DE-41-EE

Rohaib Bhatti

Anam Sheikh

Shifa Hamid

Demand Response in PV Dominated Active Buildings

COLLEGE OF

ELECTRICAL AND MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

RAWALPINDI

2023

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING

DE-41 EE

PROJECT REPORT

Demand Response in PV-Dominated Active Buildings

Submitted to the Department of Electrical Engineering

in partial fulfillment of the requirements

for the degree of

Bachelor of Engineering

in

Electrical

2023

DECLARATION

We affirm that the content presented in this Project Thesis is original and has not been submitted in support of any other degree or qualification at this or any other university or institute of learning. We take full responsibility for any act of plagiarism and acknowledge that we may face disciplinary actions, including degree cancellation, for any proven offense of plagiarism.

Rohaib Bhatti

NUST ID: ______________ Signature: ______________

Anam Sheikh

NUST ID: ______________ Signature: ______________

Shifa Hamid

NUST ID: ______________ Signature: ______________

CERTIFICATE OF APPROVAL

It is to certify that the project "Demand Response in PV Dominated Active Buildings" was done by NS Rohaib Bhatti, NS Anam Sheikh, and NS Shifa Hamid under the supervision of Dr. Sarmad Majeed Malik and the co-supervision of Assistant Professor Kamran Aziz Bhatti. This project is submitted to the Department of Electrical Engineering, College of Electrical and Mechanical Engineering (Peshawar Road Rawalpindi), National University of Sciences and Technology, Pakistan in fulfillment of requirements for the degree of Bachelor of Engineering in Electrical Engineering.

Students:

1-Rohaib Bhatti
NUST ID: ______________________ Signature: _____________________

2-Anam Sheikh
NUST ID: ______________________ Signature: _____________________

3-Shifa Hamid
NUST ID: ______________________ Signature: _____________________

Approved by:

Project Supervisor: __________________ Date: ___________________

Dr. Sarmad Majeed Malik

Project Co-Supervisor: __________________ Date: ___________________

Kamran Aziz Bhatti

ACKNOWLEDGMENTS

We would like to express our sincere gratitude to our supervisor, Dr. Sarmad Majeed Malik, for his invaluable guidance, encouragement, and support throughout the course of our project. His expertise, insight, and dedication have been instrumental in shaping this project, and I am deeply grateful for the opportunity to work under his supervision.

We would also like to thank our co-supervisor, Assistant Professor Kamran Aziz, for his insightful feedback and guidance, which have helped us to refine our research and to develop a deeper understanding of the subject matter.

Finally, we would like to express our thanks to all the participants who generously gave their time and expertise to contribute to this project. Their contributions were essential to the success of this project, and we are truly grateful for their support.

ABSTRACT

This project aims to develop an accurate power forecast of a photovoltaic (PV) system for active buildings, using machine learning (ML) models. The objective is to create a demand response system that regulates energy usage in active buildings by providing recommendations for efficient power usage to the user.

In this project, we have collected historical data on environmental factors such as solar irradiance and compared various ML models to select the best one based on its accuracy. The selected ML model is implemented to generate the power forecast of the PV system. To create the demand response system, a mobile application is developed that uses the power forecast to provide recommendations for efficient power usage to the user.

Our results show that the ML model accurately predicts the power output of the PV system and that the demand response system based on the power forecast can help to manage energy consumption in active buildings. This project demonstrates the potential of ML-based power forecasting and demand response systems in achieving sustainable and energy-efficient buildings.

SUSTAINABLE DEVELOPMENT GOALS

This project is linked with Sustainable Development Goals (SDGs) 7 and 11, which aim to promote affordable and clean energy, and sustainable cities and communities. By promoting the use of solar panels to produce clean energy, this project contributes to SDG 7. Additionally, solar energy is a reliable and sustainable source of energy, which contributes to SDG 11 by promoting sustainable cities and communities. Overall, this project aligns with the United Nations' goals for sustainable development by promoting the use of clean and sustainable energy sources for building a better and more equitable future.

Figure 1 Sustainable Development Goals

TABLE OF CONTENTS

Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

Latin Letters

A area

E energy

H annual average solar irradiation

P power

PR performance ratio

r solar panel yield

T time

Ω Ohm

Acronyms

ARIMA Autoregressive Integrated Moving Average

CNN Convolutional Neural Network

DR Demand Response

FRNN Full Recurrent Neural Network

IoT Internet of Things

GCPV Grid-connected Photovoltaic System

LSTM Long Short-term Memory

ML Machine Learning

PCA Principal Component Analysis

PV Photovoltaic

RF Random Forest

RNN Recurrent Neural Network

SVR Support Vector Regression

NARX Non-Linear Autoregressive Exogenous Model

Chapter 1 - Introduction

Demand Response (DR) is a concept that is becoming increasingly important in the realm of energy management. With the advent of renewable energy sources such as solar, wind, and hydropower, the need for an efficient and reliable DR system has become more pronounced. A demand response system is a mechanism or program that allows electricity consumers to actively participate in managing their energy consumption in response to fluctuations in electricity supply and demand. It involves modifying electricity usage patterns or reducing power consumption during peak demand periods or when the grid is stressed. This system enables grid operators to balance electricity supply and demand more effectively, reduce strain on the grid, avoid blackouts or brownouts, and improve overall system reliability [20]. It can also provide financial incentives to participants for their willingness to adjust their energy usage based on grid conditions. DR is essentially a mechanism that allows for the adjustment of electricity demand in response to changes in supply, typically to help balance the grid during periods of peak demand [19]. In recent years, DR has gained popularity due to its potential to reduce energy costs, improve grid reliability, and promote the integration of renewable energy sources into the grid [18].

The focus of this thesis is to develop a DR system for PV-dominated active buildings that utilizes machine learning techniques to achieve accurate power forecasting. PV-dominated active buildings refer to buildings that are designed and equipped to actively generate and utilize solar energy through photovoltaic (PV) systems. These buildings prioritize the use of renewable energy sources, particularly solar power, to meet their energy needs [17]. These buildings often employ advanced energy management systems to optimize the use of solar power, store excess energy in batteries or other storage systems, and efficiently distribute energy throughout the building. Around 4.4% of the energy worldwide comes from solar whereas the 11.5% of the renewable energy is extracted from solar [16]. Renewable energy sources contribute 38% to the total energy production [16]. The proposed system has used historical data on environmental factors such as solar irradiance to predict the power output of a photovoltaic (PV) system. The historical data has been analyzed using various machine learning models, including linear regression, Prophet, Random Forest, and ARIMA, to identify the model that provides the best accuracy in predicting the power output. The selected models have been used to predict the power output of the PV system, and the predicted data has been sent to a mobile application that provides users with recommendations for efficient power usage.

In addition to power forecasting and demand response, the proposed system also incorporates home automation features. Users can turn on and off home appliances remotely via the mobile application, allowing for better control over energy usage. By accurately predicting the power output of a PV system and providing recommendations for efficient power usage, the system helps in managing energy consumption and promote the integration of renewable energy sources into the grid. The integration of home automation features also provides users with better control over energy usage, allowing them to avoid a blackout.

1.1 Objectives

- Develop accurate power forecasting models for PV systems, utilizing historical data of environmental factors such as solar irradiance, wind direction and wind speed etc.

- A user-friendly mobile application has been created to integrate with the power forecasting models. It provides real-time updates and recommendations based on the predicted power output. Users can now monitor energy consumption, receive alerts, and make informed decisions about adjusting their energy usage patterns to optimize their energy consumption.

- Ensure an uninterrupted power supply to active buildings by offering personalized energy usage recommendations.

1.2 Advantages

Certainly, here are some advantages of our project:

- **Increased Use of Clean and Sustainable Energy Sources**

By promoting the use of solar panels to produce clean energy, our project contributes to the achievement of SDG 7 (Affordable and clean energy). The use of solar energy is reliable and sustainable, reducing the dependence on fossil fuels and their associated environmental impacts.

- **Improved Energy Efficiency**

The power forecast generated by our machine learning models creates a demand and response system for active buildings. This system provides recommendations to users on how to manage their energy load more efficiently, resulting in reduced energy consumption and lower energy bills.

- **Integration with Home Automation**

Our project includes integration with home automation, which allows users to turn on and off home appliances using the mobile application. This feature enhances convenience for users and reduces unnecessary energy consumption.

- **Real-Time Monitoring and Prediction of Solar Irradiance**

Our project's machine learning model is used to predict solar irradiance based on historical data. This feature allows for real-time monitoring and prediction of solar irradiance, which is crucial for efficient energy management.

- **Increased Affordability and Accessibility of Energy**

By providing safe and affordable energy for houses, our project contributes to SDG 11 (sustainable cities and communities). The use of solar energy as a sustainable and affordable energy source can be particularly beneficial for low-income households, reducing their energy bills and improving their quality of life.

Overall, our project has several advantages that contribute to the achievement of the SDGs and improve energy efficiency, accessibility, and sustainability.

Chapter 2 – Literature Review and Background

2.1 Background

The global energy challenges have posed significant obstacles in our daily lives and hindered our professional development compared to international competitors. To address these difficulties, the utilization of alternative energy resources for power generation has emerged as a viable solution, with Solar Energy being at the forefront. Initially, Solar Panels were primarily available for industrial applications, but over time, they have become more accessible for domestic use as well, at a higher cost. However, despite the widespread adoption of Solar Panels, many users lack awareness regarding their efficient utilization, resulting in inefficient and ineffective use of the generated solar energy and significant wastage.

To tackle this issue within the constraints of our country, our project provides a software product that optimizes the usage of Solar Panels. By leveraging machine learning models, the Application developed delivers precise energy utilization schedules that users with deployed Solar Panels can implement to achieve energy-saving strategies. The model estimates solar energy production during peak hours. Based on the generated energy, the app provides an optimal schedule, taking into account the user's behavioral patterns. This approach provides maximize reliance on the generated solar energy instead of the traditional grid, resulting in cost savings for the user. By addressing the lack of awareness and providing efficient energy utilization schedules, our project seeks to bridge the gap between solar energy generation and its effective utilization. The application empowers users to optimize their energy consumption patterns and reduce reliance on conventional energy sources, thereby promoting sustainability, cost savings, and a more environmentally friendly approach to energy usage.

2.2 Literature Review

In recent years, there has been a significant shift towards sustainable and renewable energy sources, with photovoltaic (PV) technology emerging as a promising solution for generating clean electricity. Due to the need to shift towards sustainable energy systems, considerable research has taken place worldwide in the domain of power forecasting and demand response systems. PV systems are being increasingly integrated into buildings, transforming them into active structures that actively generate and consume energy. The rise of PV-dominated active buildings not only offers a pathway towards achieving energy independence but also presents new challenges and opportunities for optimizing energy consumption patterns [21].

One of the key challenges associated with PV-dominated active buildings is the fluctuating nature of solar energy production. Solar power generation is heavily dependent on environmental factors such as sunlight availability, weather conditions, and seasonal variations. Consequently, the energy supply from PV systems can vary significantly throughout the day, week, and year [22]. This intermittency poses a unique challenge for effectively managing the energy demand in PV-dominated active buildings, as it requires balancing energy consumption with energy generation.

To address this challenge, the concept of Power forecasting and Demand Response (DR) systems has gained considerable attention in the field of smart grids and energy management. DR systems enable the adjustment of energy consumption in response to fluctuations in energy supply or grid conditions. By leveraging advanced control and communication technologies, DR systems empower building owners and occupants to actively participate in the energy market, optimize energy use, and contribute to grid stability. After integration of power forecasting into energy management, there were different horizons of power forecasting as shown in Figure 2.1 and all of them had different applications and fluctuations [23].

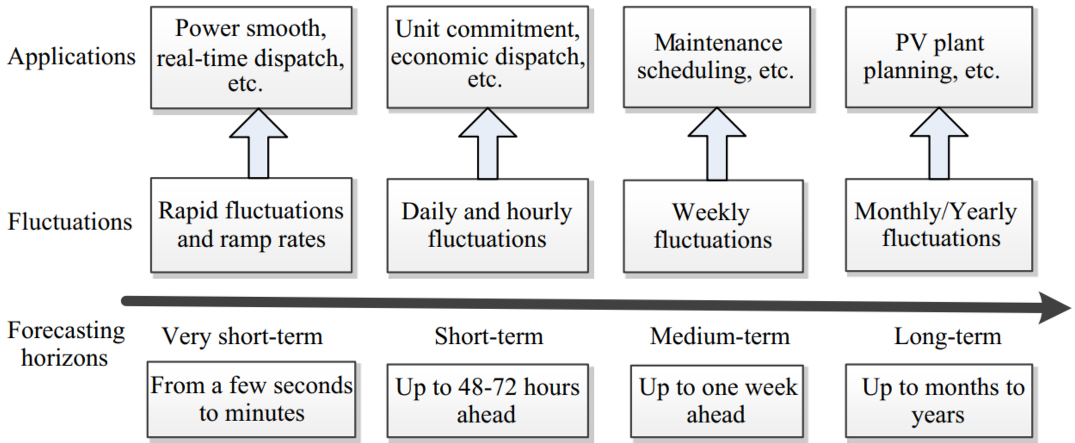

Figure 2.1 Forecasting horizons, their Fluctuations and Applications

Prominent research done by H. A. Kazem gained popularity due to its high level of accuracy. The methodology employed in this study has focused on the evaluation of FRNN and PCA. FRNN being a multilayered neural network model as shown in figure 2.2 was quite impressive in this research as it provided an accuracy of 72.4%. The researchers collected one year of experimental data at second and hourly intervals, resulting in a dataset of 50,400 daily samples. The GCPV system was assessed based on various parameters such as power and efficiency. The FRNN and PCA models were utilized to predict GCPV current and power, and their results were compared against measured values for validation. The findings demonstrated that the FRNN model outperformed the PCA model in simulating the experimental results, highlighting its potential for accurate long-term power forecasting in GCPV systems [1]. Additionally, a comprehensive comparison and evaluation of the measured and predicted data reinforced the technical and economic viability of GCPV systems, showcasing their suitability in the study area with notable yield, capacity factor, energy cost, and payback period values.

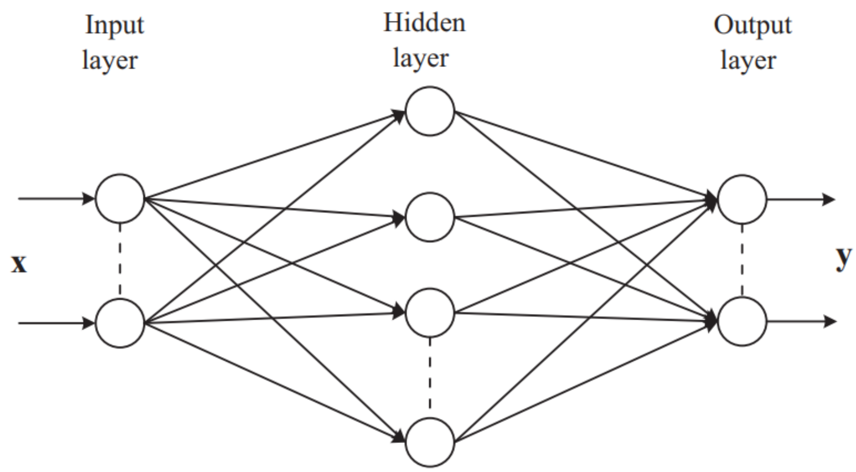

Figure 2.2 Typical Structure of a Feed-Forward Neural Network

The increasing relevance of forecasting in renewable energy sources applications, particularly in the context of PV power generation, has led to the exploration of various machine learning and hybrid techniques. In a research paper, the authors focus on the characteristics of artificial neural networks and their training approaches within a hybrid method for day-ahead PV output power forecasting. By analyzing different data set compositions and employing both traditional and newly defined performance indexes, the study evaluates the effectiveness of the training methods. The results, validated using one year of experimentally measured data, indicate that method C1 outperforms others when a substantial amount of historical data is available, while method A is more suitable for newly deployed PV plants with limited historical data. The findings also demonstrate the importance of ensembles composed of independent trials and highlight the significance of accumulating a minimum period of measurement data for accurate forecasting in newly deployed PV facilities [2]. Overall, this study provides insights into optimizing training data sets for effective PV power production forecasting.

A recent study done by H.Wang did some pioneering work on integrating Artificial Intelligence techniques to do solar power forecasting. As shown in Figure 2.3, they experimented with a range of machine learning models to find out their accuracy and use the most accurate model for forecasting. In the end, the research finalized down to four models which were the most accurate and made a hybrid model consisting of results from those models and that model provided them with overall a lower accuracy [44].

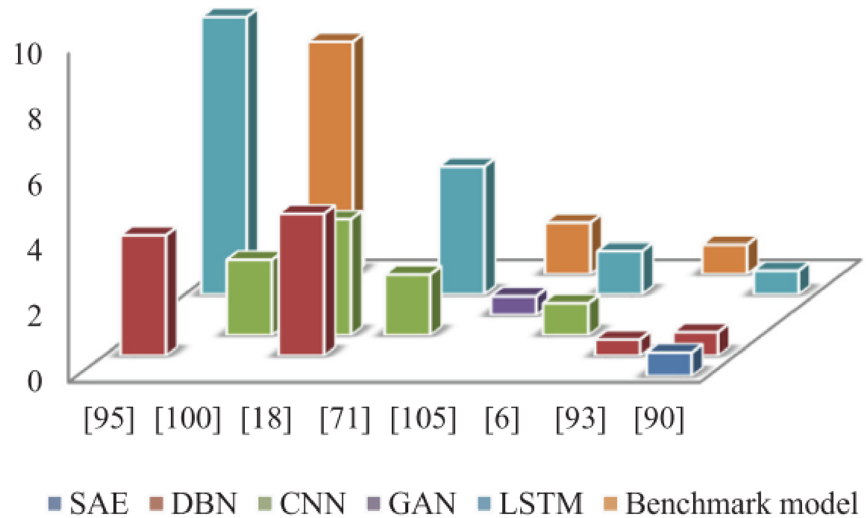

Figure 2.3 RMSE Comparison of Different Machine Learning Models

The demand for renewable energy has led to the adoption of photovoltaic (PV) panels as a clean energy source globally, including in Bangladesh. Accurate forecasting of PV panel output is crucial for efficient energy distribution and capacity planning. A study introduces the Prophet Model, a machine learning model designed for univariate time series forecasting, to predict the energy output of PV panels. Data was collected from an outdoor experimental setup in Bangladesh during the winter season. The proposed method successfully predicts the short circuit current of the PV panels one day in advance, with reliable results and a high coefficient of determination averaging at 0.9772 for one-day-ahead PV panel output energy forecasting [3].

Uncertainty in renewable energy integration, particularly in solar photovoltaic (PV) systems, due to its dependence on meteorological phenomena, presents a significant challenge. To address this issue, numerous forecasting tools have been developed, with recent emphasis on physics-informed machine learning (ML) methods. However, the lack of replicability and fair comparisons among these studies has led to misleading assessments. A paper conducts a comprehensive literature review and comparison of the most relevant ML methods. The authors propose a novel methodology that integrates a PV-performance model within ML models to forecast PV power several hours ahead with a 5-minute resolution. The methodology involves expanding the basic dataset, consisting of power production and meteorological measurements. This expansion allows the ML models to learn the physical interdependencies among different features, thereby potentially improving the accuracy compared to conventional methods. To further refine the model, a physics-informed feature selection approach is introduced to narrow down the search space for the best-performing model. The proposed methodology is validated using a case study of a PV array in Denmark, employing both the original and expanded datasets. The results consistently demonstrate that the ML models incorporating physics-informed features outperform other models in terms of accuracy. By leveraging physics-based information, the models can make more accurate predictions, while the feature selection process helps streamline the training and tuning stage [4]. Overall, this research contributes to the forecasting community by providing benchmark performance for ML and statistical methods, as well as offering a replicable and comparable framework for future studies in PV power forecasting.

Accurately predicting power output in solar photovoltaic (PV) facilities is essential for assessing their energy generation potential. However, the effectiveness of existing prediction models on new sites is not well understood due to variations in topography and meteorological conditions. Another research presents a monthly PV power forecasting model that utilizes data from 164 PV sites, including plant capacity, electricity trading data, weather conditions, and solar irradiation estimates, collected over a significant time period. The model employs a recurrent neural network (RNN) with long short-term memory (LSTM) to capture temporal patterns. The model's performance is evaluated using testing data from new plants. The results demonstrate the model's ability to achieve accurate predictions, with low error rates and good capture of temporal patterns [5]. This approach is valuable for planning officials in selecting suitable locations for PV plants in different areas, considering weather and terrain data.

Another paper presents a novel approach aimed at addressing the long-term forecasting challenge and reducing the uncertainty in PV forecasts. The proposed method involves a series of pre-and post-processing steps designed to optimize data preparation before inputting it into the forecasting model and refining the forecast output. Particularly, the method focuses on converting the non-stationary historical solar PV radiation data into a stationary format, enabling the utilization of a larger dataset for more accurate forecasting [6]. The performance of the proposed method is demonstrated through numerical simulations, highlighting its effectiveness in improving long-term PV forecasting.

Huang's research in 2019 outlined traditional demand response systems and their future as shown in Figure 2.4. He highlighted multiple benefits of installing a demand response system which included greater efficiency, reduced energy consumption and off-peak energy consumption [25]. This research included multiple types of energy forms to develop this demand response system.

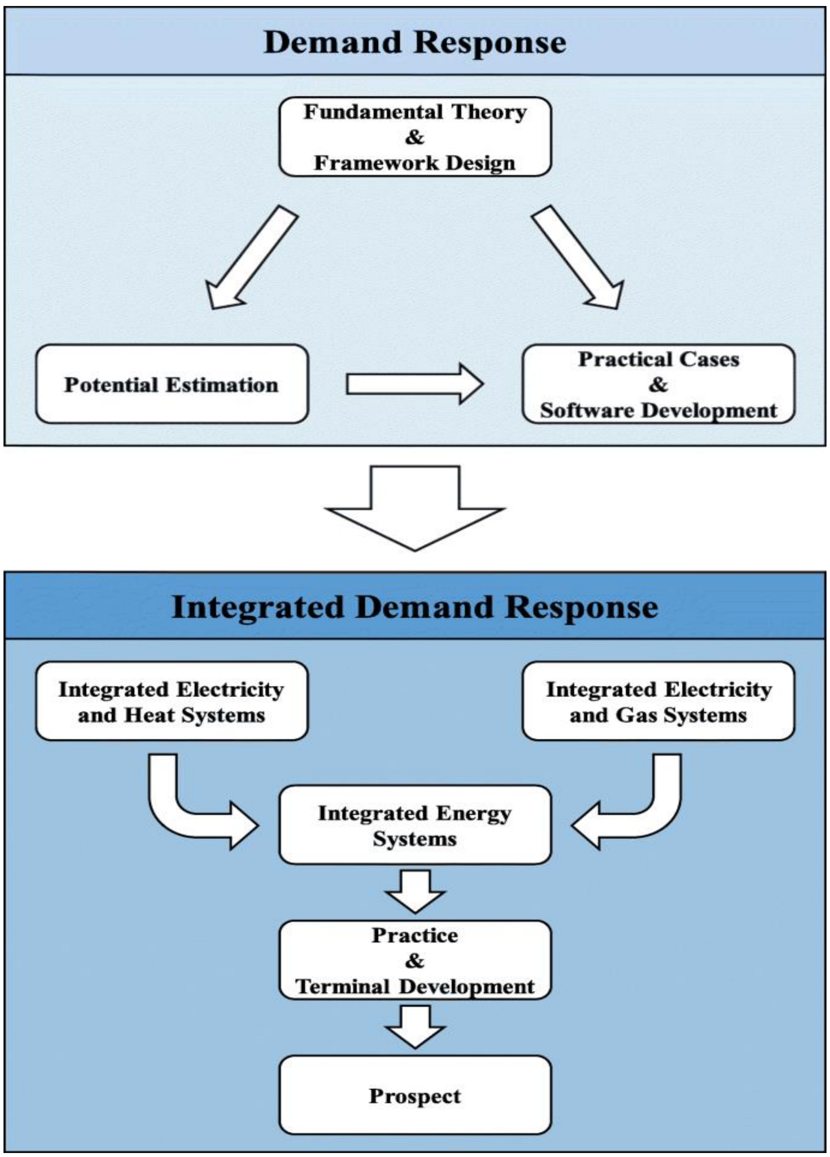

Figure 2.4 Layout of Demand Response System

Similarly, another research focusing on overview of demand response systems in the electricity market highlights its importance and benefits to the consumers as well as the market. Figure 2.5 can be seen below which demonstrates the advantages in a distributive manner to all the stakeholders of this system [26]. The effect of DR system on the electricity market prices can be seen in Figure 2.6 as per this research.

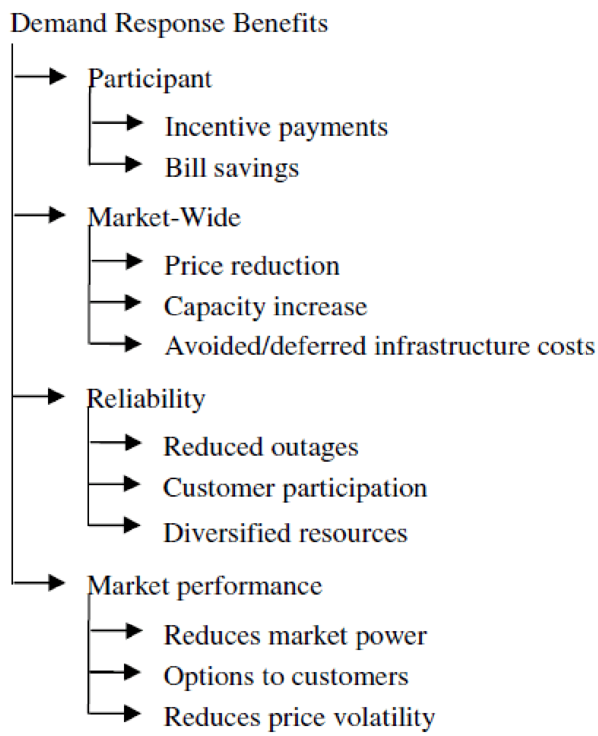

Figure 2.5 Benefits of Demand Response System

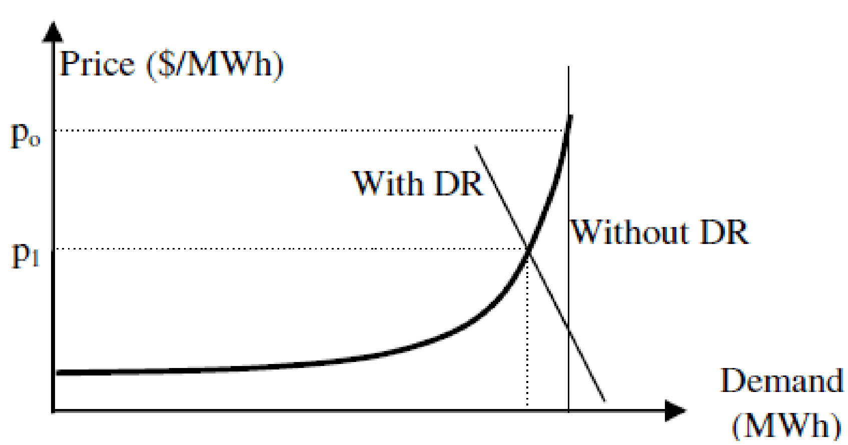

Figure 2.6 Effects of Demand Response System on Electricity Prices

Demand response systems face several challenges that can impact their effectiveness and widespread implementation. One key challenge is participant engagement. Encouraging active involvement from consumers and building owners can be difficult due to factors such as lack of awareness, understanding, or motivation. Educating and incentivizing participants becomes crucial in order to foster their engagement. Additionally, human behavioral patterns present a challenge [27]. Consumption habits and preferences can vary greatly, making it challenging to accurately forecast and respond to demand fluctuations. Understanding and predicting individual behaviors in response to incentives is complex and requires careful consideration.

Technological integration poses another significant challenge. Demand response systems often need to integrate diverse and sometimes outdated energy management systems. Ensuring interoperability, data compatibility, and seamless communication between devices, sensors, and control systems can be complex and time-consuming. Data privacy and security also come into play [28]. Demand response relies on collecting and analyzing sensitive energy consumption data. Protecting privacy and ensuring data security are essential for gaining consumer trust and preventing unauthorized access or misuse of data.

Infrastructure limitations can hinder the effective implementation of demand response. Existing grid infrastructure may have capacity, stability, or flexibility constraints that need to be addressed. Upgrading infrastructure to support bidirectional communication and accommodate distributed energy resources (DERs) is necessary but can be costly and require significant time and effort. Moreover, regulatory and policy barriers can impede progress. Outdated policies, complex regulations, or a lack of clear guidelines may not adequately support or incentivize demand response programs [29]. Aligning policies and market structures to facilitate demand response becomes crucial for its success.

Evaluation and measurement present their own set of challenges. Assessing the effectiveness of demand response programs requires accurate baseline establishment, quantifying energy savings, and determining the impact on grid stability and reliability. Robust evaluation methodologies and data analysis techniques are necessary to gauge the true benefits of demand response accurately. Lastly, stakeholder collaboration plays a critical role [30]. Collaboration among consumers, utilities, grid operators, and policymakers is necessary for effective implementation. Coordinating efforts, aligning interests, and fostering collaboration can be challenging due to diverse priorities, fragmented markets, and differing levels of awareness.

Addressing these challenges requires a comprehensive approach that combines technical advancements, behavioral interventions, regulatory reforms, and market-oriented solutions. Overcoming these hurdles can lead to wider adoption of demand response systems, enabling more efficient and sustainable energy management practices.
From the given literature review, we have selected the best research papers from the literature review and made a simplified table to show the advantages and disadvantages of the research papers.

No.	**Title, Year, Journal**	**Methodology**	**Pros**	**Cons**
1	Long-term power forecasting using FRNN and PCA models for calculating output parameters in solar photovoltaic generation,2022, Heliyon	ML Models: FRNN and PCA	FRNN is better at simulating the experimental results curve compared with PCA	Only two models have been compared No comparison of ml-based models and statistical models has been made.
2	Comparison of Training Approaches for Photovoltaic Forecasts by Means of Machine Learning, 2018, Applied Sciences	Physical Hybrid Artificial Neural Network	Higher accuracy and less error	Only day ahead which is of less benefit

3	Forecasting PV Panel Output Using Prophet Time Series Machine Learning Model, 2020, IEEE Xplore	Prophet Model	The prophet model is encouraging and reliable	Only one model has been studied. One-day ahead forecasting
4	Benchmarking physics-informed machine learning-based short-term PV-power forecasting tools, 2022, Energy Reports	RF, SVR, CNN, LSTM, CNN-LSTM, and Persistence Recurrent neural network	RF proved to be the best performer Casted more accurate predictions	Only benchmarking was done in this article No real application was tested for verification
5	Long short-term memory recurrent neural network for modeling temporal patterns in long-term power forecasting for solar PV facilities: A case study of South Korea, 2020, Journal of Cleaner Production	Recurrent Neural Network	Allow planning officials to evaluate suitable locations for PV plants in a wide area.	Power forecasting was just 1hr ahead which is too small to be of any importance in the long term
6	Long-term solar generation forecasting, 2016, IEEE Xplore	NARX Model	Long-term forecasting is done which is of greater importance	The long-term forecast has lesser accuracy

Table 2.1 Literature Review Table

Chapter 3 – Methodology

3.1 System Design

In this portion, we will describe in detail the design of the system.

3.1.1 Block Diagram of the System

The Block diagram showcases the operational process of demand response in PV-dominated active buildings, incorporating various interconnected components. The system begins by acquiring daily weather forecast data, which serves as crucial input. This weather data is then fed into a machine learning model, which leverages historical data and current weather information to predict key parameters such as energy demand and PV power output. The predicted data is subsequently transmitted to a mobile application, empowering users with real-time insights and control over their energy consumption.

On the other side of the system, the main power supply is composed of PV panels. To monitor and measure the electrical parameters, voltage, and current sensors are attached to the power supply. These sensors continuously collect data of the voltage and current levels, providing real-time information on energy consumption and PV power production. The collected data is then relayed to a microcontroller, which acts as an intermediary.

To facilitate seamless communication, a Wi-Fi module is employed to wirelessly transmit the data from the microcontroller to the mobile application. This integration enables users to access and manage the data remotely through the mobile application interface. Additionally, the mobile application facilitates load control by utilizing relays connected to the voltage and current sensors. Users can remotely activate or deactivate specific loads, allowing for efficient load management and responsive demand response strategies.

By combining accurate weather forecasting, machine learning prediction models, real-time data monitoring, and mobile application control, the system empowers users to optimize their energy consumption, adapt to dynamic energy conditions, and actively participate in demand response activities. This comprehensive approach fosters sustainability, efficiency, and effective energy management in PV-dominated active buildings.

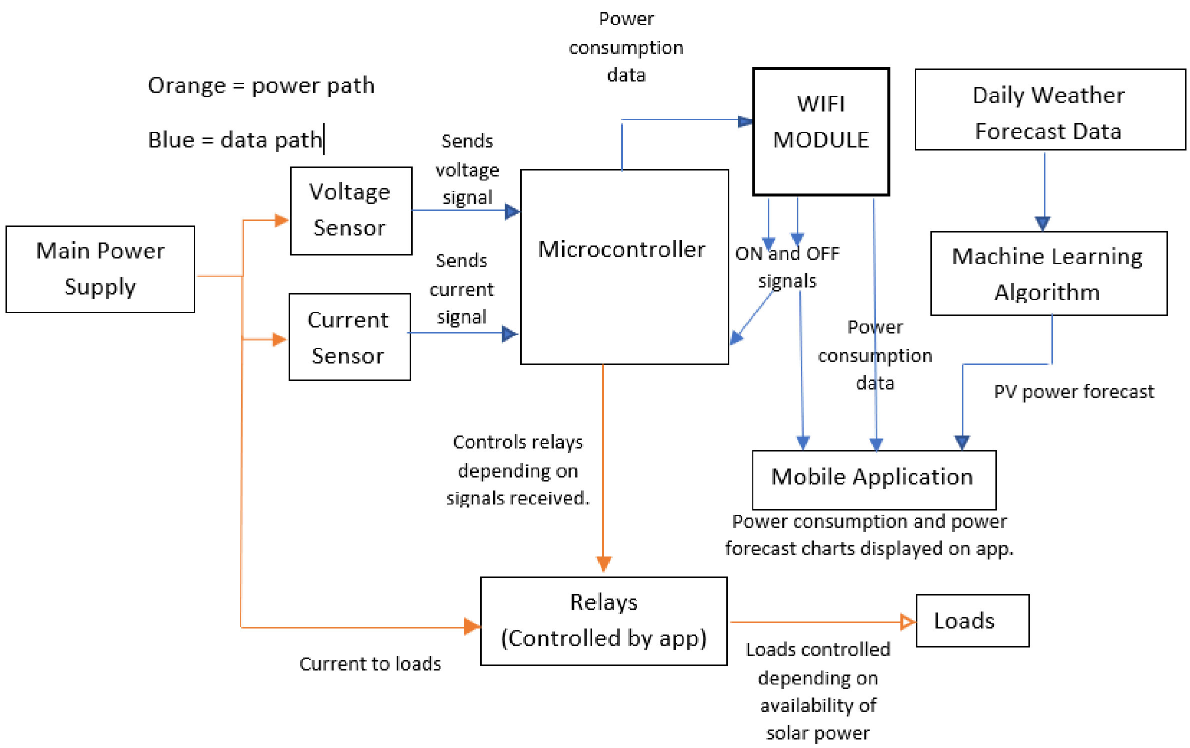

Figure 3.1 Systematic Block Diagram

3.2 Data Collection and Preparation

3.2.1 Data Collection

The solar irradiance data used in this study was obtained from the PK_Solar_Islamabad_NUST site, located at the National University of Science and Technology (NUST). The data collection was carried out using a Tier 1 station equipped with solar trackers and thermopile irradiation sensors. The site is situated at an elevation of 500 meters, with a latitude of 33.6419 (positive north, in decimal degrees) and a longitude of 72.9838 (positive east, in decimal degrees). The data was sourced from the website of the World Bank [9]. Regarding data preparation, the original data had a temporal resolution of 10 minutes. To facilitate analysis and modeling, several steps were taken.

3.2.2 Data Preparation

Firstly, the data was summarized by calculating the average value within each 10-minute interval. This allowed for a smoother representation of the solar irradiance data at a higher level of granularity. Subsequently, the data was converted into a one-day interval, providing a more suitable time scale for analysis and forecasting. Normalization was then applied to the data to ensure consistency and comparability. The values were normalized by dividing each data point by the largest value in the dataset. This normalization process brought all the data within the range of 0 to 1, enabling easier interpretation and manipulation during the subsequent modeling stages. By conducting these data collection and preparation steps, the solar irradiance data from the PK_Solar_Islamabad_NUST site was transformed into a suitable format for further analysis and utilization in the machine learning models.

3.2.3 Machine Learning Model Training

During the ML model training phase, various models were evaluated based on a comprehensive literature review, including Prophet, Linear Regression, Random Forest, and ARIMA. Among these models, Random Forest emerged as the most suitable choice for our project, which focuses on predicting solar irradiance using time series data.

Random Forest is an ideal choice for our specific application due to several advantages it offers. Firstly, it exhibits robustness and stability, making it highly effective in handling outliers and noisy data. It has the ability to handle diverse data patterns and is less prone to overfitting, ensuring reliable predictions even in the presence of irregularities. Additionally, Random Forest provides valuable insights into feature importance, enabling us to analyze the contribution of different variables in predicting solar irradiance. This understanding allows us to enhance the model's performance by focusing on the most influential features. Moreover, Random Forest utilizes ensemble learning, combining multiple decision trees to make predictions. This ensemble approach helps reduce bias and variance, resulting in more accurate and robust predictions. By leveraging the collective decision-making process, the model becomes adept at capturing intricate relationships and patterns within the data.

After selecting Random Forest as our ML model, we proceeded with the training phase using the collected and preprocessed data. To evaluate the model's performance, we considered several metrics, including Root Mean Square Error (RMSE), Mean, and R-squared (R2) score. The trained Random Forest model demonstrated exceptional performance in our project, achieving a low RMSE value, and indicating precise predictions of solar irradiance. Moreover, the high R2 score showcased the model's ability to explain a significant portion of the variance in the solar irradiance data, indicating a good fit. Overall, Random Forest exhibited robustness, stability, and the ability to handle complex data patterns. Its feature importance analysis provides valuable insights, and the ensemble learning approach improves prediction accuracy. With its strong performance, Random Forest proves to be a reliable and effective model for solar irradiance prediction in our project.

3.2.3.1 Code's Explanation

The code provided in annex A performs the following steps: It starts by importing the required libraries and reading the data from a CSV file into a Panda DataFrame. The data is then prepared by renaming columns and visualizing the time series. Next, lag features are created by shifting the 'ghi_pyr' column by different time intervals. The data frame is then cleaned by dropping any rows with missing values. The code imports the Random Forest Regressor model from scikit-learn and prepares the input and output variables for training. The data is split into training and testing sets based on a specific date. The Random Forest model is trained on the training data, and predictions are made on the testing data. Finally, the predictions are plotted against the actual values, and performance metrics such as RMSE (Root Mean Squared Error), mean, and R2 score are calculated and printed.

3.2.3.2 Formula Used for the Calculation of Energy

The formula that has been used to Calculate energy from the predicted solar irradiance is given below:

$$E = A \cdot r \cdot H \cdot PR \tag{1}$$

Where A is the Area of the panel, PR is the performance ratio, H is the annual average solar irradiation and r is the solar panel yield [7].
The energy has further been used to calculate the power using the following formula:

$$E = \frac{P}{T} \tag{2}$$

In the equation 2, P is the power, E is the energy and T is the time [8].

3.3 Home Automation Integration

In our project, home automation refers to the integration of technology and systems to automate and control various aspects of a home environment. It involves the use of sensors to monitor and manage functions such as energy consumption. The goal of home automation is to enhance convenience, comfort, efficiency, and security for homeowners.

Home automation using Wemos involves the integration of Wemos D1 Mini, an ESP8266-based microcontroller board, with various sensors, actuators, and smart devices to automate and control different aspects of a home environment. Here is a brief explanation of the working of home automation using Wemos:

- **Hardware Setup:** The Wemos D1 Mini board is connected to the home Wi-Fi network, and sensors and actuators are connected to its GPIO pins. These sensors include current and voltage sensors, while actuators include relays.

- **Blynk Integration:** The Blynk platform has been used to create a mobile application for controlling and monitoring the home automation system. Blynk provides an easy-to-use interface to create a customized user interface with buttons, sliders, and other widgets to interact with the Wemos board.

- **Communication:** The Wemos board communicates with the Blynk server through the Wi-Fi network. It sends data from the connected sensors to the server and receives commands from the mobile application to control the actuators.

- **Data Monitoring and Control:** The Wemos board continuously monitors the sensor data, such as current and voltage. It sends this data to the Blynk server, which is accessed by the user through the mobile application. The user now views real-time data from the sensors and control the connected actuators remotely.

- **Automation and Rules:** The home automation system can be configured to automate certain actions based on predefined rules. For example, if the generated power is less then the user is notified accordingly so the user can manage the load accordingly.

- **Alerts and Notifications:** The system has been set up to send alerts or notifications to the user's mobile device.

3.3.1 Explanation of Code

The code in annex B is an Arduino sketch that establishes a connection between the ESP8266-based Wemos D1 Mini board and the Blynk cloud platform using the Blynk library. The sketch sets up Wi-Fi credentials and defines virtual pins for controlling digital outputs. It handles Blynk events triggered by changes in the virtual pin states and performs corresponding actions to control the digital output pins. Additionally, it includes a timer event that sends the Arduino's uptime to a virtual pin every second. The setup function initializes the necessary configurations, and the loop function continuously runs Blynk and timer events. Overall, the code demonstrates the integration of the Wemos board with the Blynk platform for remote monitoring and control applications.

In order to validate the functionality and performance of the developed circuit, the Blynk platform was utilized as a testing tool. Blynk, a user-friendly mobile application, provided a convenient interface for monitoring and controlling the circuit remotely. This allowed for immediate verification of the circuit's operation and ensured that the readings were accurately captured. The Blynk application also offered control features, enabling the activation or deactivation of specific loads connected to relays. This functionality provided an additional means of assessing the responsiveness and effectiveness of the circuit's load control capabilities. The utilization of Blynk for circuit testing demonstrated the practical feasibility of the developed application and provided valuable insights for further refinement and optimization.

3.4 App Development

The mobile application developed for this project aims to provide real-time monitoring and control of power usage in households through an IoT network. Built using Flutter, an open-source mobile UI software development kit, the Android app offers a user-friendly interface to visualize data collected from multiple current and voltage sensors. This section elaborates on the key features and components of the mobile application which we have named as SolarSense.

3.4.1 App Development Environment:

The app was developed using Android Studio IDE and Flutter framework, which leverages Dart as the underlying programming language. This combination allowed for efficient native app development with a focus on smooth user experience and cross-platform compatibility.

3.4.2 Real-Time Data Display and WiFi-based Device Control:

The mobile application integrates with an IoT network comprising multiple sensors monitoring the power usage of household appliances. Real-time data, including sensor readings and weather forecasts, is retrieved from the Firebase Real-time Database, a cloud-hosted NoSQL database. The app provides an intuitive interface to display this data, allowing users to track their power consumption and make informed decisions.

The mobile application enables users to remotely control household appliances connected to the IoT network. By leveraging WiFi connectivity, users can turn devices on and off conveniently through the app. This feature enhances convenience, allowing users to manage their appliances from anywhere within the network range.

3.4.3 Energy Efficiency Recommendations and Alerts:

In addition to real-time power consumption information in form of percentages, the SolarSense offers users valuable recommendations and alerts to enhance energy efficiency. Through intelligent analysis of the collected data, users receive personalized suggestions on optimizing their energy usage and reducing dependence on the electric grid. This feature empowers users to make conscious choices and contributes to sustainability efforts.

3.4.4 User Interface and Navigation:

The SolarSense consists of five screens/pages, designed to provide a seamless user experience:

a. Splash/Loading Screen: Displays an initial screen while the app loads.

b. Home Screen: Features interactive widgets that guides users to other sections and provide an overview of the percentage of forecast PV power utilized.

c. Power Usage Page: Displays the percentage of forecast PV power utilized and real-time power consumption of each appliance, allowing users to monitor their energy usage patterns this page contains a master switch as well for all the appliances.

d. Devices Page: Enables users to control the on/off states of various household appliances connected to the IoT network, providing convenient device management.

e. Notifications Page: Shows users the latest app-generated recommendations and alerts aimed at maximizing energy efficiency and promoting sustainable practices.

The developed mobile application serves as a powerful tool for real-time monitoring and control of power usage in IoT-based active buildings. By leveraging Flutter and Dart, the app offers a user-friendly interface, enabling users to visualize real-time data, receive energy efficiency recommendations, and control devices remotely. The app's multi-screen architecture provides a comprehensive user experience, facilitating effective power management and contributing to a sustainable and energy-efficient lifestyle.
The code for the application has been provided in the link at the appendix section.

Chapter 4 - Hardware

4.1 IOT Network Schematic

The hardware setup of the circuit diagram comprises several components and connections that enable the monitoring and control of electrical loads. The circuit begins with a 5-volt power supply provided to the ESP32 microcontroller. To protect the circuit from excessive current, resistors are connected to the output pins of the microcontroller, limiting the current flow.

An optocoupler is utilized to connect the isolated circuits within the system. This component ensures that the high-voltage circuit on the right side of the diagram remains separate from the low-voltage circuit on the left side, maintaining safety and preventing any potential damage. Capacitors are also used to create a low pass filter to aid in noise reduction in the sensor reading.

The circuit includes a total of four current sensors, with three of them connected in parallel to each load. These current sensors measure the electrical current consumed by each individual load. Additionally, a voltage sensor is connected to measure the voltage across each appliance, providing valuable information on power consumption.

To control the loads, three relays are incorporated, each accompanied by a diode to prevent any reverse current flow. Furthermore, to protect the LEDs from burning out, a 1k resistor is attached to each LED. The circuit also incorporates multiple 50-ohm resistors to vary the load. With the help of resistors 3, 6 and 9 Watts.

A battery is integrated into the system, and a current sensor and voltage sensor are connected to it. These sensors measure the current flowing through, and the voltage supplied by the battery, respectively. This allows for monitoring the power provided by the battery to the loads.

On one side of the circuit, there is a provision for connecting a battery as the power source for the loads, while on the other side, a solar panel can be attached as an alternative power source. This setup enables flexibility in selecting the power supply based on availability and requirement.

Overall, the schematic diagram illustrates a well-designed circuit configuration that facilitates the monitoring, control, and efficient management of electrical loads through the integration of various components, including microcontrollers, sensors, relays, resistors, and power sources. The software used for making this schematic is Altium designer.

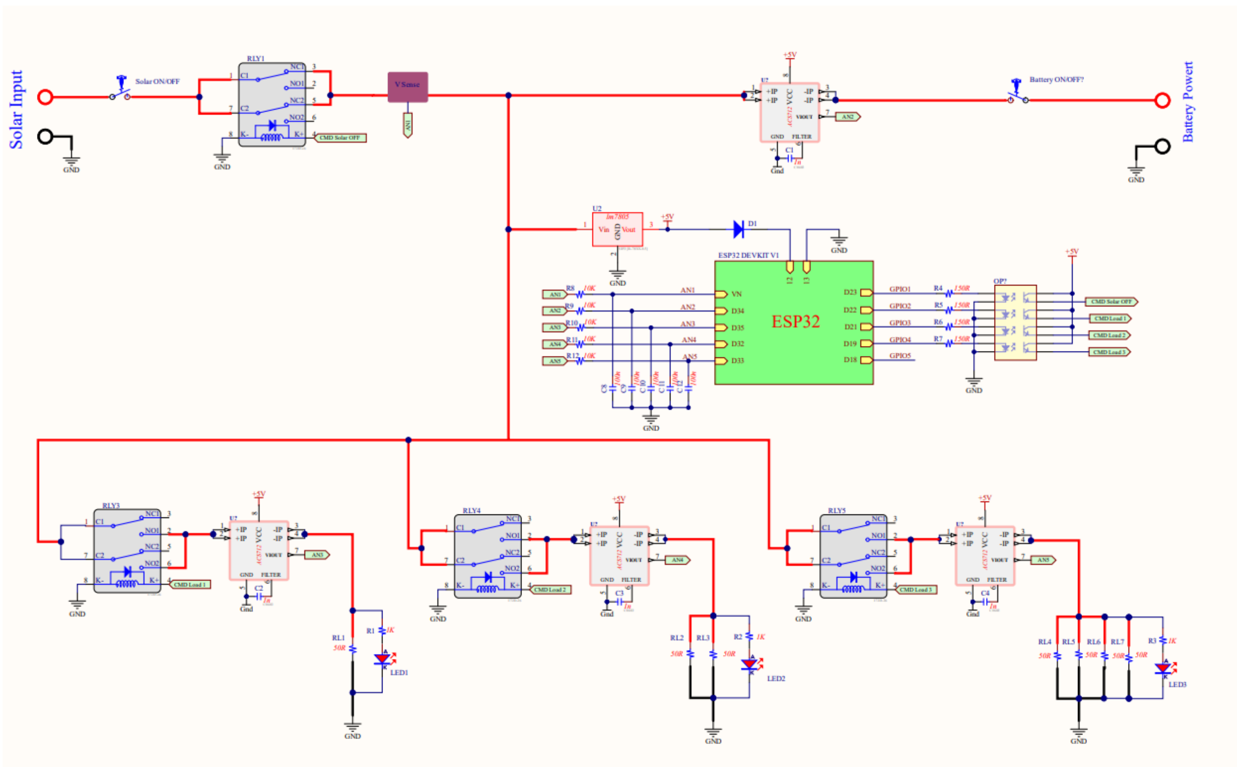

Figure 4.1 IoT Network Schematic

4.2 Components and Usage of Components in the Project

The following components have been used in our project.

- **Esp32 DOIT dev kit v1**

  The ESP32 microcontroller serves as the central control unit in our project, acquiring data from sensors, processing it, and communicating with the mobile application. It facilitates real-time monitoring, control, and management of power usage within the IoT network, enhancing energy efficiency and user convenience.

Figure 4.2 Esp32 DOIT dev kit v1 [9]

- **0-25 V DC Voltage Sensor (1x)**

The 0-25 V DC voltage sensor in our project enables accurate measurement and monitoring of voltage levels across various appliances(loads) and measures the voltage supplied by the battery.

- **ACS712 5A Current Sensor (4x)**

The ACS712 5A current sensor in our project is used to accurately measure and monitor the electrical current flowing through the circuit. It provides real-time data on current consumption, enabling precise power analysis and control of the connected loads.

Figure 4.3 ACS712 5A Current Sensor [10]

- **Diode (3x)**

The use of diodes in our project is to prevent the reverse flow of current in the circuit.

- **Resistors (1 kΩ (3x), 15 Ω (4x), and 50 Ω (6x))**

The purpose of a 1 kΩ resistor across the LED is to prevent it from burning out, the purpose of a 50 Ω resistor is to vary the load, and the purpose of a 15 Ω resistor is to prevent the circuit from damage.

- **TLP521-4(GB) Photocoupler (1x)**

The use of a photocoupler in our project is to separate the high voltage from the low voltage side of the circuit to avoid any damage.

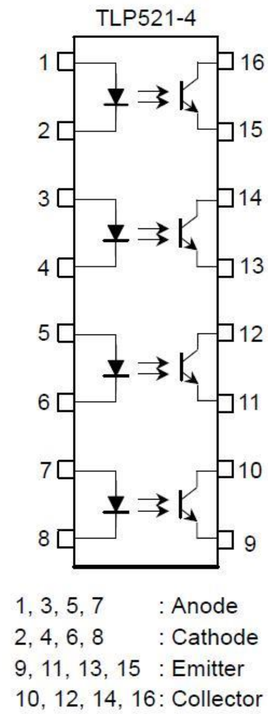

Figure 4.4 TLP521-4(GB) Photocoupler [11]

- **7805BT 1A Voltage Regulator (1x)**

The purpose of the 7805BT 1A voltage regulator in our project is to stabilize and regulate the incoming voltage to a constant 5V output. It ensures a consistent power supply for the connected components, protecting them from voltage fluctuations and providing reliable operation.

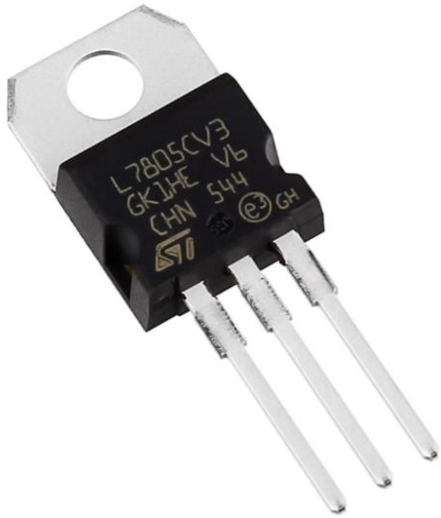

Figure 4.5 7805BT 1A Voltage Regulator [12]

- **LED Lights (3x)**

  The LEDs in our circuit serve the purpose of the load.

- **Transistor (1x)**

  The purpose of a transistor in our project is to switch electronic signals. It acts as a control element, allowing us to regulate the flow of current.

- **Vero Board (1x)**

  The Vero board in our project serves as a reliable and convenient platform for assembling and connecting electronic components, enabling the development of the circuitry.

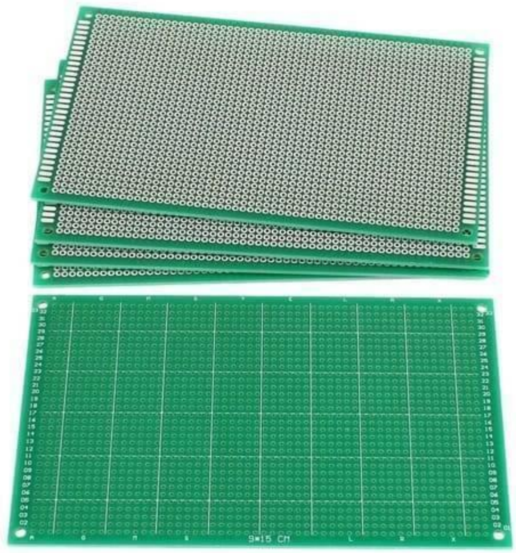

Figure 4.6 Vero Board [13]

- **20W Solar panel (1x)**

The solar panel used in our project consists of a single panel with a maximum power output of 20W. The panel has an area of 0.1736 square meters. The yield (r) of the panel is determined to be 22.5. The performance ratio (PR) of the panel is 0.75. The purpose of using this solar panel in our project is to capture solar energy and convert it into electrical power.

Figure 4.7 20W Solar Panel [14]

- **3.2 Ah Battery (1x)**

The purpose of the 12 V battery in our project is to store the power generated by the solar panel and then supply the power to the appliances(loads).

Figure 4.8 3.2Ah Battery [15]

4.3 Comparison of Actual and Predicted Power (Testing)

The performance of our solar panel system was tested by comparing the predicted data from our model with the actual data obtained from a real solar panel system. The tested solar panel system consisted of 19 panels, with each panel having a maximum power output of 540W. The total area covered by the 19 panels was 49.0827 square meters. The yield (r) of the system was determined to be 22.5. The irradiance (H) and performance ratio (PR) were taken into account during the testing process. The tilt angle of the panels was set at 20-25 degrees facing south, with a raised structure at a height of 12 ft. This comparison allowed us to assess the validity and accuracy of our predictive model in simulating the performance of the solar panel system.

The table presents the predicted and actual generation of solar energy in kilowatt-hours (KWh) over a specific time period. A comparison between the predicted and actual values allows for assessing the accuracy and performance of the prediction model. The data highlights variations in generation on different dates, providing insights into the effectiveness of the solar panel system in capturing and converting solar energy.

The table presents the results of the testing phase, comparing the predicted generation (in kWh) with the actual generation. The average mean square error (MSE) is calculated to be 104.7105144, indicating the overall deviation between the predicted and actual values. The root mean square error (RMSE) is 10.23281557 (or 10.23%), representing the average magnitude of the prediction errors. The accuracy of the model is measured at 89.77%, reflecting its ability to accurately forecast the solar energy generation.

Date	Predicted Generation (KWh)	Actual Generation (KWh)
23/04/2023	55.24809943	60
24/04/2023	55.10056572	58
25/04/2023	54.29937475	53
26/04/2023	47.82634759	44
27/04/2023	49.86424997	51
28/04/2023	43.34071143	47
29/04/2023	53.50358026	53
30/04/2023	51.23027649	52
01/05/2023	46.24693172	25
02/05/2023	48.96588291	45
03/05/2023	52.25747795	54
04/05/2023	46.62231026	29
05/05/2023	55.6600678	56
06/05/2023	47.79018151	41
07/05/2023	44.68317735	9
08/05/2023	57.10523946	58
09/05/2023	57.64230817	60
10/05/2023	59.63788286	62.5
11/05/2023	49.37899304	49
12/05/2023	52.86293766	54
13/05/2023	47.13618402	31
14/05/2023	52.93141585	55
15/05/2023	55.05378113	55
16/05/2023	45.05604353	25
17/05/2023	49.69721177	50
18/05/2023	54.5978481	55
19/05/2023	56.28922486	57
20/05/2023	47.58593482	37

Table 4.1 Predicted and Actual Generation in KWh

The bar chart visually represents the comparison between the actual power generated and the predicted power. It provides a clear illustration of the variances between the two values, allowing for a quick assessment of the model's accuracy. The height of each bar indicates the deviation between the actual and predicted power.

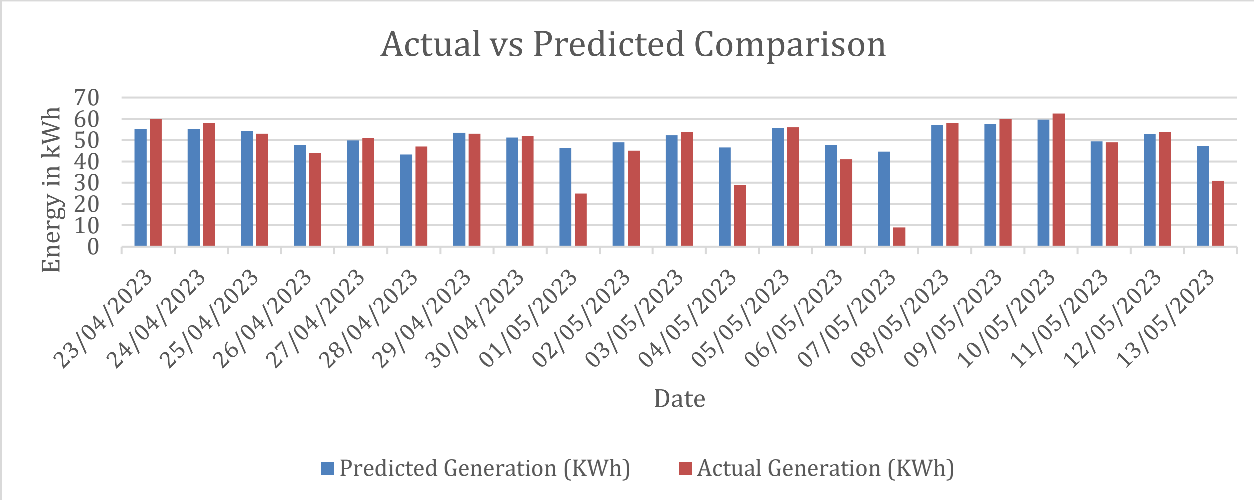

Figure 4.9 Actual vs Predicted Power Comparison

4.4 Working of IoT Network

The IoT network in the project plays a crucial role in connecting and facilitating the exchange of data between different components. The microcontroller, in our case it is ESP32, serves as the central hub for gathering sensor data and retrieving data from the Firebase database.

The microcontroller receives sensor data from various sources, such as current and voltage sensors attached to the battery and appliances (loads). These sensors measure the power consumption of each household appliance (load) and provide the corresponding data to the microcontroller.

Additionally, the microcontroller interacts with the Firebase database, which serves as a cloud-hosted database in this project. It retrieves data from the database. The microcontroller processes the collected data, performs necessary computations, and updates the values on the Firebase database.

On the other hand, the mobile application acts as the interface for users to access and visualize the collected data. It retrieves the sensor data from the Firebase database and displays it on the application's user interface. Users can view real-time power usage, forecasted power generation, and other relevant information related to energy consumption.

Furthermore, the mobile application also allows users to interact with the IoT network and control certain aspects remotely. Users can control signals via the mobile application, which can, in turn, adjust the operation of devices or appliances connected to the system.

Overall, the IoT network establishes a seamless connection between the microcontroller, the Firebase database, and the mobile application. It enables the real-time monitoring of power usage, data exchange between different components, and remote control of devices, providing users with valuable insights and control over their energy consumption.

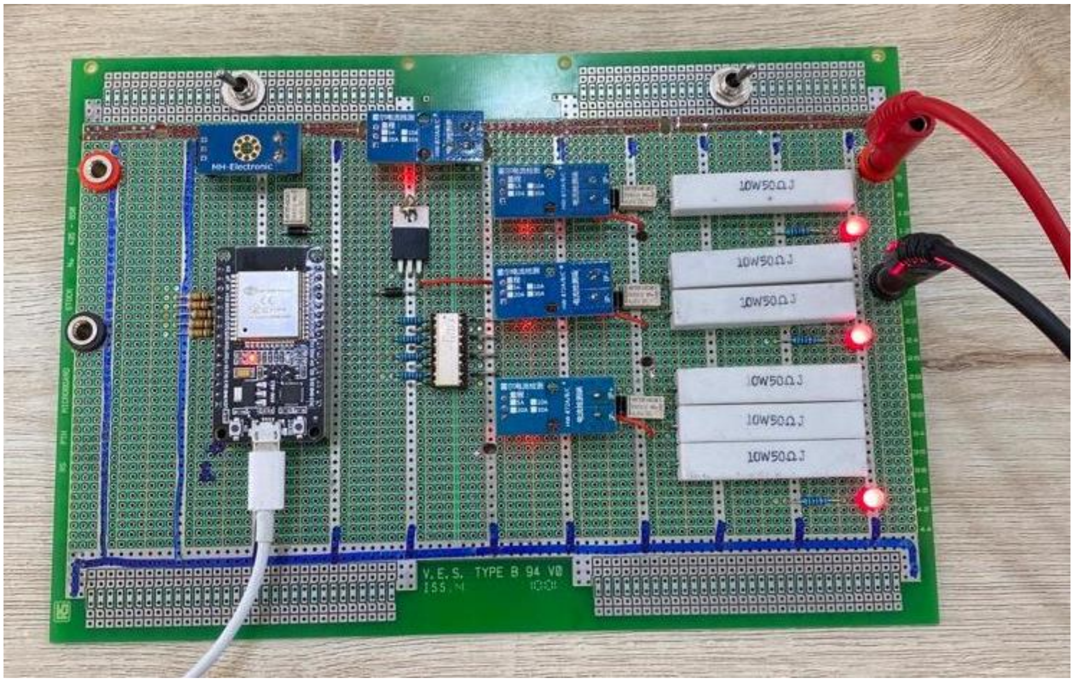

Figure 4.10 IoT Network Circuitry

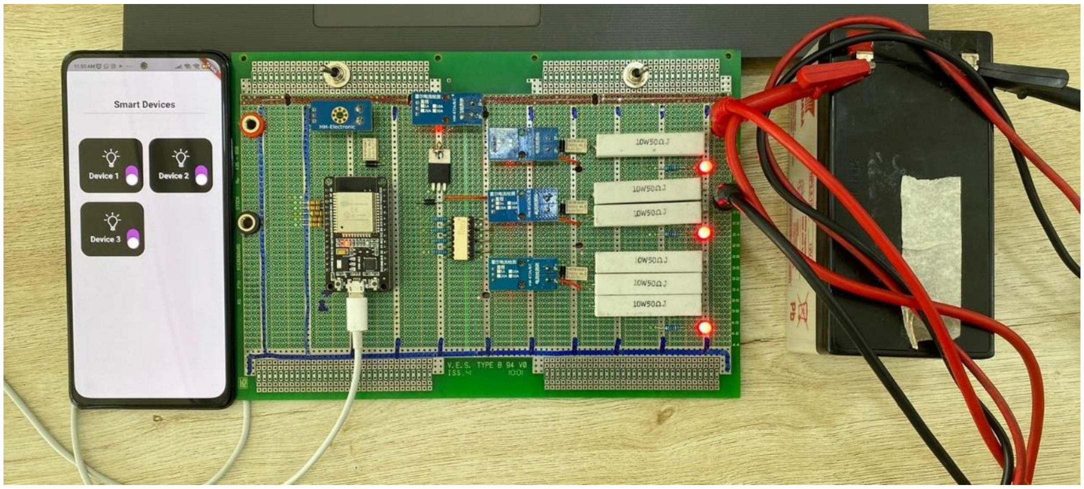

Figure 4.11 Hardware Setup

4.4.1 ESP32 Code Explanation

The explanation of the functions used in the code provided in the Annex C is given below.

- GetRelayStatus() function to get state of devices from database. States are changed when user turns device on/off from mobile app.
- GetSensorData() function gets analog values from sensors and converts to currents/amps using formula derived from sensor characteristics and calibration values obtained from testing performed on each sensor individually.
- GetEnergyConsumed() function to calculate total energy consumption from sensor values by accumulating power values over time.
- PublishSensorData() to publish all calculated energy data and sensor values to the online database so it is accessible by the mobile app.

Chapter 5 – Results and Discussion

5.1 Performance Evaluation of Machine Learning Model

In this portion, the performance of our machine learning model will be evaluated with the help of different parametrics.

5.1.1Graphical Results of ML Model

The graph displays the comparison between the predicted and actual irradiance levels over a specific time period. The graph consists of two distinct curves: one in blue and the other in orange. The blue curve represents the predicted irradiance values, which were estimated using the forecasting model. The orange curve represents the actual measured irradiance values obtained from on-site sensors.

Both curves exhibit variations and trends throughout the time period. The curves display fluctuations, rises, or falls in response to changes in environmental conditions. The shape of the curves suggests that the predicted irradiance generally aligns closely with the actual measured irradiance, indicating the effectiveness of the forecasting model.

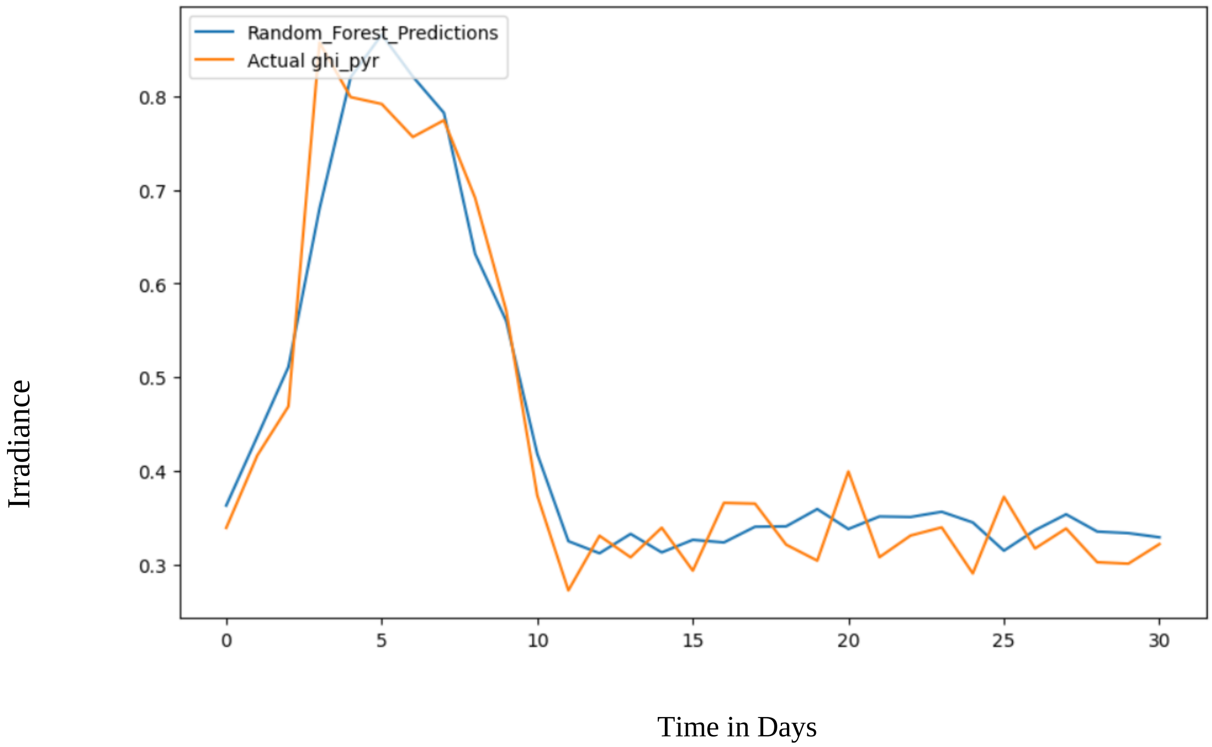

Figure 5.1 Graphical Results of ML Model

5.1.2 Accuracy of the Model

The accuracy of the model was evaluated using various performance metrics.

The R2 value, a measure of how well the model fits the data, was found to be 0.924. This indicates that approximately 92.4% of the variability in the irradiance levels can be explained by the model. A high R2 value signifies a strong correlation between the predicted and actual values, suggesting the model's effectiveness in capturing the underlying patterns in the data.

The mean absolute error (Mean) was calculated to be 0.4389. This represents the average difference between the predicted and actual irradiance values. A lower mean absolute error indicates a higher accuracy of the model's predictions, as it reflects the average deviation from the true values.

Furthermore, the root mean square error (RMSE) was computed and found to be 0.049. The RMSE provides a measure of the model's accuracy by quantifying the average magnitude of the errors between the predicted and actual values. A lower RMSE value indicates a smaller average prediction error and, hence, a higher level of accuracy.

These performance metrics demonstrate the effectiveness and accuracy of the model in predicting irradiance levels. The high R2 value, along with the low mean absolute error and RMSE, indicates a strong correlation between the predicted and actual values, showcasing the model's reliability in capturing and forecasting irradiance accurately.

5.2 Application Results

This portion will include the presentation usability of the mobile application.

5.2.1 Presentation and Usability of the Mobile Application's User Interface

Figure 5.2 illustrates the first page of the mobile interface, featuring buttons for controlling different devices. Users can easily toggle the devices on and off using these buttons. Moving on to the second page, depicted in Figure 5.3, users can access valuable information such as the power forecast for the day, battery status, and power meter readings. Continuing with the mobile application, the intuitive user interface allows for seamless device management and provides real-time updates on power consumption and battery status. This user-friendly design empowers users to make informed decisions and optimize their energy usage for enhanced efficiency and convenience.

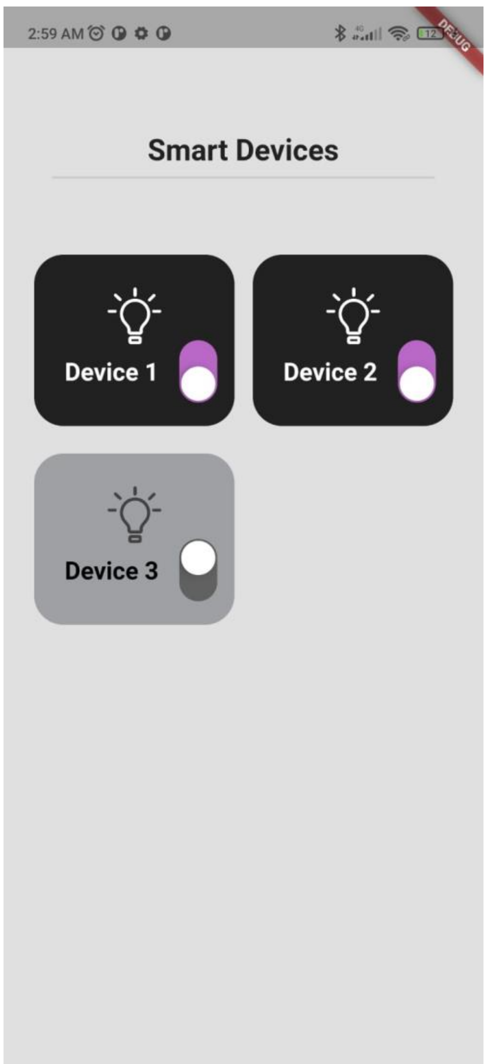

Figure 5.2 Interface Page 1

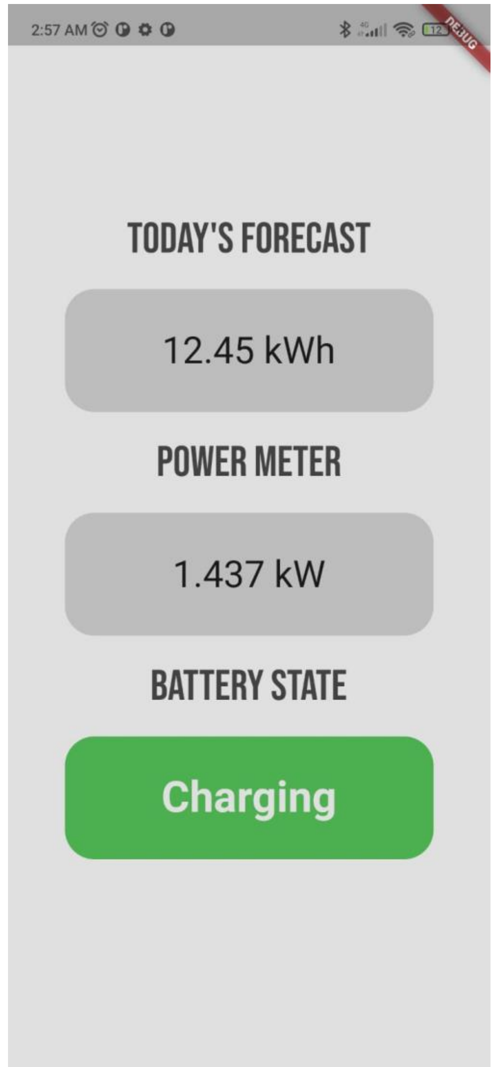

Figure 5.3 Interface Page 2

Moving on to the next features of the mobile application, the third page (Figure 5.4) provides a detailed breakdown of the percentage of forecasted power consumed, highlighting the individual power consumption of each device. This allows users to gain insights into their energy usage patterns and make informed decisions for efficient load management. Additionally, the fourth page (Figure 5.5) presents the total percentage of power consumed, enabling users to effectively manage their overall power load and make adjustments based on the valuable suggestions provided by the application. These features enhance user control and promote optimal energy utilization for improved sustainability.

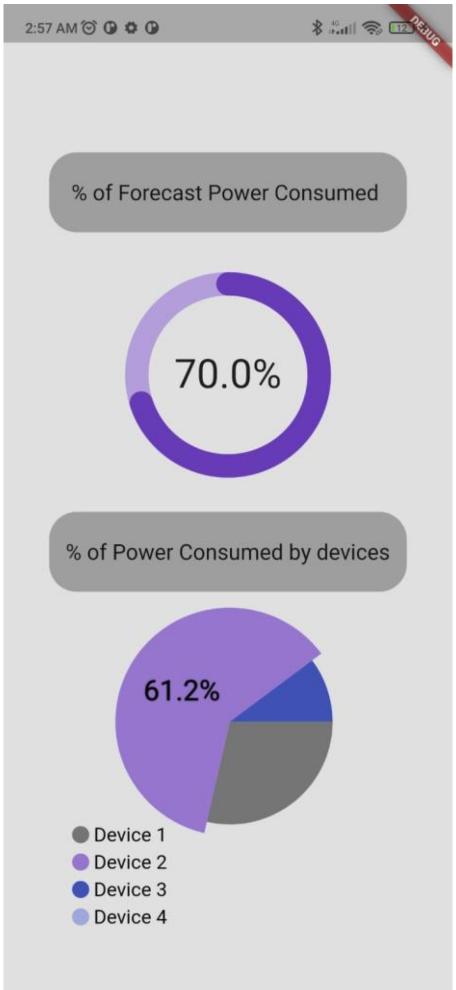

Figure 5.4 Interface Page 3

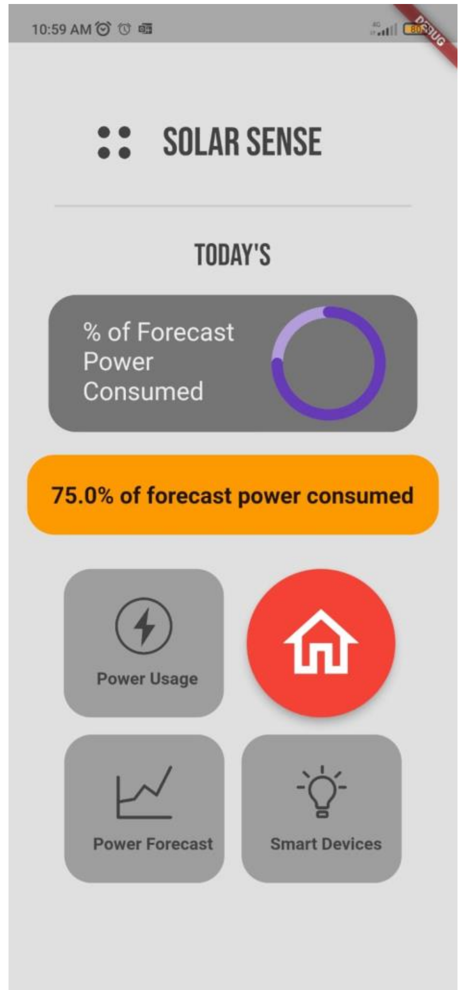

Figure 5.5 Interface Page 4

Chapter 6 – Conclusion and Future Work

6.1 Conclusion

This project marks a significant milestone in the realm of renewable energy and home automation, showcasing the potential for sustainable and efficient energy management in active buildings. By seamlessly integrating power forecasting capabilities, an IoT network, and a user-friendly mobile application named as SolarSense, we have created a comprehensive system that empowers users to optimize their energy consumption. Through accurate power forecasting using machine learning and weather data, users can make informed decisions about their energy usage. The integration of an IoT network with current and voltage sensors enables real-time monitoring of power usage, while the SolarSense provides a centralized control hub for remotely managing devices and receiving energy efficiency recommendations. Together, these advancements demonstrate the synergy between renewable energy, data analytics, and user-centric design, offering a promising pathway towards smarter, greener, and more sustainable homes.

6.2 Future Work

In order to further enhance the accuracy and effectiveness of the thesis research, several avenues for future work can be explored. One potential direction More features can be added to the app to enable automated load optimization and prioritization. Additionally, the inclusion of additional variables can significantly enhance the accuracy of the forecasting model. Consideration of factors like temperature, wind speed, and atmospheric conditions can provide valuable insights into their influence on solar energy generation. By incorporating these variables, the forecasting model can capture a more comprehensive understanding of the environmental factors impacting solar panel performance. Expanding the features of the accompanying application to enable monitoring of net metering can also be a fruitful area for future work. By developing the app to include functionalities such as energy balance, users can gain a more comprehensive understanding of their energy usage. This enhanced monitoring capability will facilitate better management of the net metering system and enable users to optimize their energy consumption patterns.

By pursuing these avenues of future work, the thesis can make significant contributions to the field of solar energy forecasting and provide practical solutions for enhancing the efficiency and accuracy of solar energy systems.

6.3 Limitations of the Project

The project, despite its achievements, is subject to several limitations that need to be acknowledged. Firstly, the installation of a Maximum Power Point Tracking (MPPT) system to extract the maximum power from the solar panel was not implemented.

Another limitation is the absence of a Solar Tracking System, which would enable the panel to rotate and align with the sun's position throughout the day. This feature can optimize the panel's exposure to sunlight and enhance its power generation capacity. Since the solar panel system from which the data was initially taken to train the model had a solar tracking system.

It is important to note that the project did not consider multiple factors that could influence the overall performance of the system. Factors such as wind speed, temperature variations, etc., were not taken into account during the analysis and forecasting processes. Incorporating these factors into future iterations of the project could lead to more accurate predictions and assessments that’s why recognizing these limitations provides insights into potential areas of improvement in our project.

References

[1] H. A. Kazem, J. H. Yousif, M. T. Chaichan, A. H. A. Al-Waeli, and K. Sopian, "Long-term power forecasting using FRNN and PCA models for calculating output parameters in solar photovoltaic generation," *Heliyon*, vol. 8, no. 1, p. e08803, Jan. 2022, doi: https://doi.org/10.1016/j.heliyon.2022.e08803.

[2] A. Dolara, F. Grimaccia, S. Leva, M. Mussetta, and E. Ogliari, "Comparison of Training Approaches for Photovoltaic Forecasts by Means of Machine Learning," *Applied Sciences*, vol. 8, no. 2, p. 228, Feb. 2018, doi: https://doi.org/10.3390/app8020228.

[3] Md. M. Hasan Shawon, S. Akter, Md. K. Islam, S. Ahmed, and Md. M. Rahman, "Forecasting PV Panel Output Using Prophet Time Series Machine Learning Model," *IEEE Xplore*, Nov. 01, 2020. https://ieeexplore.ieee.org/document/9293751

[4] D. V. Pombo, P. Bacher, C. Ziras, H. W. Bindner, S. V. Spataru, and P. E. Sørensen, "Benchmarking physics-informed machine learning-based short term PV-power forecasting tools," *Energy Reports*, vol. 8, pp. 6512–6520, Nov. 2022, doi: https://doi.org/10.1016/j.egyr.2022.05.006.

[5] Y. Jung, J. Jung, B. Kim, and S. Han, "Long short-term memory recurrent neural network for modeling temporal patterns in long-term power forecasting for solar PV facilities: Case study of South Korea," *Journal of Cleaner Production*, vol. 250, p. 119476, Mar. 2020, doi: https://doi.org/10.1016/j.jclepro.2019.119476.

[6] M. Alanazi, A. Alanazi, and A. Khodaei, "Long-term solar generation forecasting," *IEEE Xplore*, May 01, 2016. https://ieeexplore.ieee.org/document/7519883 (accessed Nov. 21, 2021).

[7] “Here is how you can calculate the annual solar energy output of a photovoltaic system,” *www.saurenergy.com*, Mar. 25, 2016. https://www.saurenergy.com/solar-energy-blog/here-is-how-you-can-calculate-the-annual-solar-energy-output-of-a-photovoltaic-system#:~:text=Globally%20a%20formula%20E%20%3D%20A
[8] “Power Formula,” *VEDANTU*. https://www.vedantu.com/formula/power-formula

[9] “Data Catalog,” datacatalog.worldbank.org. https://datacatalog.worldbank.org/search/dataset/0038550

[10] “Current Sensor ACS712-5A,” icmasteronline. https://icmasteronline.com/product/current-sensor-acs712-5a/ (accessed May 24, 2023).

[11] “TLP521-4 Original supply, US $ 1-10 , [Toshiba] Toshiba Semiconductor, TLP521-4 Supplier - SeekIC.com,” oedcl.en.seekic.com. http://oedcl.en.seekic.com/product/integrated_circuits_ics/TLP521_4.html (accessed May 24, 2023).

[12] “7805 Voltage Regulator IC 5V Regulator 7805 5V-1A,” Majju PK. https://www.majju.pk/product/12v-to-5v-lm7805-voltage-regulator-ic-5v-1a/ (accessed May 24, 2023).

[13] “9cmx15cm FR-4 Single Side Fiber Prototype Veroboard Stripboard PCB Circuit Board In Pakistan,” Digilog.pk. https://hallroad.org/products/9cmx15cm-single-side-veroboard-stripboard-pcb-circuit-board-in-pakistan (accessed May 24, 2023).

[14] “HI-SAFE SOLAR PANEL 20W-12V MONOCRYSTALLINE Solar Panel Price in India - Buy HI-SAFE SOLAR PANEL 20W-12V MONOCRYSTALLINE Solar Panel online at Flipkart.com,” Flipkart.com. https://www.flipkart.com/hi-safe-solar-panel-20w-12v-monocrystalline/p/itmcba17f76f668c (accessed May 24, 2023).

[15] “Yucel Y3.2-12 VRLA AGM 12 Volt 3.2Ah Battery | County Battery,” www.countybattery.co.uk. https://www.countybattery.co.uk/vrla-batteries/alarm/yucel-y3-2-12-vrla-agm-12-volt-3-2ah-battery/ (accessed May 24, 2023).

[16] A. Ruiz, “35 Latest Solar Power Statistics, Charts & Data [2023],” theroundup.org, Feb. 11, 2022. https://theroundup.org/solar-power-statistics/#:~:text=Around%204.4%25%20of%20total%20global

[17] B. Ceran, J. Jurasz, A. Mielcarek, and P. E. Campana, “PV systems integrated with commercial buildings for local and national peak load shaving in Poland,” Journal of Cleaner Production, vol. 322, p. 129076, Nov. 2021, doi: https://doi.org/10.1016/j.jclepro.2021.129076.

[18] K. Panicker, P. Anand, and A. George, “Assessment of Building Energy Performance Integrated with Solar PV: Towards a Net Zero Energy Residential Campus in India,” Energy and Buildings, p. 112736, Dec. 2022, doi: https://doi.org/10.1016/j.enbuild.2022.112736.

[19] S. Widergren, C. Marinovici, T. Berliner, and A. Graves, “Real-time pricing demand response in operations,” IEEE Xplore, Jul. 01, 2012. https://ieeexplore.ieee.org/abstract/document/6345195 (accessed May 24, 2023).

[20] S. J. Darby and E. McKenna, “Social implications of residential demand response in cool temperate climates,” Energy Policy, vol. 49, pp. 759–769, Oct. 2012, doi: https://doi.org/10.1016/j.enpol.2012.07.026.

[21] P. Bacher, H. Madsen, and H. A. Nielsen, “Online short-term solar power forecasting,” *Solar Energy*, vol. 83, no. 10, pp. 1772–1783, Oct. 2009, doi: https://doi.org/10.1016/j.solener.2009.05.016.

[22] R. Ahmed, V. Sreeram, Y. Mishra, and M. D. Arif, “A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization,” *Renewable and Sustainable Energy Reviews*, vol. 124, p. 109792, May 2020, doi:

https://doi.org/10.1016/j.rser.2020.109792.

[23] C. Wan, J. Zhao, Y. Song, Z. Xu, J. Lin, and Z. Hu, “Photovoltaic and solar power forecasting for smart grid energy management,” *CSEE Journal of Power and Energy Systems*, vol. 1, no. 4, pp. 38–46, Dec. 2015, doi: https://doi.org/10.17775/cseejpes.2015.00046.

[24] H. Wang *et al.*, “Taxonomy research of artificial intelligence for deterministic solar power forecasting,” *Energy Conversion and Management*, vol. 214, p. 112909, Jun. 2020, doi: https://doi.org/10.1016/j.enconman.2020.112909.

[25] W. Huang, N. Zhang, C. Kang, M. Li, and M. Huo, “From demand response to integrated demand response: review and prospect of research and application,” *Protection and Control of Modern Power Systems*, vol. 4, no. 1, May 2019, doi: https://doi.org/10.1186/s41601-019-0126-4.

[26] M. H. Albadi and E. F. El-Saadany, “Demand Response in Electricity Markets: An Overview,” *2007 IEEE Power Engineering Society General Meeting*, Jun. 2007, doi: https://doi.org/10.1109/pes.2007.385728.

[27] O′ConnellN., P. Pinson, H. Madsen, and O′MalleyM., “Benefits and challenges of electrical demand response: A critical review,” *Renewable and Sustainable Energy Reviews*, vol. 39, pp. 686–699, Nov. 2014, doi: https://doi.org/10.1016/j.rser.2014.07.098.

[28] S. Nolan and M. O’Malley, “Challenges and barriers to demand response deployment and evaluation,” *Applied Energy*, vol. 152, pp. 1–10, Aug. 2015, doi: https://doi.org/10.1016/j.apenergy.2015.04.083.

[29] J. Medina, N. Muller, and I. Roytelman, “Demand Response and Distribution Grid Operations: Opportunities and Challenges,” *IEEE Transactions on Smart Grid*, vol. 1, no. 2, pp. 193–198, Sep. 2010, doi: https://doi.org/10.1109/tsg.2010.2050156.

[30] P. Guo, V. O. K. Li, and J. C. K. Lam, “Smart demand response in China: Challenges and drivers,” *Energy Policy*, vol. 107, pp. 1–10, Aug. 2017, doi: https://doi.org/10.1016/j.enpol.2017.04.019.

Annexure

Annex A

Machine Learning Code

```
import pandas as pd
df=pd.DataFrame()
df = pd.read_csv('alterdata.csv',index_col='Date',parse_dates=True)
df.tail()
df.columns = ['ghi_pyr']
df.plot(figsize=(12,8))
df['ghi_pyr_LastMonth']=df['ghi_pyr'].shift(+1)
df['ghi_pyr_1Monthsback']=df['ghi_pyr'].shift(+2)
df['ghi_pyr_2Monthsback']=df['ghi_pyr'].shift(+3)
df['ghi_pyr_3Monthsback']=df['ghi_pyr'].shift(+4)
df['ghi_pyr_4Monthsback']=df['ghi_pyr'].shift(+5)
df['ghi_pyr_5Monthsback']=df['ghi_pyr'].shift(+6)
df['ghi_pyr_6Monthsback']=df['ghi_pyr'].shift(+7)
df['ghi_pyr_7Monthsback']=df['ghi_pyr'].shift(+8)
df['ghi_pyr_8Monthsback']=df['ghi_pyr'].shift(+9)
df['ghi_pyr_9Monthsback']=df['ghi_pyr'].shift(+10)
df['ghi_pyr_10Monthsback']=df['ghi_pyr'].shift(+11)
df
df=df.dropna()
df
from sklearn.ensemble import RandomForestRegressor
model=RandomForestRegressor(n_estimators=100,max_features=3, random_state=1)
import numpy as np
x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,y=df['ghi_pyr_LastMonth'],df['ghi_pyr_1Monthsback'],df['g
```

```
hi_pyr_2Monthsback'],df['ghi_pyr_3Monthsback'],df['ghi_pyr_3Monthsback'],df['ghi_pyr_4Mon
thsback'],df['ghi_pyr_5Monthsback'],df['ghi_pyr_6Monthsback'],df['ghi_pyr_7Monthsback'],df['
ghi_pyr_8Monthsback'],df['ghi_pyr_9Monthsback'],df['ghi_pyr']
x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,y=np.array(x1),np.array(x2),np.array(x3),np.array(x4),np.ar
ray(x5),np.array(x6),np.array(x7),np.array(x8),np.array(x9),np.array(x10),np.array(x11),np.array
(y)
x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,y=x1.reshape(-1,1),x2.reshape(-1,1),x3.reshape(-
1,1),x4.reshape(-1,1),x5.reshape(-1,1),x6.reshape(-1,1),x7.reshape(-1,1),x8.reshape(-
1,1),x9.reshape(-1,1),x10.reshape(-1,1),x11.reshape(-1,1),y.reshape(-1,1)
final_x=np.concatenate((x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11),axis=1)
print(final_x)
#X_train,X_test,y_train,y_test=final_x[:-1000],final_x[-200:],y[:-1000],y[-200:]
#import numpy as np

# assuming final_x is a numpy array of shape (n_samples, n_features)
dates = pd.to_datetime(df.index)  # convert index to datetime

# get the indices of the dates in the desired range
train_idx = np.where(dates <= '2022-11-30')[0]
test_idx = np.where(dates >= '2022-12-01')[0]

# slice the numpy array using the indices
X_train, X_test = final_x[train_idx], final_x[test_idx]
y_train, y_test = y[train_idx], y[test_idx]
pred=model.predict(X_test)
import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (12,8)
plt.plot(pred,label='Random_Forest_Predictions')
plt.plot(y_test,label='Actual ghi_pyr')
plt.legend(loc="upper left")
plt.show()
```

```
import matplotlib.pyplot as plt
import numpy as np

# Plot the predictions and actual values
plt.rcParams["figure.figsize"] = (12,8)
plt.plot(pred,label='Random_Forest_Predictions')
plt.plot(y_test,label='Actual ghi_pyr')
plt.legend(loc="upper left")
plt.show()

# Get the data points from the graph
x_points = np.arange(len(pred))
pred_points = pred.reshape(-1)
actual_points = y_test.reshape(-1)

# Create a table
table = []
for i in range(len(pred_points)):
    table.append([x_points[i], pred_points[i], actual_points[i]])

# Print the table
print("Data Table:")
print("{:<10} {:<20} {:<20}".format("Index", "Predicted Value", "Actual Value"))
for row in table:
    print("{:<10} {:<20} {:<20}".format(row[0], row[1], row[2]))
from sklearn.metrics import mean_squared_error
from math import sqrt
rmse_rf=sqrt(mean_squared_error(pred,y_test))
print('Mean Squared Error for Random Forest Model is:',rmse_rf)
# Calculate mean
mean_rf = np.mean(pred)
```

```
# Print mean results
print('Mean for Random Forest Model is:', mean_rf)
from sklearn.metrics import r2_score
# Calculate r2 score
r2_rf = r2_score(y_test, pred)
# Print r2 score results
print('R2 Score for Random Forest Model is:', r2_rf)
```

Annex B

Home Automation Code

```
#define BLYNK_TEMPLATE_ID "TMPL6Ri0OVpKe"
#define BLYNK_TEMPLATE_NAME "Wemos"
#define BLYNK_AUTH_TOKEN "KCsFfFYcaw_g5WpBgCdBLN5Qbd4Aulyj"
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#define USE_WEMOS_D1_MINI
char ssid[] = "New55";
char pass[] = "12345312";
BlynkTimer timer;
int pin0 = 5;
int pin1 = 4;
int pin2 = 0;
int pin3 = 2;

BLYNK_WRITE(V7)
{
   int value = param.asInt();
   if(value == 1)
     digitalWrite(pin0, HIGH);
   else
     digitalWrite(pin0, LOW);
}
BLYNK_WRITE(V8)
{
   int value = param.asInt();
   if(value == 1)
     digitalWrite(pin1, HIGH);
   else
     digitalWrite(pin1, LOW);
}
BLYNK_WRITE(V9)
{
   int value = param.asInt();
```

```
    if(value == 1)
      digitalWrite(pin2, HIGH);
    else
      digitalWrite(pin2, LOW);
}
BLYNK_WRITE(V6)
{
    int value = param.asInt();
    if(value == 1)
      digitalWrite(pin3, HIGH);
    else
      digitalWrite(pin3, LOW);
}

{
  // Change Web Link Button message to "Congratulations!"
  Blynk.setProperty(V3, "offImageUrl", "https://static-image.nyc3.cdn.digitaloceanspaces.com/general/fte/congratulations.png");
  Blynk.setProperty(V3, "onImageUrl",  "https://static-image.nyc3.cdn.digitaloceanspaces.com/general/fte/congratulations_pressed.png");
  Blynk.setProperty(V3, "url", "https://docs.blynk.io/en/getting-started/what-do-i-need-to-blynk/how-quickstart-device-was-made");
}

void myTimerEvent()
{
  Blynk.virtualWrite(V2, millis() / 1000);
}
void setup()
{
  Serial.begin(115200);
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);

  delay(100);
  pinMode(pin0, OUTPUT);
  pinMode(pin1, OUTPUT);
  pinMode(pin2, OUTPUT);
  pinMode(pin3, OUTPUT);
  digitalWrite(pin0, HIGH);
  digitalWrite(pin1, HIGH);
```

```
  digitalWrite(pin2, HIGH);
  digitalWrite(pin3, HIGH);
  timer.setInterval(1000L, myTimerEvent);
}
void loop()
{
  Blynk.run();
  timer.run();

}
```

Annex C

ESP32 Code

```
#include <WiFi.h> // esp32 library
#include <IOXhop_FirebaseESP32.h>
#include <ArduinoJson.h>
#include <StopWatch.h>

#define FIREBASE_HOST "fyp1-d368c-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "opIEHzpGfErpN1bGu79I1SYnKQXdtiaDAVj2evHF"
#define WIFI_SSID "Repeater_AP"
#define WIFI_PASSWORD "Repeater@1234" //password of wifi ssid
const int NumofDevices=3;
const int NumofSensors=4;
String RelayStatus[]={"OFF","OFF","OFF"};
int RelayPin0=23;
const int RelayPins[]={21,19,18};
int VoltagePin=A3;
int CurrentPins[]={A6,A7,A4,A5};
float voltage=0.0;
float totalpower=0.0;
float energy[NumofSensors]={0.0,0.0,0.0,0.0};
float previousenergy[]={0.0,0.0,0.0,0.0};
int homestate=0;
float current[NumofSensors]={0.0,0.0,0.0,0.0};
float power[NumofSensors]={0.0,0.0,0.0,0.0};
int calcurrent=2931; //calibration value
float battery_state=0.0;
float prevenergy=0.0;
```

```
unsigned long lastSample=0.0;
unsigned long timeFinishedSetup = 0;
StopWatch sw_secs(StopWatch::SECONDS);
float elapsedtime;

void WiFi_Setup(){
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD); //try to connect with wifi
  Serial.print("Connecting to ");
  Serial.print(WIFI_SSID);
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("Connected to ");
  Serial.println(WIFI_SSID);
  Serial.print("IP Address is : ");
  Serial.println(WiFi.localIP());
}

void Firebase_Setup(){
  Serial.println ("firebase is running");
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH); // connect to firebase
  Serial.println ("firebase connection completed");
}

void GetRelayStatus(){

  homestate=Firebase.getInt("Home_State/State");
  if(homestate==0){
     digitalWrite(RelayPins[0], LOW);
```

```
    digitalWrite(RelayPins[1], LOW);
    digitalWrite(RelayPins[2], LOW);
    Serial.println("Master switch: OFF");
  }
  else if(homestate==1){

    for(int i=0;i<NumofDevices;i++){
    String path="Device_States/Device"+ String(i+1);
     RelayStatus[i]=Firebase.getString(path);
        if (RelayStatus[i] == "ON") { // compare the input of led status received from firebase
        digitalWrite(RelayPins[i], HIGH); // turn led ON
      }
      else if (RelayStatus[i] == "OFF") { // compare the input of led status received from firebase
        digitalWrite(RelayPins[i], LOW); // turn led OFF
      }
      else {
         Serial.println("Wrong Credential! Please send ON/OFF");
      }

   };
  }

  }



void GetSensorData(){
  float analog_val=0.0;
  float temp_vol=0.0;
  float temp_curr=0.0;
  //getting voltage sensor data
```

```
for(int i=0; i<100;i++){
  analog_val = analogRead (VoltagePin);
  temp_vol= temp_vol+(analog_val)/231;
}
Serial.print("analog value voltage:");
Serial.println(analog_val);
voltage=(temp_vol/100);
Serial.println ("Voltage sensor value: "+ String(voltage) + " volts");
//getting current sensor's data
for(int i=0;i<NumofSensors;i++){
  float temp_curr=0.0;
 for(int j=0; j<100;j++){
  analog_val = analogRead (CurrentPins[i]);
  temp_curr = temp_curr+(analog_val-calcurrent)*(2);

 }
  current[i] = (temp_curr/100)-80;
  for (int k=1;k<NumofSensors;k++){
  if((current[k]<0)||(current[k]<7)){
	current[k]=0;
	}
   }
   if(current[0]<-4000){
	current[0]=0;
	}
  Serial.println ("Current sensor " + String(i) + " value: "+ String(current[i]) + " mA");
}
if(current[0]<0)
{battery_state=1.1;
}
else
```

```
 {battery_state=0.1;}
 }



 void GetEnergyConsumed(int time){
    Serial.println(String(power[0])+" Watts");
   for(int i=0; i<NumofSensors;i++){
    energy[i]=(current[i]*voltage*time)/1000;
    previousenergy[i]= previousenergy[i]+energy[i];
    power[i]=current[i]*voltage/1000;
    Serial.print("Energy "+String(i)+": ");
    Serial.println(previousenergy[i]);
    Serial.print("Power "+String(i)+": ");
    Serial.println(String(power[i])+" Watts");
     }
 }

void PublishSensorData(){
 for(int i=0;i<NumofSensors;i++){
   String currpath="/Sensor_Data/Current_Sensor"+String(i);
   String powerpath="/Power_Data/Device"+String(i);
   String energypath="/Energy_Data/Device"+String(i);
   Firebase.set (currpath,current[i]);
   GetRelayStatus();
   GetSensorData();
   elapsedtime=sw_secs.elapsed()-elapsedtime;
   GetEnergyConsumed(elapsedtime);
   elapsedtime=sw_secs.elapsed();
   Firebase.set(powerpath, power[i]);
   GetRelayStatus();
   GetSensorData();
```

```
    elapsedtime=sw_secs.elapsed()-elapsedtime;
    GetEnergyConsumed(elapsedtime);
    elapsedtime=sw_secs.elapsed();
    Firebase.set(energypath, previousenergy[i]);
    GetRelayStatus();
    GetSensorData();
    elapsedtime=sw_secs.elapsed()-elapsedtime;
    GetEnergyConsumed(elapsedtime);
    elapsedtime=sw_secs.elapsed();
    }
    Firebase.set("/Power_Data/Battery_State",battery_state);

}

void setup() {

  Serial.begin(9600);
  delay(1000);

  for(int i=0; i<NumofSensors;i++){
    pinMode(RelayPins[i], OUTPUT);
    pinMode(CurrentPins[i], INPUT);
  }
  pinMode (VoltagePin,INPUT);
  WiFi_Setup();

  Firebase_Setup();
  digitalWrite(RelayPins[2], HIGH);
  digitalWrite(RelayPins[1], HIGH);
  digitalWrite(RelayPins[0], HIGH);
  previousenergy[0]=Firebase.getInt("Energy_Data/Device0");
```

```
  Serial.println(previousenergy[0]);
  sw_secs.start();



}
void loop() {
   GetRelayStatus();
   Serial.println(RelayStatus[0]);
   Serial.println(RelayStatus[1]);
   Serial.println(RelayStatus[2]);
   Serial.println(RelayStatus[3]);
    GetSensorData();
    elapsedtime=sw_secs.elapsed()-elapsedtime;
    GetEnergyConsumed(elapsedtime);
    elapsedtime=sw_secs.elapsed();
    PublishSensorData();

}
```

Appendix

Drive Link for the project file of mobile application code.

1. https://drive.google.com/drive/folders/1-7hcNdOwTRU2JFdnn4BxpQbD1ymXUaAK?usp=sharing
2. https://docs.google.com/document/d/1-_wit7GwwsHt-AcRfQbThK2yaiR80Bwdnv6igTFd7tE/edit?usp=sharing