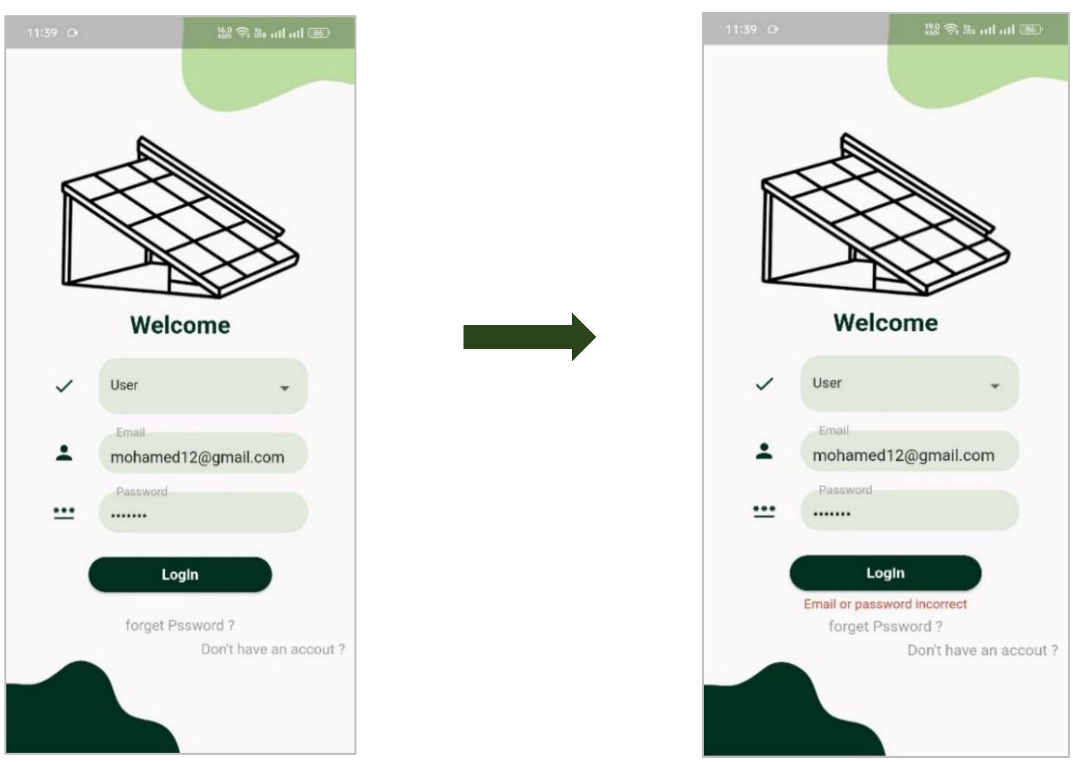


Test Case 3: This test case checks the registration process when the user enters all valid information and successfully registers.

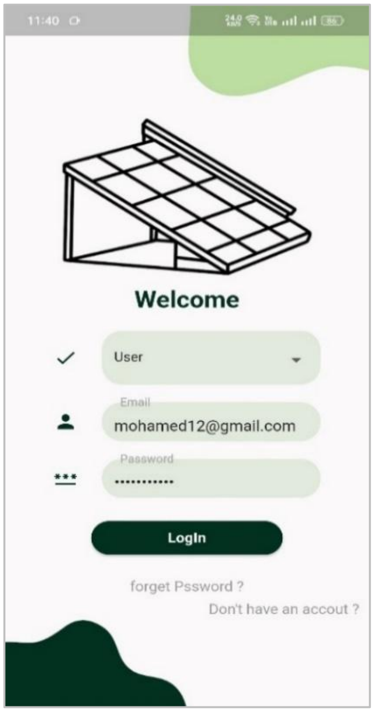


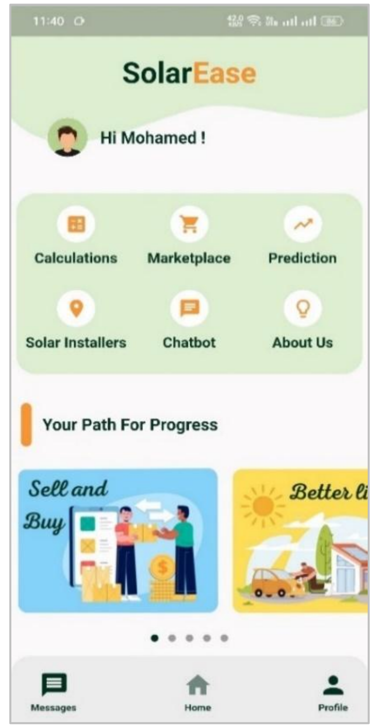
5.2.2 Login

Test Case 1: This test case checks the login process when the user enters incorrect information. An error message is displayed, and the login fails.



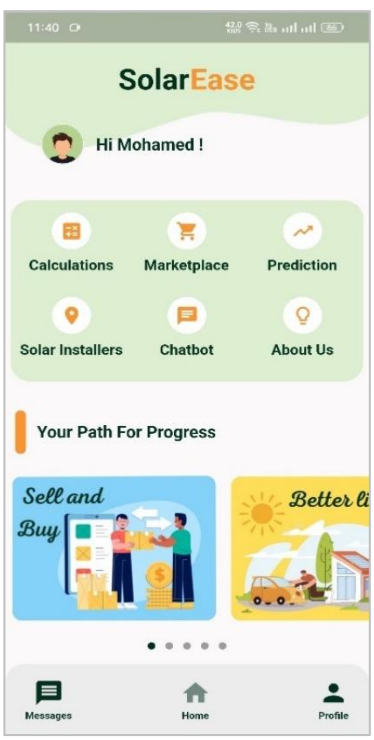
Test Case 2: This test case checks the login process when the user enters all valid information and successfully logs in and goes to the home page.



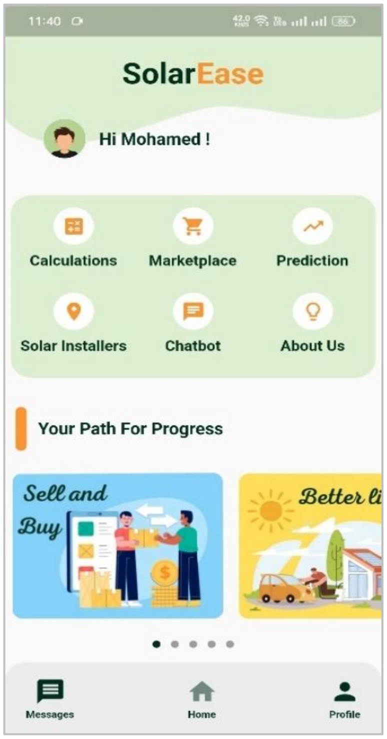


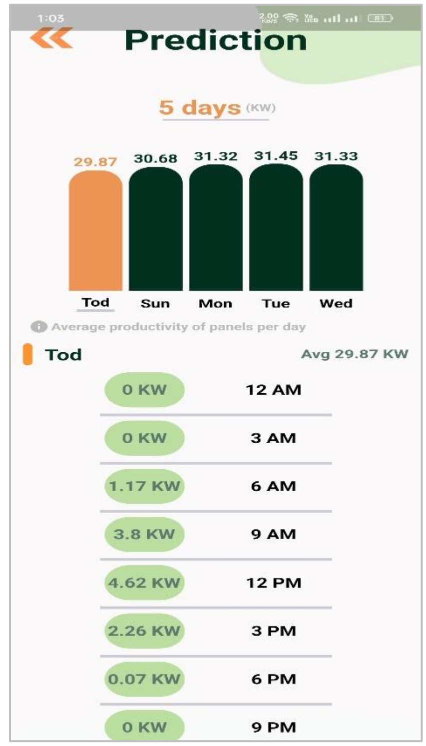
5.2.3 Prediction

Test Case 1: This test case for prediction where user clicks on prediction icon, but he hasn't filled system size in his profile.

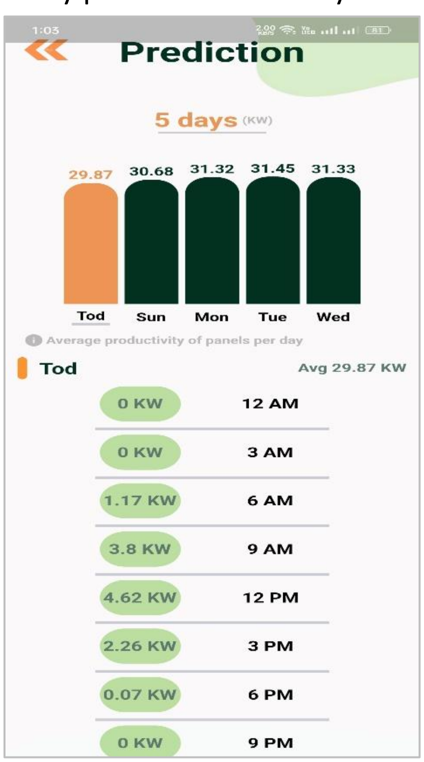


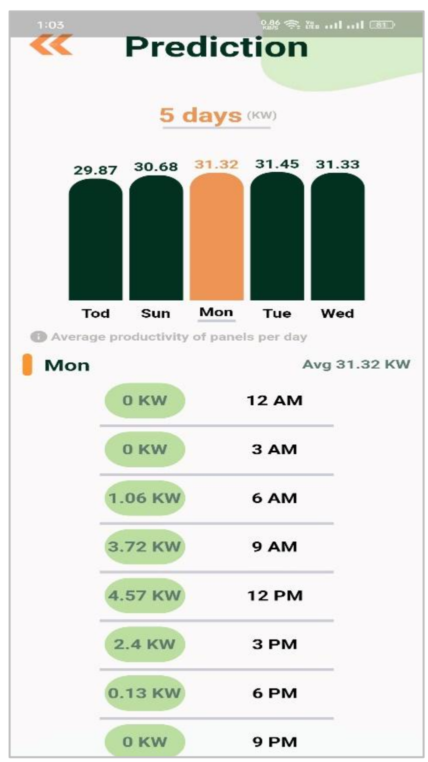
Test Case 2: This test case for prediction where user clicks on prediction icon to see his system prediction based on his location for 5 days from the current day and each day has 8 hours prediction with gap 3 hours between them.





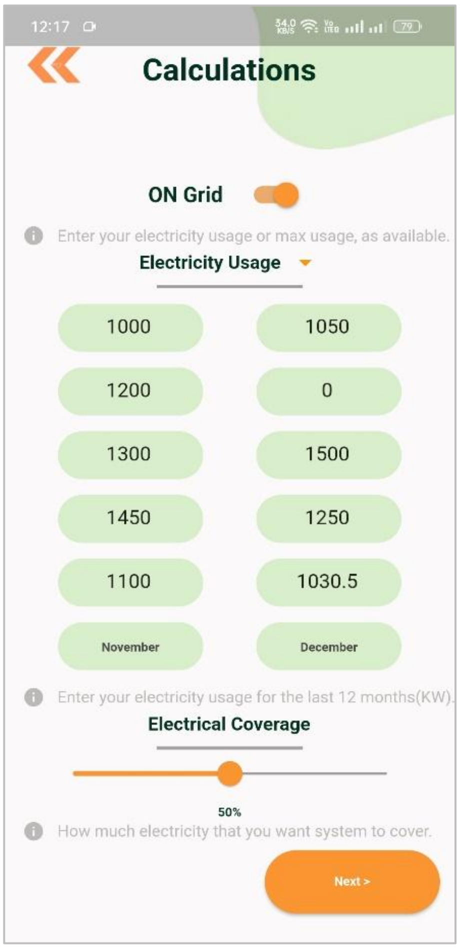
Test Case 3: This test case for prediction where user clicks on another day's icon to see this it's hourly prediction for his system.

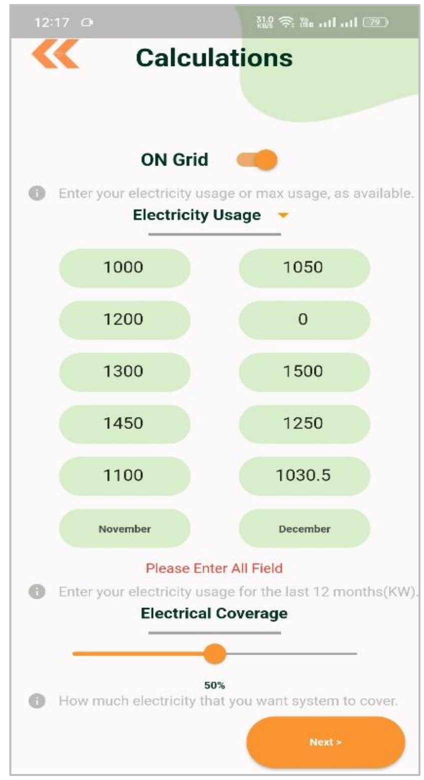


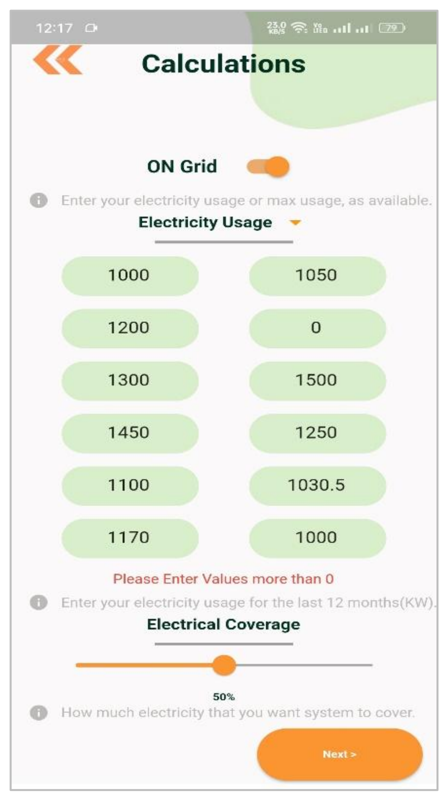


5.2.4 Calculation

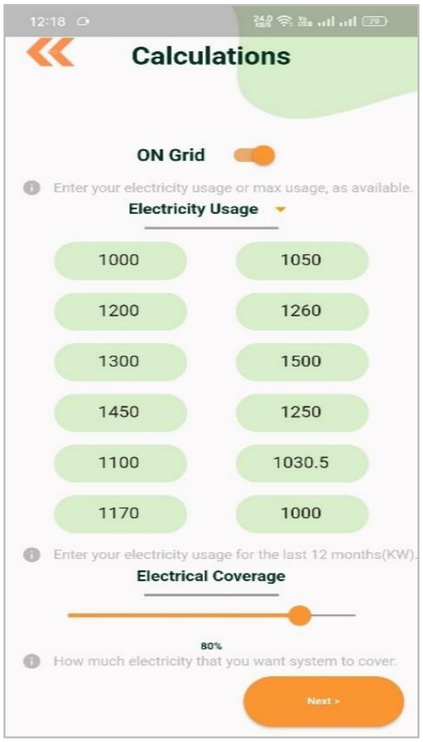
Test Case 1: This test case for calculator where user fills in his electricity consumption for last 12 months but leaves fields unfilled and one field with a zero value, gets error message.

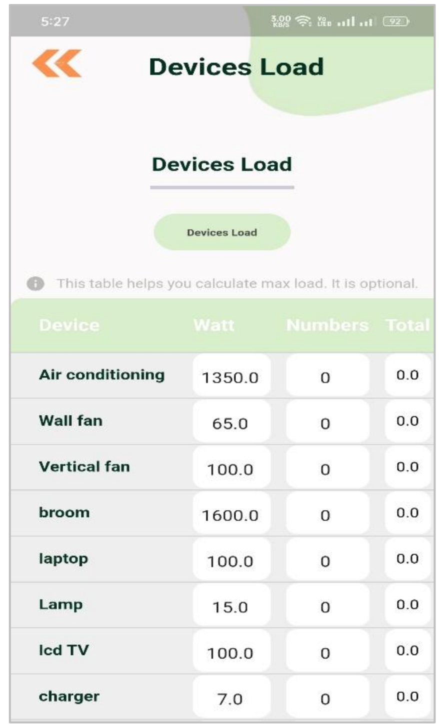




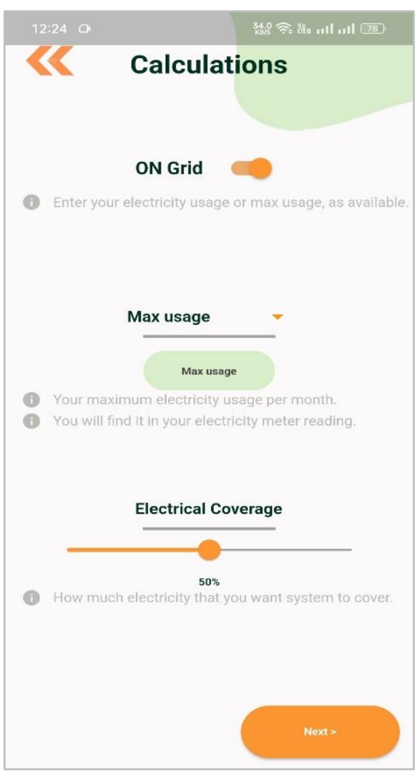


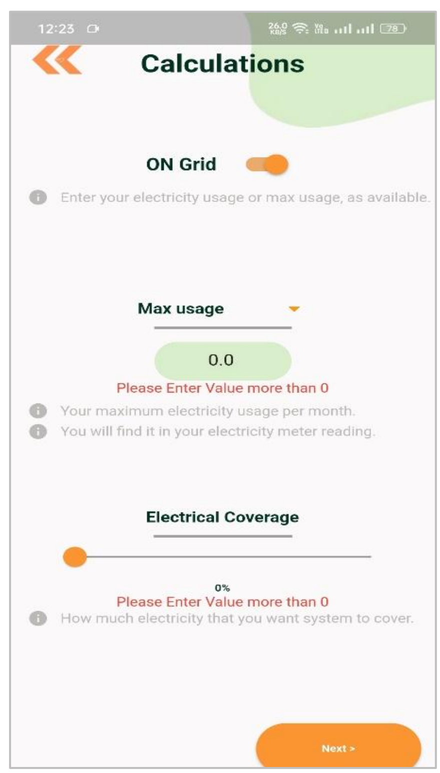
Test Case 2: This test case User enters electricity consumption for last 12 months and desired electricity coverage correctly, then click next. System navigates to device load page.



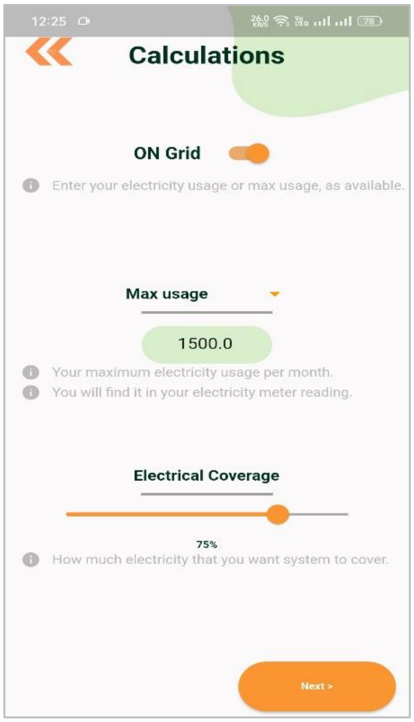


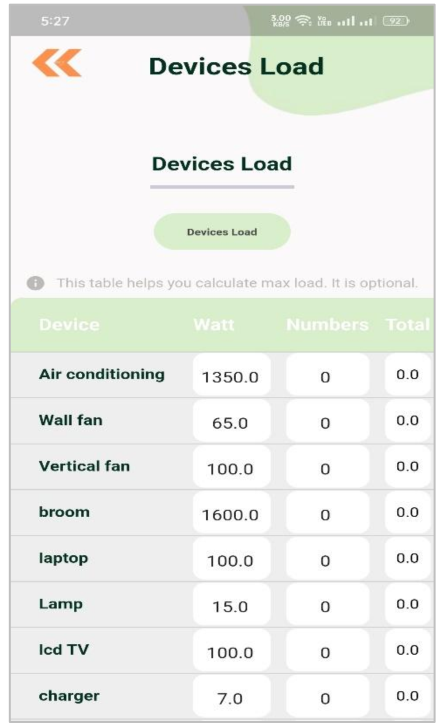
Test Case 3: This test case user enters zero values for max usage and electricity coverage, then clicks next. The system prevents the user from navigating to the device load page.



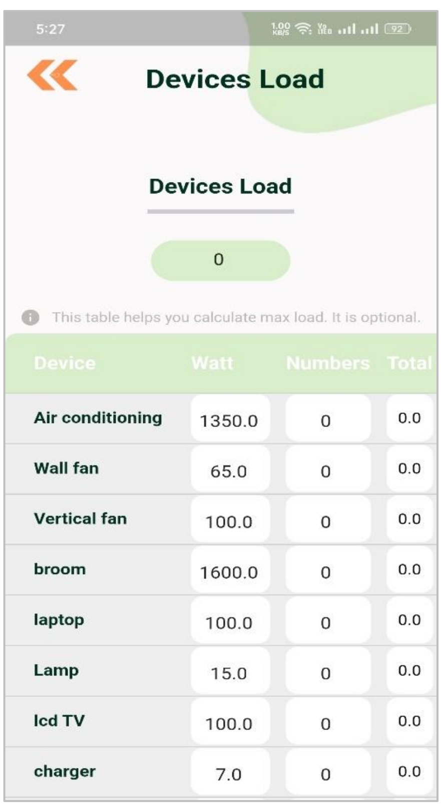


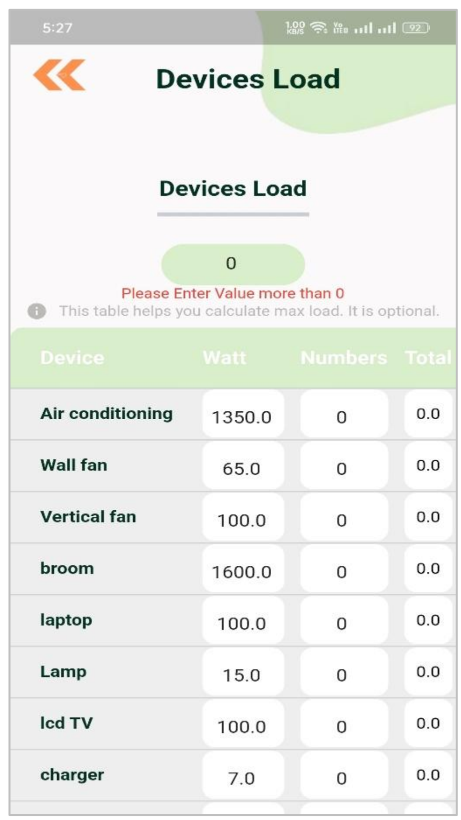
Test Case 4: This test case user enters the max usage value and desired electricity coverage correctly, then clicks next. The system navigates to the device load page.



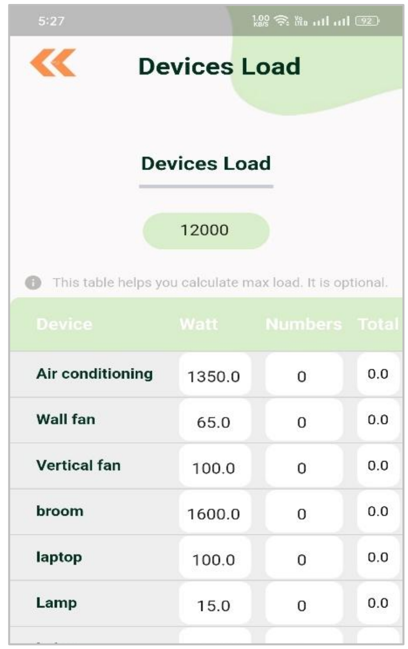


Test Case 5: This test case for calculation where user enter 0 as his devices load and he gets error for that because it should be more than 0.



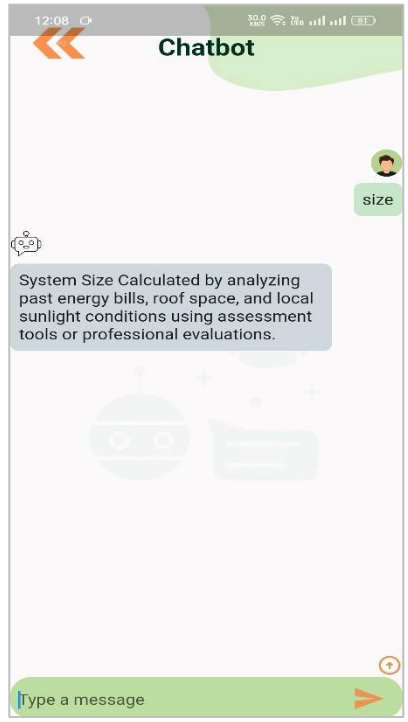


Test Case 6: This test case for calculation where user can either enter his devices load at its field at the top or can choose devices and its quantities and load then the app will calculate for him appearing at the bottom in total load field then clicks on next to see his calculations.

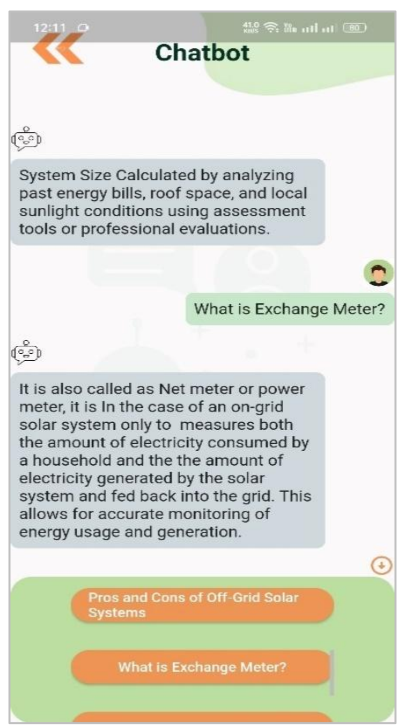


5.2.5 Chatbot

Test Case 1: This test case, user asks about solar system size and the chatbot provides an appropriate response.

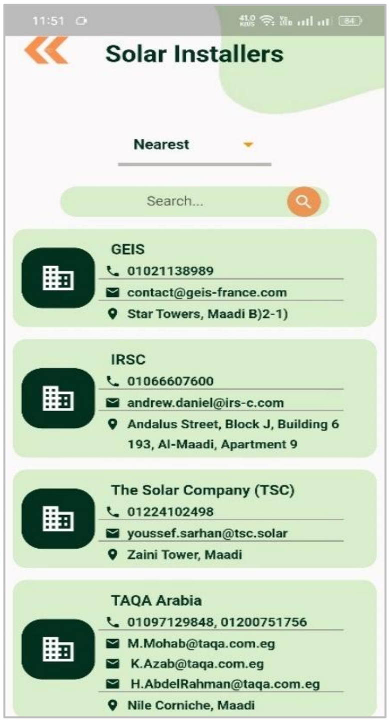


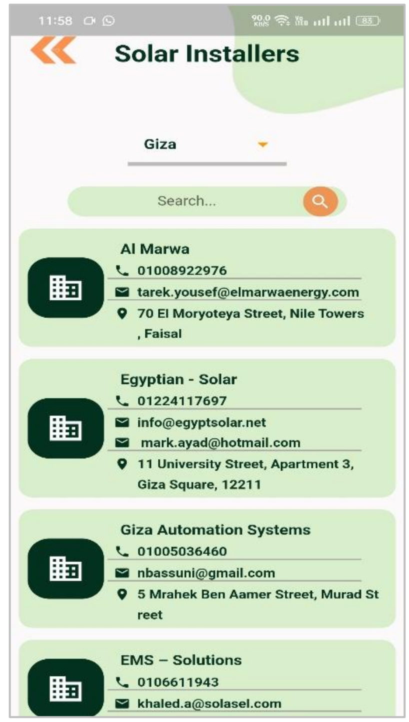
Test Case 2: This test case navigate to the category-based chatbot and choose "What is the exchange meter?" and the chatbot answers appropriately.



5.2.6 Solar Installers

Test Case 1: This test case is for finding solar Installer companies ordered by the nearest and user filter by government and result also ordered by the nearest through government.

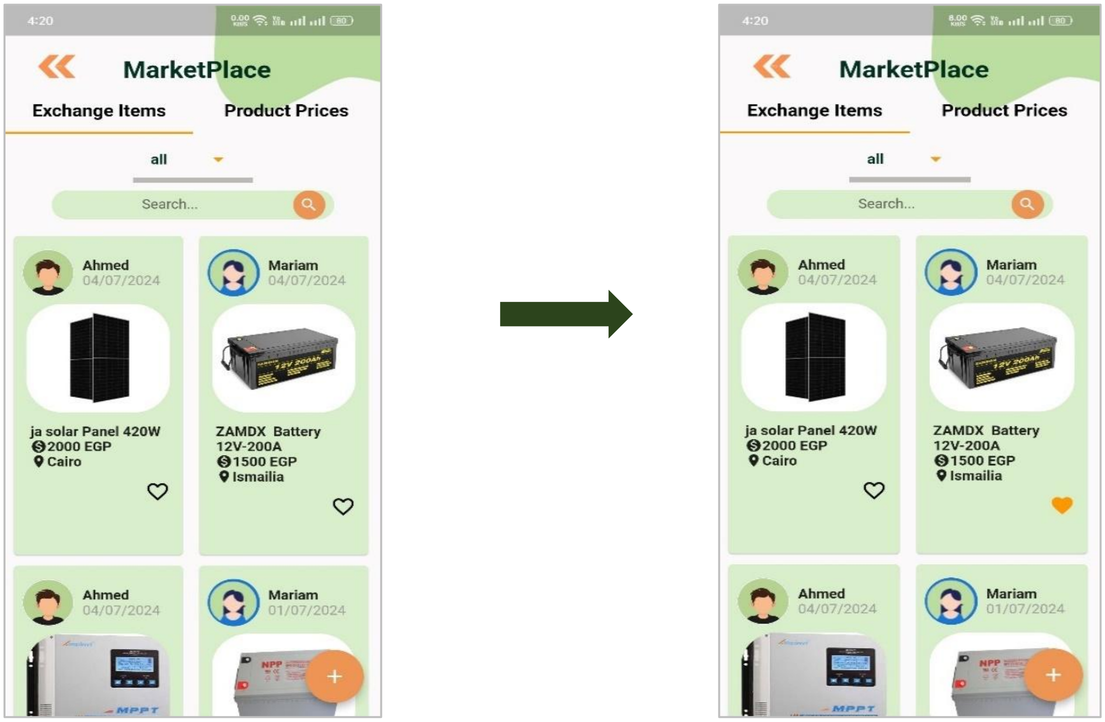




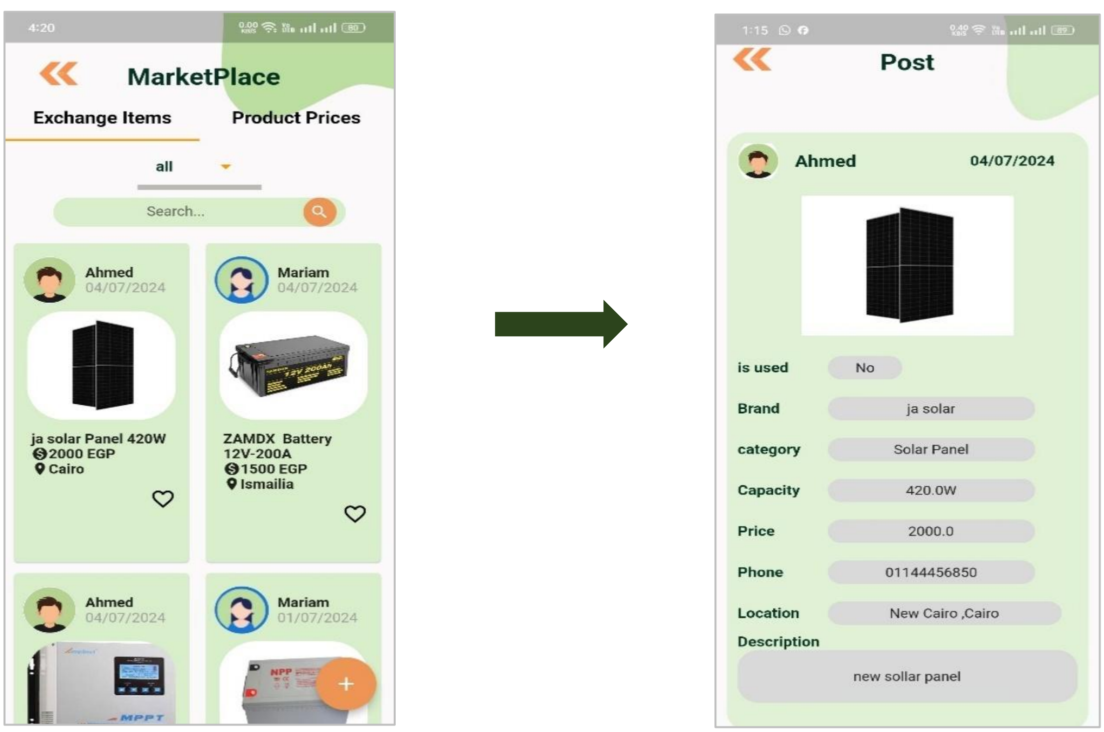
Test Case 2: This test case after filtering by the government the user search in this government by Installer company's name which is the returned result matches the search.

5.2.7 Marketplace (Exchange Items)

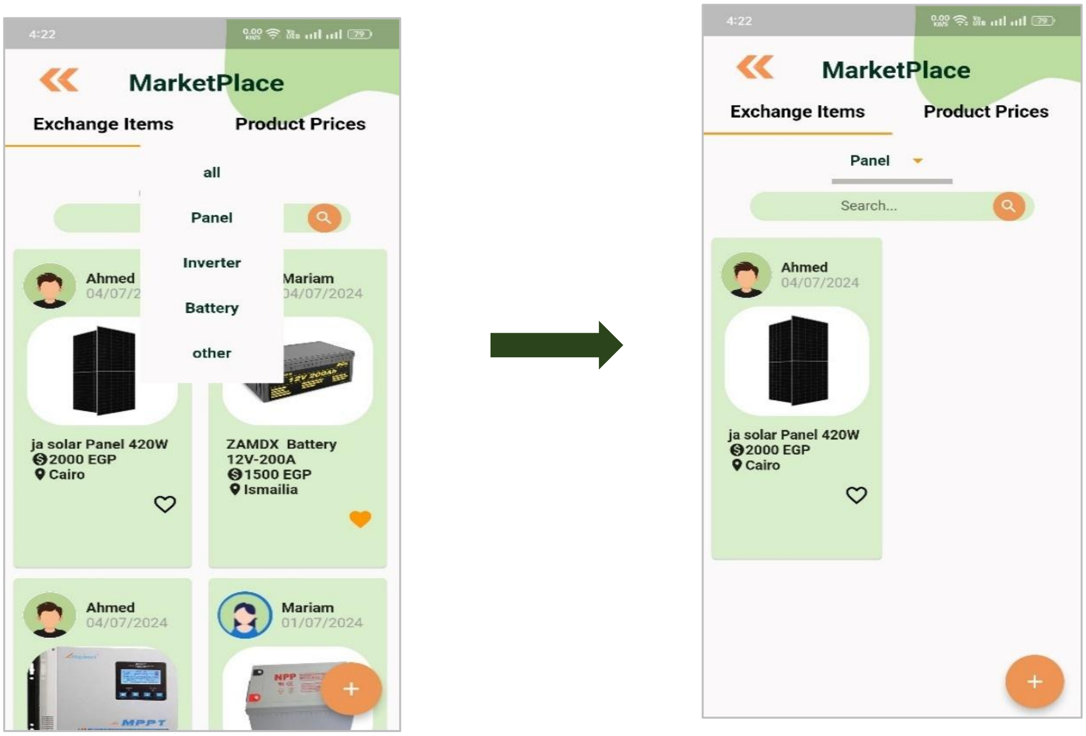
Test Case 1: This test case is for marketplace in section exchange items where user clicks on favorite icon to mark an item as a favorite.



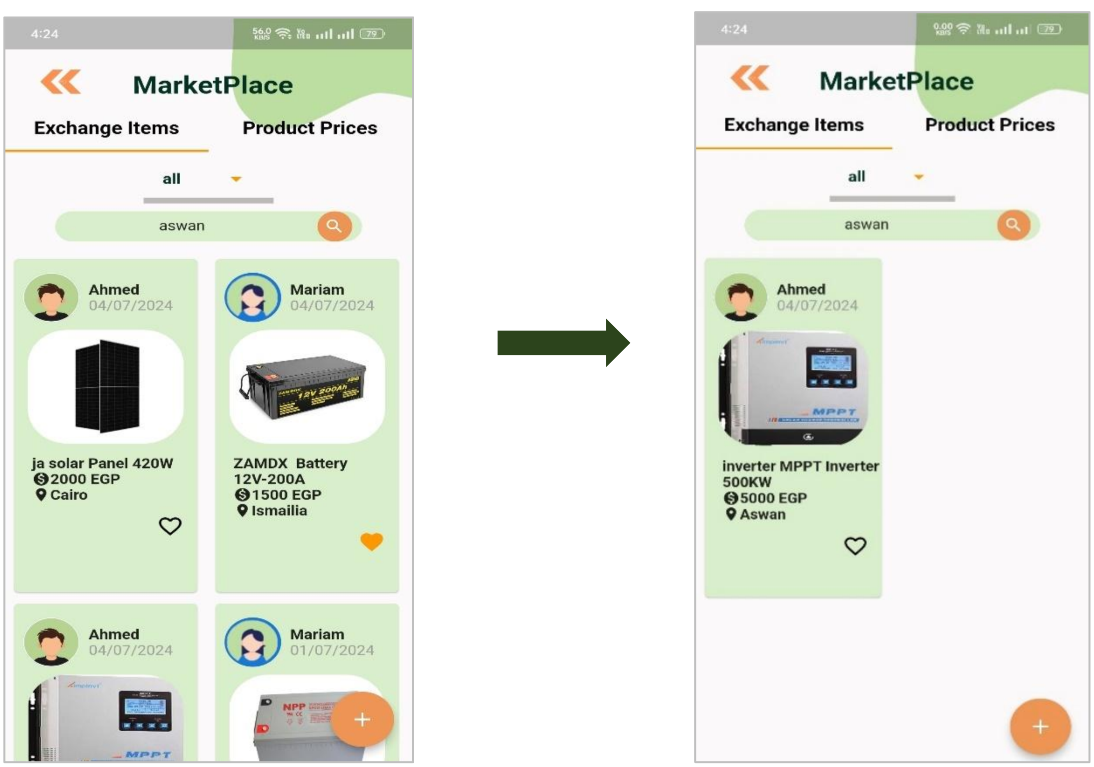
Test Case 2: This test case for marketplace in section exchange items where user clicks on specific exchange item to open a new page having all detailed information.



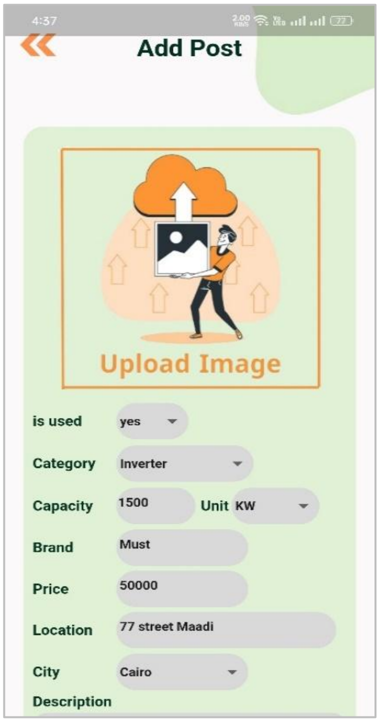
Test Case 3: This test case for marketplace in section exchange items where user filter exchange items by category choosing panels to see related exchange items.



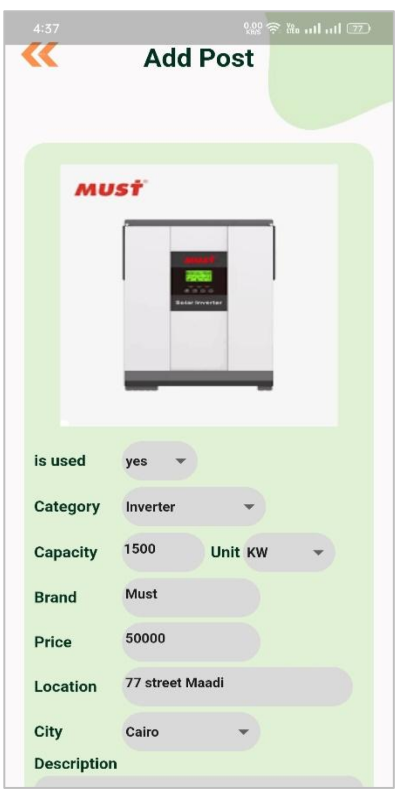
Test Case 4: This test case for marketplace in section exchange items where user search by city among exchange items to see related ones.

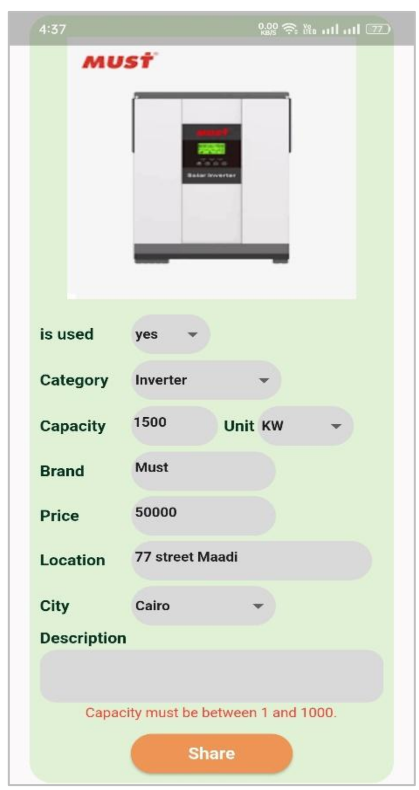


Test Case 5: This test case is for a marketplace where user didn't upload image while adding item. An error message is displayed, and adding exchange item fails (select image).

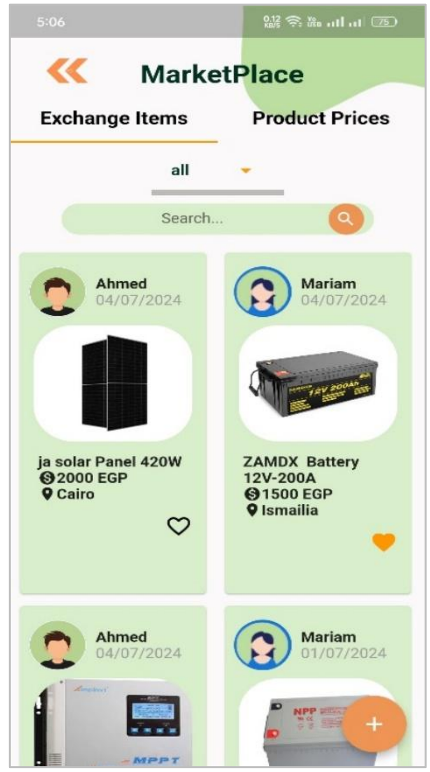


Test Case 6: This case for marketplace in section exchange items where user enter invalid number in capacity field while adding item and gets an error message.



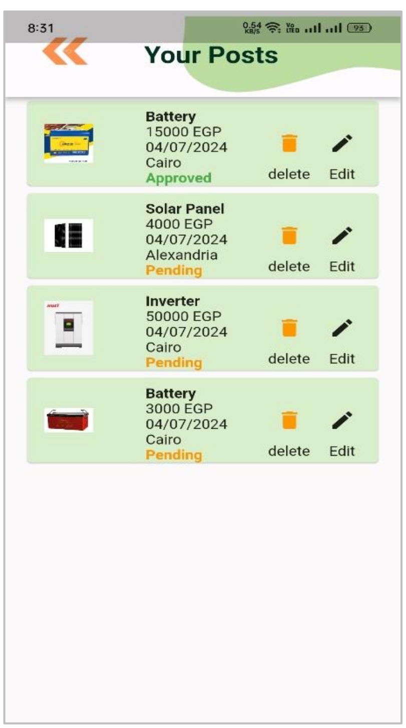


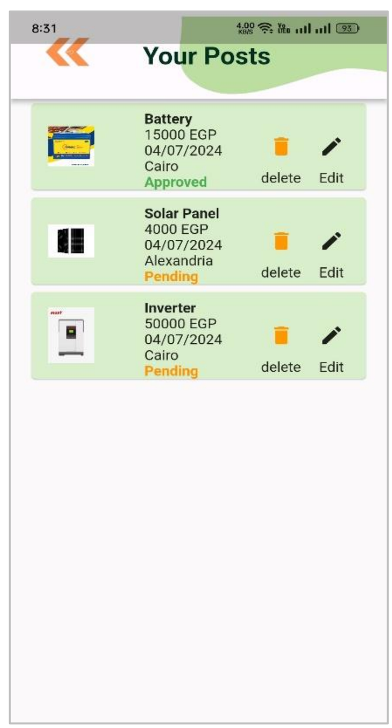
Test Case 7: This test case for marketplace in exchange items where user enter all data field correctly then clicks on share button (post will be shared after admin approval).



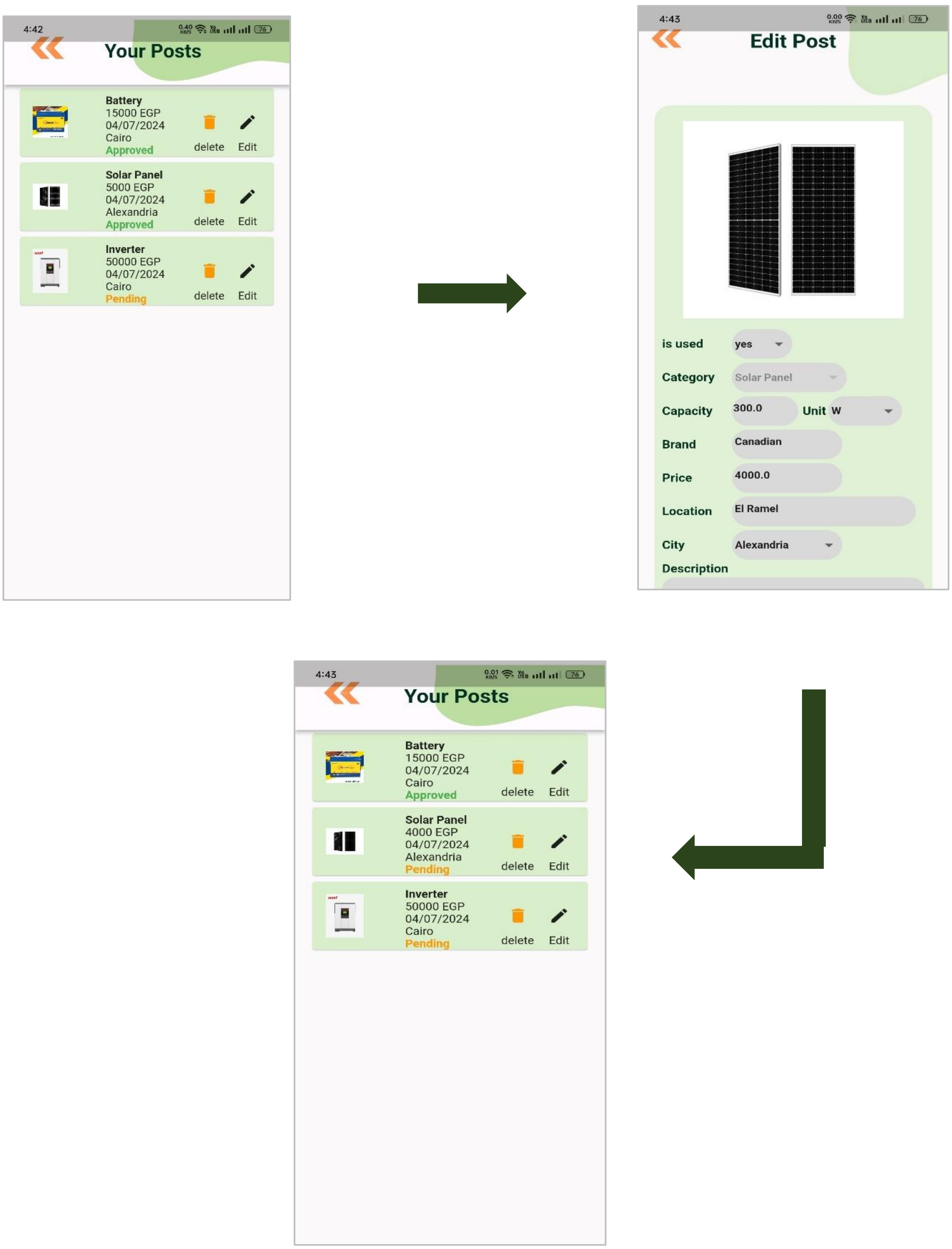
5.2.8 Your Posts

Test Case 1: This test case for deleting posts where the user can delete posts.



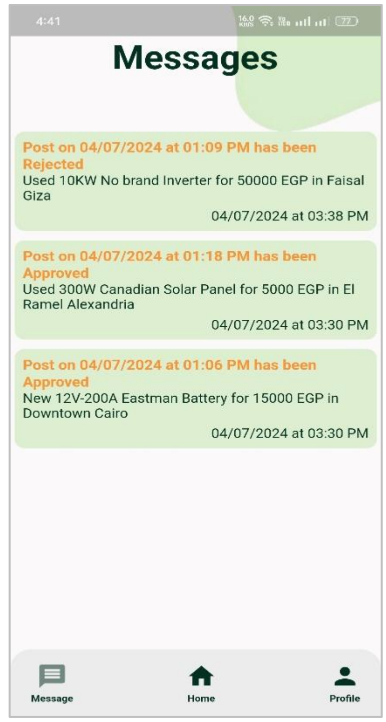


Test Case 2: This test case for edit posts where the user can edit post like change price and enter all data field correctly then the post status changed from Approved to Pending.



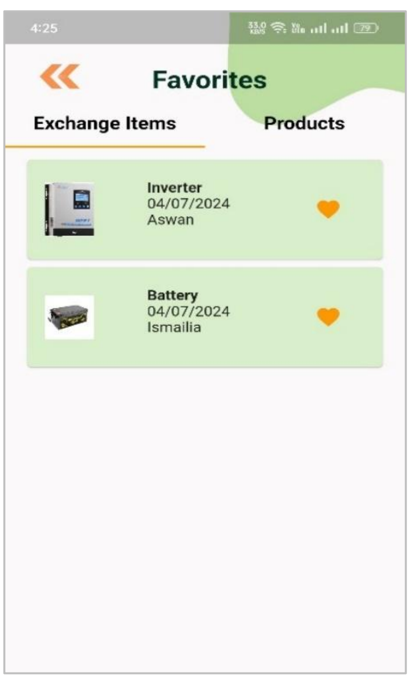
5.2.9 Filter Posts

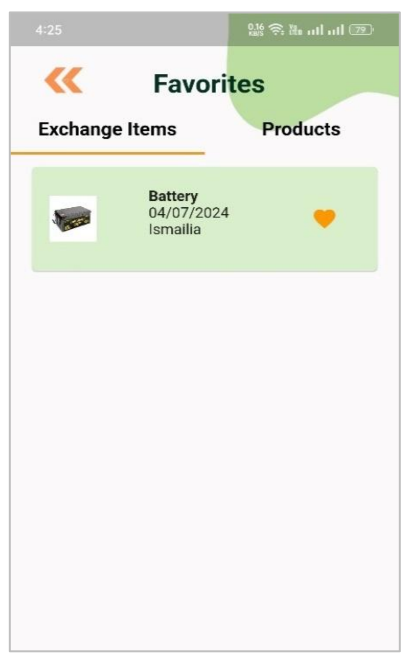
Test Case 1: This test case for admin when filtering user's posts in exchange items section in marketplace and user gets a message with post approval or rejection.



5.2.10 Favorites (Exchange Items)

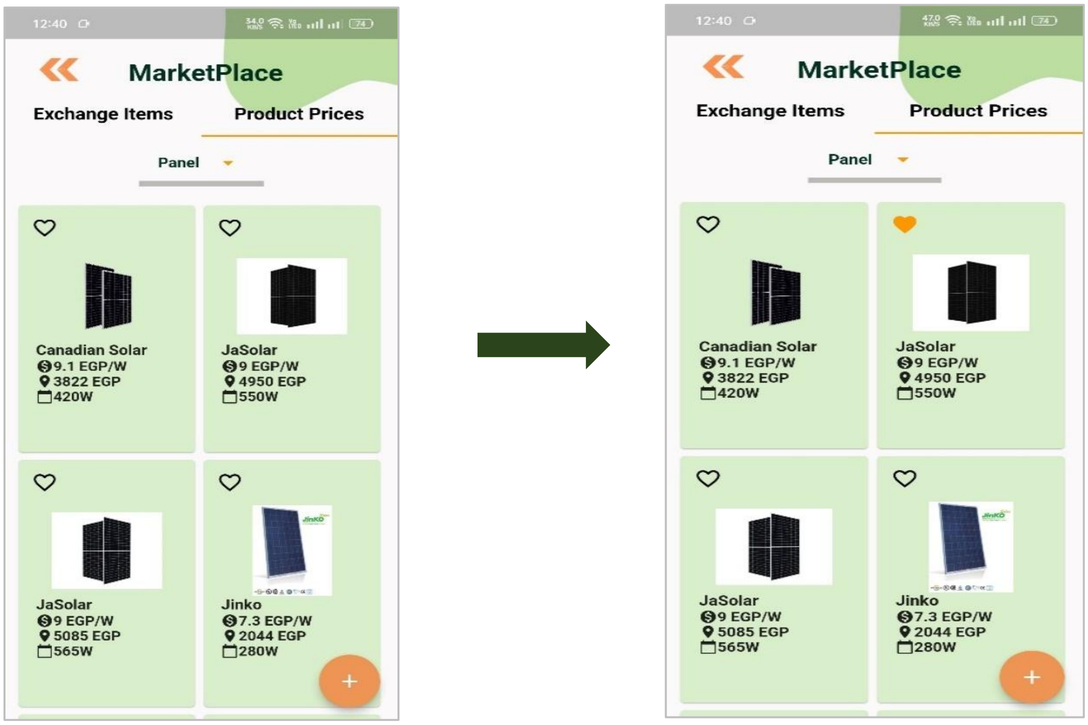
Test Case 1: This test case for favorites in section exchange items where user unmark product to remove this exchange item from favorite.



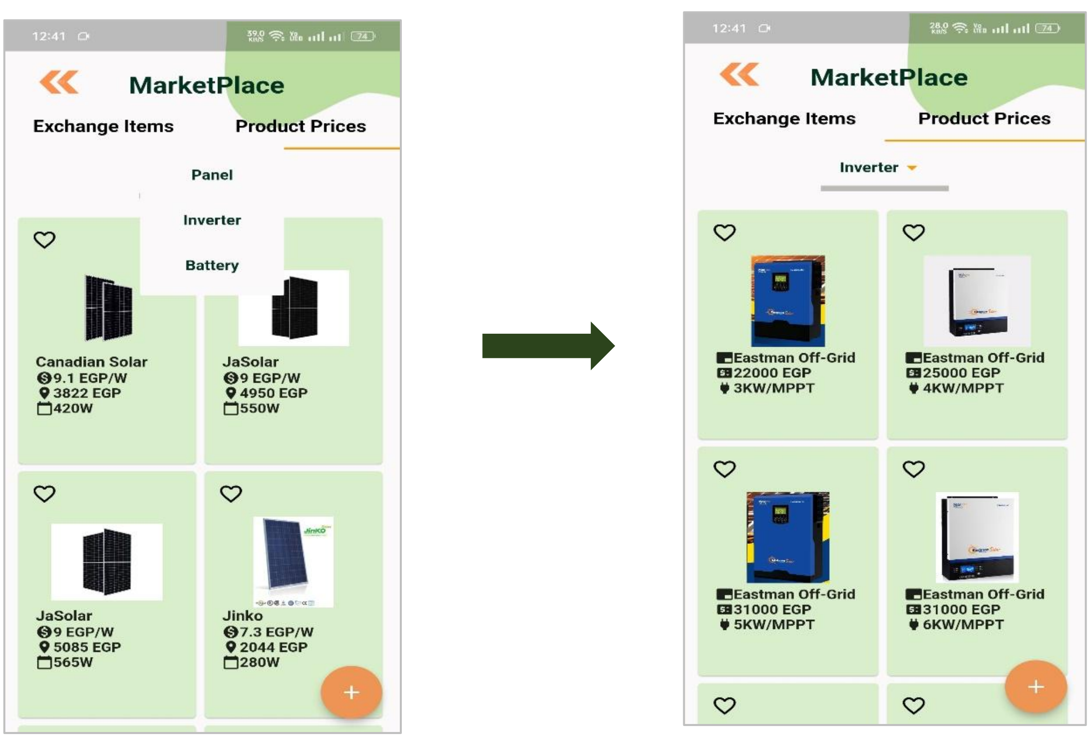


5.2.11 Marketplace (Product Prices)

Test Case 1: This test case is for the Product prices section in Marketplace where the user can mark a product as a favorite.

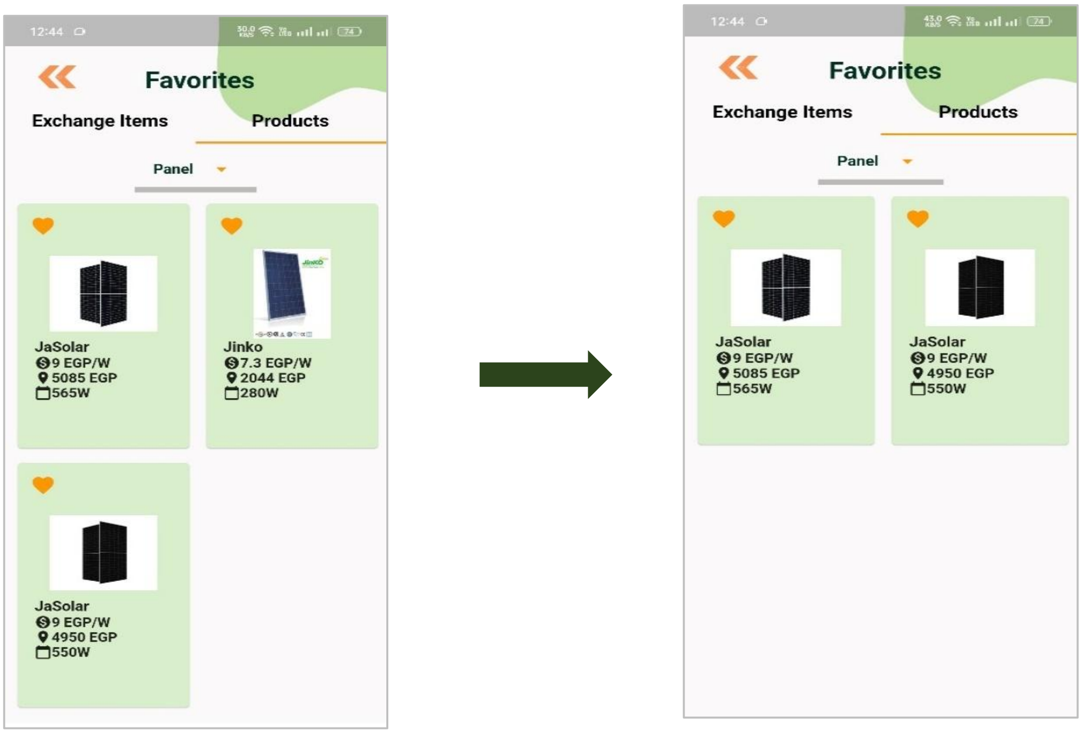


Test Case 2: This test case for Product prices where the user can filter products (Inverter, Panel, Battery) which is the returned products are based on selected category (Battery).

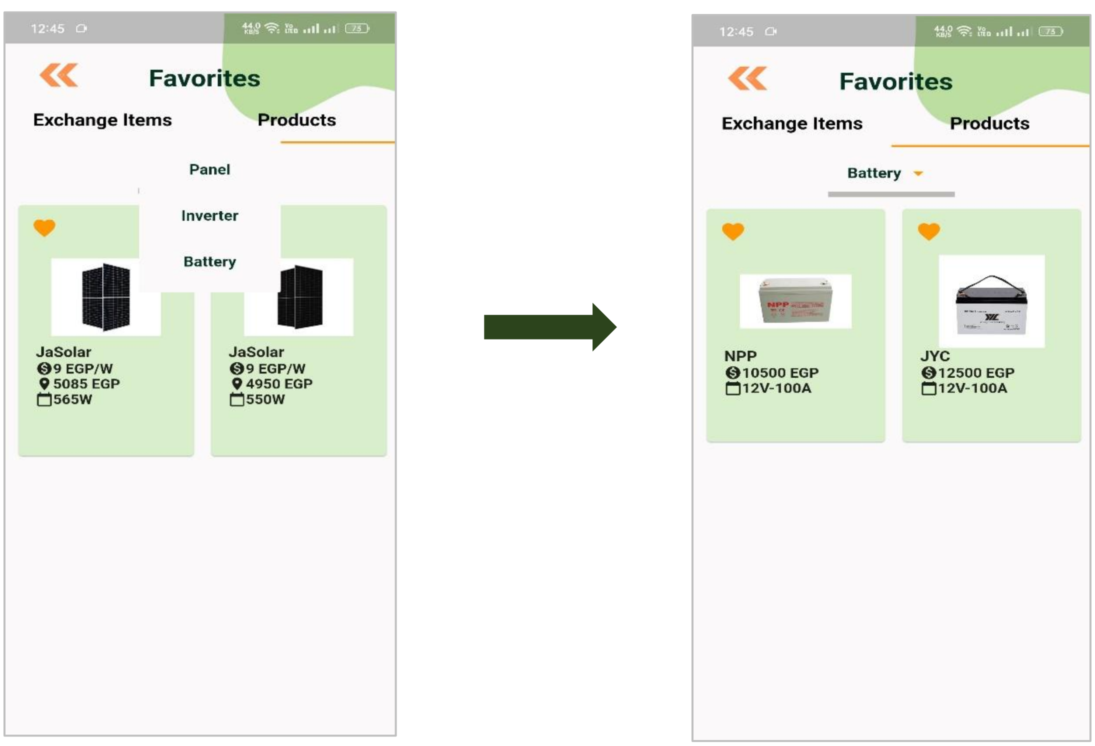


5.2.12 Favorites (Product Prices)

Test Case 1: This test case for favorites products where user clicks on heart icon to unmark it to remove this product from favorite products.

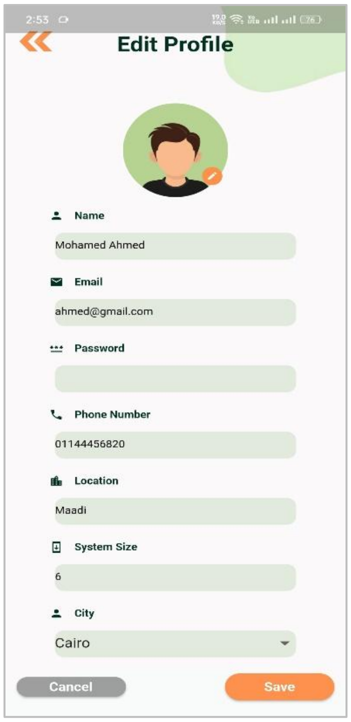


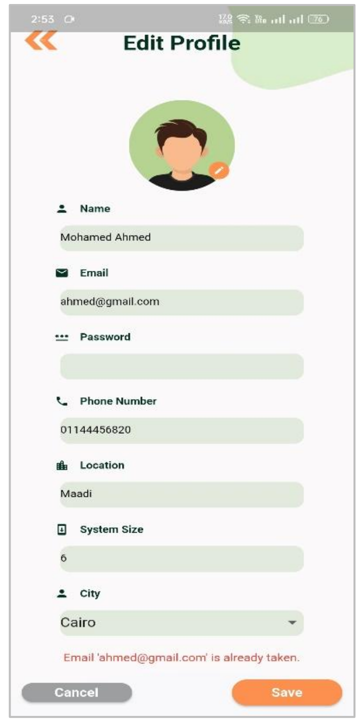
Test Case 2: This test case for favorites products where user filters favorite products and select battery to see its favorites.



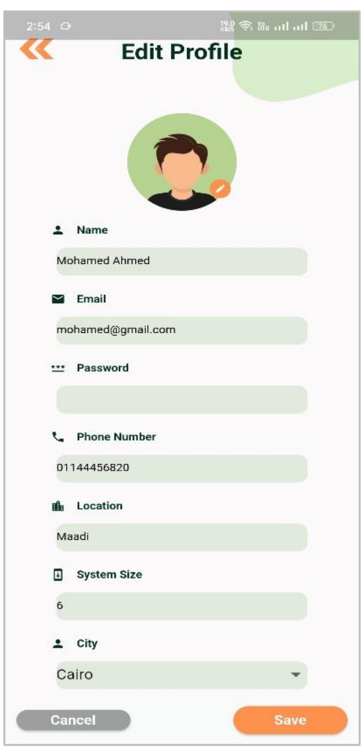
5.2.13 Edit Profile

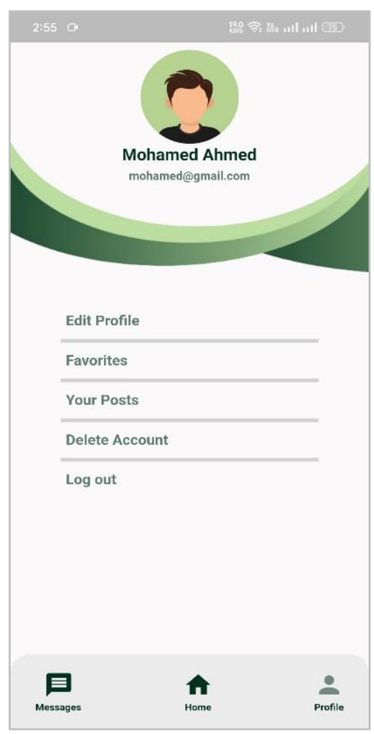
Test Case 1: User attempts to edit profile and enter email that is already taken by another user. The system displays error message indicating that the email address is already in use.





Test Case 2: User attempts to edit profile and enter email that is already taken by another user. The system displays error message indicating that the email address is already in use.





References

- Muhammed A. Hassan et al. (2017). Exploring the potential of tree-based ensemble methods in solar radiation modeling. https://www.sciencedirect.com/science/article/pii/S0306261917308437

- Meenal and Immanuel Selvakumar (2018). Assessment of SVM, empirical and ANN based solar radiation prediction models with most influencing input parameters. https://www.sciencedirect.com/science/article/pii/S0960148117312016

- Hui He, Nanyan Lu, Yongjun Jie, Bo Chen, and Runhai Jiao (2020). Probabilistic solar irradiance forecasting via a deep learning-based hybrid approach. https://onlinelibrary.wiley.com/doi/10.1002/tee.23231

- Basaran, Özçift, and Kılınç (2019). A New Approach for Prediction of Solar Radiation with Using Ensemble Learning Algorithm. https://link.springer.com/article/10.1007/s13369-019-03841-7

- Benjamin Chanakot and Charun Sanrach (February 2024). Developed a Thai-language chatbot for analyzing mosquito-borne diseases. https://www.researchgate.net/publication/376271629_A_Thai-language_chatbot_analyzing_mosquito-borne_diseases_using_Jaccard_similarity

- Suphakit Niwattanakul et al. (2013). Using of Jaccard Coefficient for Keywords Similarity https://www.researchgate.net/publication/317248581_Using_of_Jaccard_Coefficient_for_Keywords_Similarity