DEVELOPMENT OF IOT BASED SYSTEM FOR ENERGY MANAGEMENT SYSTEM

MUHAMMAD RIDZWAN BIN BAHSHARUDIN

BACHELOR OF MECHATRONICS ENGINEERING WITH HONOURS

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

DEVELOPMENT OF IOT BASED SYSTEM FOR ENERGY MANAGEMENT SYSTEM

MUHAMMAD RIDZWAN BIN BAHSHARUDIN

A report submitted
in partial fulfilment of the requirements for the degree of
Bachelor of Mechatronics Engineering with Honours

Faculty of Electrical Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

DECLARATION

I declare that this thesis entitled "DEVELOPMENT OF IOT BASED SYSTEM FOR ENERGY MANAGEMENT SYSTEM is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Signature :

Name : Muhammad Ridzwan Bin Bahsharudin

Date : 12 January 2024

APPROVAL

I hereby declare that I have checked this report entitled " DEVELOPMENT OF IOT BASED SYSTEM FOR ENERGY MANAGEMENT SYSTEM", and in my opinion, this thesis fulfils the partial requirement to be awarded the degree of Bachelor of Mechatronics Engineering with Honours

Signature	:	
Supervisor Name	:	AINAIN NUR BINTI HANAFI
Date	:	12 Januari 2024

DEDICATIONS

To my beloved parents,

Dr Bahsharudin and Dr Rohaidah Omar.

ACKNOWLEDGEMENTS

I have cooperated with many new people and academicians throughout the process of completing this project. They have contributed in countless different ways, and I owe them all my gratitude. Firstly, my sincere appreciation goes to the main project supervisor, Dr. Ainain Nur Binti Hanafi who has provided me with extensive professional guidance and taught me a great deal about scientific research and literature reviews. Also, special gratitude to the Universiti Teknikal Malaysia Melaka (UTeM) for providing adequate relevant research materials and works of literature. Without the guidance and support from these people, this project would not have materialized.

My greatest appreciation goes to my parents and family for their love and kind support and for being a source of unending inspiration. Finally, I am also indebted to my fellow friends with whom I have had the pleasure of working in pursuing this project. Their great assistance, views, and suggestions were indeed useful on various occasions.

ABSTRACT

Energy Management Systems (EMS) have become essential in optimizing energy consumption in the built environment, particularly in commercial buildings. This project presents the development and application of a sophisticated IoT-based EMS, aimed at enhancing energy efficiency in air conditioning and lighting systems. At its core, a rule-based algorithm is employed to improve decision-making regarding energy use intensity and timing. EMS integrates advanced IoT devices and sensors for continuous monitoring and intelligent control of energy usage, leading to substantial energy savings. The algorithm, designed to optimize energy consumption, takes into account parameters like power consumption, illuminance, and room temperature. Its effectiveness is evidenced by a comparative analysis based on a 24-hour monitoring experiment conducted in two phases. These experiments reveal the algorithm's significant impact on energy usage optimization. The report details the IoT infrastructure, the design and implementation of the Fuzzy Logic Rule-Based algorithm, and the data analytics methodologies used. The automated decision-making process of the system efficiently reduces overall consumption, enhances energy efficiency, and lowers operational costs in commercial settings. The result of this project highlights the successful integration of IoT with a Fuzzy Logic Rule-Based approach, significantly improving energy management. The system, characterized by its real-time monitoring and automation capabilities, demonstrated a remarkable advancement in managing energy consumption. Notably, the lighting system observed a 69.41% decrease in energy consumption, while the air conditioning system saw a 30.6% decrease. These results underscore the algorithm's precision in managing energy usage, emphasizing its contribution to sustainable energy practices. Future work will focus on developing a predictive model for energy consumption data, using the XGBoost framework for enhanced accuracy in forecasting energy needs. This advancement is crucial for effective energy management, leading to optimized energy distribution and utilization. The predictive model, as an extension of this project, marks a significant step towards intelligent, efficient, and proactive energy management, aligning with sustainable development goals.

ABSTRAK

Sistem Pengurusan Tenaga (SPT) telah menjadi penting dalam mengoptimumkan penggunaan tenaga dalam persekitaran binaan, terutamanya di bangunan komersial. Projek ini membentangkan pembangunan dan aplikasi SPT berasaskan IoT yang canggih, bertujuan untuk meningkatkan kecekapan tenaga dalam sistem penghawa dingin dan pencahayaan. Di terasnya, algoritma berdasarkan peraturan digunakan untuk memperbaiki pembuat keputusan mengenai intensiti penggunaan tenaga dan waktu. SPT mengintegrasikan peranti dan sensor IoT canggih untuk pemantauan berterusan dan kawalan pintar penggunaan tenaga, membawa kepada penjimatan tenaga yang ketara. Algoritma yang direka untuk mengoptimumkan penggunaan tenaga mengambil kira parameter seperti penggunaan kuasa, kecerahan, dan suhu bilik. Keberkesanannya dibuktikan melalui analisis perbandingan berdasarkan eksperimen pemantauan 24 jam yang dijalankan dalam dua fasa. Eksperimen ini mendedahkan kesan signifikan algoritma terhadap pengoptimuman penggunaan tenaga. Laporan ini menjelaskan infrastruktur IoT, reka bentuk dan pelaksanaan algoritma Berasaskan Peraturan Logik Kabur, serta metodologi analisis data yang digunakan. Proses pembuat keputusan automatik sistem ini berkesan mengurangkan penggunaan keseluruhan, meningkatkan kecekapan tenaga, dan menurunkan kos operasi dalam tetapan komersial. Hasil projek ini menonjolkan integrasi berjaya IoT dengan pendekatan Berasaskan Peraturan Logik Kabur, yang secara signifikan memperbaiki pengurusan tenaga. Sistem, yang dicirikan oleh kemampuan pemantauan masa nyata dan automatiknya, menunjukkan kemajuan yang luar biasa dalam mengurus penggunaan tenaga. Secara khusus, sistem pencahayaan mencatatkan penurunan penggunaan tenaga sebanyak 69.41%, manakala sistem penghawa dingin mencatatkan penurunan sebanyak 30.6%. Hasil ini menekankan ketepatan algoritma dalam mengurus penggunaan tenaga, menekankan sumbangannya kepada amalan tenaga yang mampan. Kerja masa depan akan memberi tumpuan kepada pengembangan model ramalan untuk data penggunaan tenaga, menggunakan rangka kerja XGBoost untuk ketepatan yang lebih baik dalam meramalkan keperluan tenaga. Kemajuan ini penting untuk pengurusan tenaga yang berkesan, membawa kepada pengedaran dan penggunaan tenaga yang dioptimumkan. Model ramalan, sebagai lanjutan projek ini, menandakan langkah penting ke arah pengurusan tenaga yang pintar, cekap, dan proaktif, selaras dengan matlamat pembangunan mampan.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PAGE

LIST OF SYMBOLS AND ABBREVIATIONS

AC	-	Air Conditioner
AI	-	Artificial Intelligence
ANN	-	Artificial Neural Network
AWS	-	Amazon Web Services
AOA	-	Arithmetic Optimization Algorithm
BLE	-	Bluetooth Low Energy
DSM	-	Demand-Side Management
ED	-	Economic Dispatch
EDO	-	Economic Dispatcher Optimizer
EMS	-	Energy Management System
FPGA	-	Filed-Programmable Gate Array
FLC	-	Fuzzy Logic Control
IEEE	-	Institute of Electrical and Electronics Engineers
IoT	-	Internet of Things
IR	-	Infrared
LED	-	Light Emitting Diode
LUX	-	Illuminance
MILP	-	Mix-Integer Linear Programming
HEMS	-	Home Energy Management System
ISEMS	-	Intelligent Smart Energy Management System
EM	-	Energy Management
PC	-	Polycarbonate
SED	-	Stochastic Economic Dispatch
TOU	-	Time-Of-Tarif
Wi-Fi	-	Wireless Fidelity
FLRB	-	Fuzzy Logic Rule-Based
UI	-	User Interface
DU	-	Digital Units
GBDT	-	Gradient Boosting Decision Tree

LIST OF APPENDICES

PAGE

CHAPTER 1

INTRODUCTION

1.1 Motivation

Our world is increasingly grappling with a severe energy crisis, a multifaceted problem predominantly fueled by our overreliance on non-renewable resources such as coal, oil, and natural gas. According to National Geographic [1], [2], non-renewable energy sources such as coal, natural gas, oil, and nuclear energy supply about 80 percent of the world's energy. They provide electricity, heat, and transportation while feeding the processes that make a huge range of products, from steel to plastics.

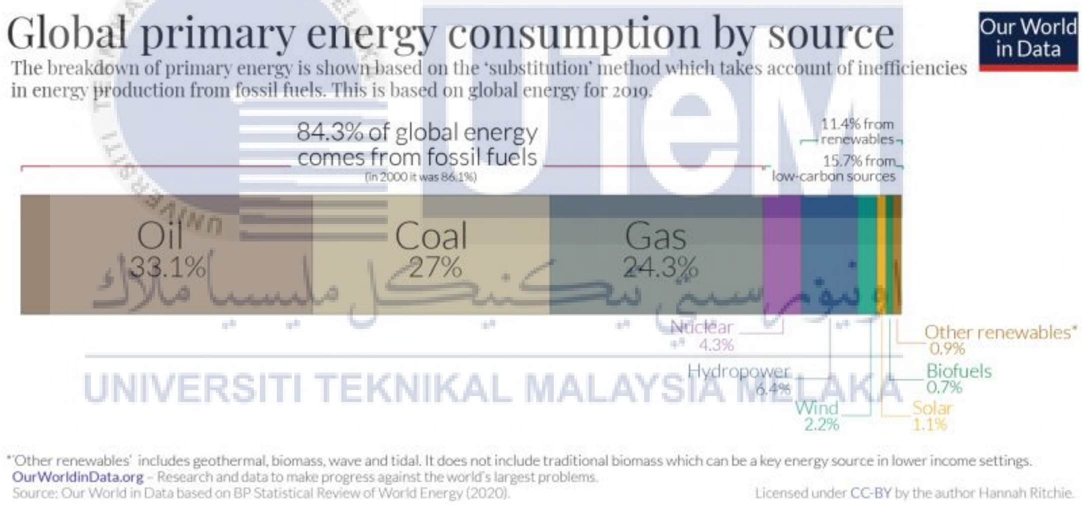


Figure 1.1 Global Energy Consumption by Source [3]

Figure 1.1 illustrates that, as of 2019, a staggering 84.3% of global energy is derived from fossil fuels, emphasizing the immense dependence on this resource to meet global energy demands. Some countries may also be showing signs of dependence on nonrenewable energy sources.

For example, in Korea, there are 60 coal-fired power plants that generate 40% of the nation's electricity [4]. In Malaysia, electricity generation is heavily dependent on coal resources. 66% of the electricity generated in the Peninsular was from imported coal fuel [5]. These resources, which have powered our modern civilization for centuries, are used extensively in electricity generation. Traditional power plants burn these fossil fuels to produce steam, which turns turbines to generate electricity [6]. Despite their widespread use and convenience, these energy sources are rapidly depleting due to excessive global consumption, and their continued usage exacerbates environmental issues like air pollution and climate change.

The generation of electricity from non-renewable energy sources is also characterized by inefficiencies and energy loss, with a significant portion of the energy from burned fossil fuels lost as waste heat. Additionally, these fuels' extraction, transportation, and burning result in substantial greenhouse gas emissions, which contribute to global warming [7].

The dwindling supply of non-renewable resources has escalated prices and market volatility. Oil, gas, and coal prices are skyrocketing due to Russia's invasion of Ukraine and the COVID-19 pandemic outbreak [8]. Even Malaysia, which imports coal, was impacted by it as commodity prices rose, resulting in a 45% increase in electricity production [8]. This disrupts economies and places a disproportionate burden on less economically developed countries and poorer societal segments, who devote a larger proportion of their income to energy costs.

In this scenario, the role of energy management systems (EMS) in buildings can be one of the solutions. By efficiently managing energy consumption, these systems offer a promising and sustainable solution to the energy crisis.

In this context, an energy management system (EMS) presents a promising solution to this crisis. An EMS is a tool that helps electric utility grid operators keep track of, regulate, and improve the performance of the generation and transmission systems [9]. These systems encompass various aspects of energy efficiency, including demand management, load balancing, peak load reduction, and overall optimization of energy usage. By providing real-time information about energy use, an EMS enables consumers and companies to make more informed decisions about their energy consumption, often leading to significant reductions in energy use [10].

1.2 Problem Statement

In the contemporary world, energy management is a pressing concern, and the efficient utilization of energy resources is paramount. Despite technological advances, many residential, commercial, and industrial establishments still struggle with the ineffective and inefficient use of energy. This issue is compounded by the fact that conventional energy management systems are manual, inflexible, and lack real-time monitoring, optimization, and control capabilities. This results in high energy costs, wastage of energy, and an unnecessary increase in the carbon footprint.

The lack of automation in the current systems also causes a lack of predictive analysis, which could preemptively identify potential areas of energy wastage or faults in the system. Due to the current system's latency, it is challenging for energy managers to get accurate, real-time data, make rapid decisions, and promptly execute necessary actions. Additionally, clear visibility and granular control over the diverse energy consumption points across a facility or multiple sites are missing.

Further, traditional energy management systems are not equipped with the ability to adapt to the ever-changing energy landscape, such as fluctuating energy prices and varying demands. This inability makes it difficult for organizations to optimize energy usage, resulting in economic losses and increased environmental impact.

Therefore, this project aims to develop an Internet of Things (IoT) based energy management system that addresses these challenges. The project seeks to integrate automation, real-time monitoring, control capabilities, predictive analytics, and adaptability into energy management to enhance efficiency, reduce costs, and minimize environmental impact. This innovative solution could provide energy managers with the tool they need to make faster and better-informed decisions, promote energy conservation, and improve the sustainability of their operations.

1.3 Objectives

- **Development and Integration of a Real-Time Monitoring and Automation System**: To create an IoT-based system that enables real-time monitoring, reporting, and automated control of energy consumption across various points in the facility or facilities. By enhancing visibility into energy usage patterns and automatically adjusting energy usage based on real-time data, this system will efficiently identify and rectify areas of waste or inefficiency, optimizing the use of energy resources.
- **Implementation of Rule-Based Techniques Algorithm**: To design a Rule-Based Techniques Algorithm within an Energy Management System (EMS). This algorithm will serve as the core decision-making component of the EMS, enabling it to optimize energy usage while maintaining user comfort.
- **Analysis the Efficiency of Algorithm Output in Energy Management System**: To assess the performance of the developed EMS algorithm by simulating its operation in a controlled Python environment. This involves using the data captured over a 24-hour period, including room vacancy, light intensity, temperature, and energy consumption, as inputs for the simulation. The goal is to compare the simulated energy consumption patterns, influenced by the EMS algorithm, against the baseline data collected without the EMS intervention..

1.4 Scope and Limitations

Limitations of the Project:

- **Technology Infrastructure**: The successful implementation of an IoT-based energy management system will rely on the existing technology infrastructure, such as internet connectivity, of the facility or facilities in question. Areas with poor connectivity may limit the system's effectiveness.
- **Security Concerns**: While the project will strive to ensure the system's security, IoT devices can be vulnerable to cyber-attacks. Despite the best efforts, there might still be a risk of security breaches.
- **Resource Availability**: The project's success will depend on the availability of resources, including hardware, software, reviewed papers, and skilled personnel, for the implementation of the IoT-based system.

The scope of this project focuses on implementing an energy management system that will concentrate on the use, optimization, and management of air conditioners, and lighting, possibly employing Internet of Things (IoT) technologies.

1. Air Conditioners:
    - Monitoring and analysis of power consumption patterns.
    - Automating temperature controls based on occupancy and weather conditions.
2. Room Lighting:
    - Monitoring power consumption of room lighting systems.
    - Automation of lighting based on occupancy and natural light availability.
    - Optimization of energy consumption through LED or other energy-efficient lighting technologies.

CHAPTER 2

LITERATURE REVIEW

2.1 Background Theory

The rising global energy demand and climate change concerns forced the development of sustainable and efficient energy management systems. Internet of Things (IoT) technology integration with these systems provides opportunities for optimizing energy consumption, reducing energy costs, and mitigating environmental impact. This section of the literature review provides an overview of the theoretical foundation for the development of Internet of Things-based energy management systems.

IoT refers to the interconnection of uniquely identifiable devices, sensors, and actuators, which enables automatic data collection, transmission, and analysis. The Internet of Things (IoT) has become an indispensable aspect of modern life, with implementations in numerous industries, including healthcare, transportation, and agriculture [11]. IoT utilizes an internet connection that provides connectivity between physical devices [12]. Key aspects of Internet of Things technology pertinent to energy management systems include:

a. Connectivity and Communication Protocols: IoT devices communicate with each other using a variety of communication protocols, including ESP8266, Wi-Fi, Bluetooth, and Raspberry PI.
b. Sensor Technology: a. Sensor Technology: Internet of Things devices with integrated sensors are used to collect real-time data on energy consumption, temperature, and other relevant parameters.
c. Data Processing and Analytics: IoT systems frequently employ cloud-based platforms to process and analyze collected data, facilitating the extraction of insightful insights and patterns.

Energy management systems (EMS) are centralized platforms that monitor, regulate, and optimize the production, distribution, and consumption of energy resources [13]. EMS seeks to reduce energy costs, reduce environmental impacts, and maintain energy reliability. The elements of energy management systems consist of the following:

a. Energy Monitoring: Real-time monitoring of energy consumption patterns that enable the identification of inefficiencies and opportunities for optimization.
b. Demand Response: Dynamic adjustment of energy consumption in response to fluctuations in energy supply and demand, which ensures grid stability and lowers energy costs.
c. Energy Forecasting: Predicting prospective energy consumption patterns based on historical data and external factors, such as weather conditions and occupancy schedules.

The integration of IoT technology has been studied in various industries, including supply chain and logistics [14], construction [11], college campus networks [15], IoT device sharing [16], and smart agriculture [17]. The integration of IoT technology into energy management systems offers several benefits:

a. Enhanced Data Collection: IoT-enabled devices provide real-time, granular data on energy consumption, enabling more precise energy monitoring and optimization.
b. Improved Decision-Making: The data collected from IoT devices can be processed and analyzed to provide actionable insights, allowing for enhanced energy management decision-making.
c. Increased Automation: IoT-based energy management systems can automate various energy management tasks, such as load scheduling and demand response, reducing the need for manual intervention.
d. Scalability and Flexibility: IoT technology can be readily scaled to accommodate larger energy management systems or adapted to specific requirements, thereby enabling customized energy management solutions.

While IoT-based energy management systems show great promise, several challenges, and research opportunities remain:

a. Interoperability: Ensuring seamless communication between IoT devices and energy management systems, regardless of the manufacturer or communication protocol[18].
b. Data Security and Privacy: Addressing concerns related to the collection, transmission, and storage of sensitive energy consumption data[19].
c. Energy Efficiency of IoT Devices: Developing low-power IoT devices that do not significantly contribute to energy consumption[20].
d. Integration of Renewable Energy Resources: Incorporating renewable energy resources, such as solar and wind power, into IoT-based energy management systems for a more sustainable energy future.

The development of IoT-based energy management systems is a viable method for reducing rising energy consumption while also addressing environmental issues. It is feasible to optimize energy use, save expenses, and limit environmental consequences by combining the capabilities of IoT technology with energy management systems.

2.2 Existing Energy Management System Overview

Several existing developments of home energy management systems proposed and created by previous researchers will be studied in this section of the literature review. Their features, methodology, and outcomes are compared to get a greater understanding of the present art of the state. The existing EMS frameworks, varying across diverse platforms and applications, utilize sophisticated technologies to monitor, control, and optimize the consumption and utilization of energy. There are many abbreviated names for Energy Management Systems, such as Home Energy Management System (HEMS or HEM), Intelligent Smart Energy Management System (ISEMS), Energy management (EM), Smart Energy Management System (SEMS) and many more. All the abbreviations can be considered a subset of a broader Energy Management System (EMS) concept, as all of them share the common objectives of optimizing energy consumption and reducing cost.

[12] discusses the integration of IOT with voice recognition applications for the Wireless Automation System. This implementation is a component of the EMS that allows voice commands to control the ON and OFF states of appliances, unlike the author's automated light fixtures. The authors also incorporated a footstep counter in which sensors count each person who passes through the front door. This saves system processing time.

[21]–[23] use smart sockets/plugs to capture energy consumption. In [21], the authors focus on the EMS design for buildings and define the EMS as HEM, also known as home energy management. The HEM system can be mounted wirelessly in any residential structure, and the authors mentioned how this system can track how much energy is used in every room it is installed in. The system consists of programmable air conditioner remote controls, smart sockets for monitoring appliance power use, and nodes for monitoring room temperature. These are all referred to as modules. All of these modules communicate with one another and collect data from sensors embedded into sockets, etc. via the Zigbee communication protocol. The proposed Smart-Rule Based HEM algorithm will then receive the energy consumption data and use it to plan the appliance energy use appropriately, hence decreasing the energy consumption. Based on the authors' proposed experimental methodology, one

potential strength is that the modules connected to HEM can monitor and control power consumption over 24 hours and can implement another 24-hour period under the same conditions without using HEM. The recording of both experiments reveals identical outcomes. The only disadvantage is that the methodology used in this setup has only been verified in a single household. Other households have varying energy consumption, and the results collected by the authors are not generalizable, as the usage patterns or appliance types of other households may vary. Additionally, the authors do not state the detailed cost-benefit analysis of the proposed system which could be useful for decision-makers considering the implementation of such a system.

In [22], the EMS system incorporates smart sockets for energy monitoring and control, cloud infrastructure for data processing and analysis for intelligent energy management. Mist hubs are also proposed to address complexities and resource demands in data processing. This approach enables energy monitoring, management, and efficiency in smart homes. Authors in [23] create IoT-based methodologies, smart plugs, wireless gateways, time-series analysis, and energy monitoring to develop a smart plug-load energy conservation and management system. The system captures energy consumption data through smart plugs, transfers it to a central database via a wireless gateway, and enables remote switching of smart plugs. Time-series analysis is applied to identify consumer behaviour and forecast total energy consumption. The system promotes energy conservation by making users aware of their energy consumption and provides data-driven approaches for minimizing energy usage.

The authors in [24] presented the Intelligent Smart Energy Management Systems (ISEMS) where this system handles the energy demand in a smart grid environment. The system proposed tested several models to predict the forecasting of the energy demand such as ANN (Artificial Neural Network), PSO-ANN (ANN Based Particle Swarm Optimization, SVR (Support Vector Regression), PSO-SVR (PSO Based Support Vector Regression and ENS (Ensemble Methods). Results show that PSO Based Support Vector Regression has better performance than other prediction models in terms of accuracy. ISEMS also uses the IOT to monitor the energy consumption of the user. The methodology of this paper has a potential strength where the prediction models make accurate forecasting of energy availability while the integration with the IOT makes the monitoring at the user end much easier.

In [25], the authors show that with the integration of IOT and the EMS, the system is able to optimize the operation of unbalanced three-phase AC microgrids. The system uses a stochastic economic dispatch optimizer (EDO), a database, a web-based graphical user interface (GUI), and an application programming interface (API). All these 5 modules show promising results. Based on the results, the PV energy significantly cut down the operating costs of the microgrids. With high PV generation, renewable energy resources (RESs) are able to sustain the conventional demand and also able to charge up the battery energy storage system (BESS) when an islanded operation happens. Furthermore, the advantage of the proposed IOT-based energy management system (EMS) for microgrids is the ability of the optimization of the distributed energy resources within a microgrid while adhering to grid constraints. However, the potential weakness or limitations of this methodology is its reliance on accurate data and forecasts for local demand and renewable generation, as well as the dependency on the availability of IoT technology.

The authors in [26] aims are similar to those in [21] where both integrate the EMS within residential buildings to achieve energy efficiency improvements. However, the authors improve their design by integrating it with renewable energy. The authors also use a rule-based strategy and AI-Based (Particle Swarm Optimization) optimization algorithm to reduce the energy consumption cost while maximizing the end user self-consumption. The proposed algorithm is validated through simulations in the MATLAB environment. With the AI-Based algorithm that is powering the system, the HEMS (related to the residential building) potential saving using the self-consumption is about 30% compared to users without the HEMS. The environmental effects also gain positive benefits as the CO_2 emission reduced by 30% when the HEMS is used in the building. The electricity used in residential buildings may be generated from fossil fuel-based power plants, and the emissions associated with electricity production are indirectly attributed to the building's CO_2 footprint.

Authors in [27] utilized STM32 microcontrollers and IoT technology to create an energy management system. The system incorporates automation to efficiently manage energy usage. The data from various sensors are monitored and analyzed using AWS IoT Analytics, enabling users to gain insights and control appliances remotely.

On the other hand, [28] employed IoT-based technologies, wireless communication, IoT-ready power converters, smart meters, smart sensors, and actuators to develop an energy management system. The integration of these technologies facilitated the implementation of a complete EMS that enables efficient control and management of energy generation, storage, and consumption. The FIWARE IoT Platform was utilized for device integration, data management, energy management algorithms, and data visualization.

In [29], the author utilized smart home technology, fuzzy logic for decision-making, IoT for data transmission, and machine-to-machine communication in the development of the energy management system. The system integrates a Battery Management System (BMS) and a Load Management System (LMS) to make intelligent decisions regarding the connection of loads to the grid or battery based on parameters such as load type, battery status, and grid availability. The implementation of fuzzy logic algorithms and the hardware model evaluation demonstrated significant improvements in energy consumption with the proposed Smart Home Energy Management System (SHEMS).

[30] works use the IoT technology, specifically Raspberry Pi, Node-RED, and NodeMCU modules, along with the Arithmetic Optimization Algorithm (AOA), to develop a multi-objective scheduling system for energy management in smart homes. The system optimizes the scheduling of home appliances to reduce electricity costs, decrease the peak-to-average ratio (PAR), and increase user comfort (UC). Real-time pricing and critical peak pricing signals are considered as energy tariffs. The integration of renewable energy sources (RES) is also explored.

The authors [31] implemented IoT-based devices and linear model algorithms to create an energy management system for residential buildings. The system accurately categorizes electricity consumption and utilization based on consumer behaviour data. Different linear model algorithms are applied for energy consumption analysis and intelligent power control. The results demonstrate improved energy forecasting efficiency using predictive models, with the RSME performance increasing by 35% in the lead time approach. The methodology focuses on using data-driven techniques and predictive modelling to optimize energy management in residential buildings.

[32] uses IoT principles and machine learning technologies to develop a self-learning home management system (SHMS) for efficient energy management in smart homes. The SHMS integrates a home energy management system, a demand-side management system, and a supply-side management system. Computational and machine learning techniques, that is Rule-Based Classifiers Technique are employed to enhance the system's capabilities, such as price forecasting, price clustering, and power alerts. Real-time power consumption data from a Singapore smart home is utilized to validate the system's performance and demonstrate its ability to customize the model for different environments compared to traditional smart home models. The SHMS is implemented on a Multi-Agent System (MAS) platform, enabling intelligent decision-making and adaptive operations.

The authors in [33] developed an energy management system that provides real-time monitoring of electricity use on home appliances using Internet of Things (IoT) technologies. The system combines Node-Red for data processing and decision-making, Message Queuing Telemetry Transport (MQTT) for effective data transfer, and cloud computing for data storage and analysis. The suggested system offers a simple, transportable, and affordable method for maximizing electricity usage by utilizing these technologies. Users can assess their weekly and daily electricity usage, and the system also offers Grafana visualization features to improve energy awareness and appliance control.

Another work [34]–[37], mentioned the Smart Grid as a new tech upgrade that uses digital tools and improved ways of communication to understand and react to changes in how much electricity is used. This is set to completely change how electricity is distributed, sent, and created. In a conventional network grid, the energy pattern usage is hard to determine, and any energy loss is unnoticeable which results in energy waste. Authors [35] use the big data to do the comprehensive data management function. On the other side,[36] uses FPGA or Field Programmable Gate Arrays (FPGAs) to improve the efficiency and functionality of the energy management system.

Based on the review papers, [12], [21]–[38] are related to IOT-based system where the integration between the energy comes from the grid, or PV system and the demand side management are all controlled via the Internet.

Other paper such as [39] focuses on reducing the electricity bill. The authors create a centralized home energy management system that utilizes time-of-use (TOU) tariff, demand response, and optimization techniques to minimize electricity bills for end-users. The model is developed using the MILP framework and implemented in Python, allowing for easy accessibility and utilization of open-source optimization tools.

2.2.1 Summarization of the Existing Energy Management System

Through these papers, several trends and common technologies can be identified. The reviewed papers emphasize the utilization of IoT (Internet of Things) technology, smart plugs, smart grid concepts, and time-of-use (TOU) tariffs. These technologies play a crucial role in monitoring, controlling, and optimizing energy consumption in diverse settings such as residential buildings, microgrids, and smart homes. Table 2-1 summarization on the technologies used in EMS.

Table 2-1 Specific Technologies Implementation used in EMS

Papers	Specific Technologies Implementation
[12]	IoT, voice recognition, wireless automation system
[21]	Zigbee communication protocol, smart sockets, programmable air conditioner remote controls
[22]	Smart sockets, cloud infrastructure, Mist hubs
[23]	IoT-based methodologies, smart plugs, wireless gateways, time-series analysis
[24]	IoT, artificial neural network (ANN), particle swarm optimization (PSO), support vector regression (SVR)
[25]	IoT, stochastic economic dispatch optimizer (EDO), web-based GUI, API

[26]	AI-based optimization algorithm, MATLAB simulations
[27]	STM32 microcontrollers, IoT technology, AWS IoT Analytics
[28]	IoT-based technologies, wireless communication, smart meters, smart sensors, actuators
[29]	Smart home technology, fuzzy logic, IoT, machine-to-machine communication
[30], [33]	IoT, Raspberry Pi, Node-RED, NodeMCU, Arithmetic Optimization Algorithm (AOA)
[31]	IoT-based devices, linear model algorithms
[32]	IoT, machine learning (Rule-Based Classifiers Technique), multi-agent system (MAS)
[34]–[37]	Smart Grid, big data, Field Programmable Gate Arrays (FPGAs)
[39]	Time-of-use (TOU) tariff, demand response, optimization techniques, MILP framework, Python

The application of algorithm technologies is critical in the design and implementation of energy management systems (EMS). The publications that were reviewed shed light on numerous algorithmic approaches used to optimize energy usage, improve efficiency, and reduce costs in a variety of scenarios including residential buildings, microgrids, and smart homes. Table 2-2 is the summary of the algorithm used by researchers based on the reviewed paper.

Table 2-2 Algorithm Techniques Implementation

Papers	Algorithm Techniques Implementation
[12]	Simple Automation (Integration of IoT with voice recognition)
[21], [26], [32]	Smart-Rule Based Algorithm, Rule-Based Classifiers Technique
[29]	Fuzzy logic
[23]	Time-series analysis, data-driven approaches

[24]	Artificial Neural Network (ANN), Particle Swarm Optimization (PSO), Support Vector Regression (SVR), Ensemble Methods (ENS)
[25]	Stochastic Economic Dispatch Optimizer (EDO)
[26]	AI-Based optimization algorithm (Particle Swarm Optimization)
[30]	Arithmetic Optimization Algorithm (AOA)
[31]	Linear model algorithms
[39]	MILP (Mixed Integer Linear Programming) framework

The following subtopics of the literature review begins with an exploration of the Internet of Things (IoT) and its fundamental role in the technological integration of various devices and systems. The advent of IoT has facilitated unprecedented connectivity and data exchange across numerous devices, making it a cornerstone of modern EMS.

Part of this discussion will delve into the critical role of protocol communication in the Energy Management System (EMS). Protocols form the backbone of efficient information transfer across various EMS devices and technologies. The key is to provide a comprehensive understanding of these communication protocols and their implications for the efficiency and reliability of energy management services.

In the context of IoT, the review will provide an overview of other devices and technologies currently influencing the evolution of EMS. Innovative tools like smart plugs and advanced monitoring systems, when integrated into the EMS through IoT, have the potential to improve energy efficiency and conservation. This section will explore these technologies and their impact on EMS development and performance.

Following the discussion on IoT, the focus then shifts to the algorithms embedded in the EMS. These algorithms are at the heart of the massive volumes of data processed and interpreted to inform energy distribution and consumption decisions. This section of the review delves into the inner workings of these algorithms, emphasizing their pivotal role in facilitating effective energy usage and management.

Another crucial aspect that this review seeks to investigate is machine learning. With its inherent ability to identify patterns within large datasets, machine learning can dramatically enhance the accuracy of energy consumption projections, refine decision-making processes, and improve overall EMS efficiency. This section delves deeply into a thorough examination of machine learning models, their applications, potential drawbacks, and implications in the context of EMS.

2.3 Internet of Things (IoT) on Technology Integration

The concept of the Internet of Things (IoT) is often traced back to the early 1980s, with the first practical application generally considered to be a modified Coke machine at Carnegie Mellon University [40]. This machine was internet-connected, allowing it to report its inventory and whether newly loaded drinks were cold or not. This is often considered the first "Internet-connected appliance" or the first instance of the Internet of Things.

However, it's important to note that the term "Internet of Things" wasn't coined until much later. In 1999, British technology pioneer Kevin Ashton is credited with coining the term while working for Procter & Gamble. He used the term to describe a system where the "Internet is connected to the physical world via ubiquitous sensors" [41].

In the context of technology integration, IoT began to take shape with more significance in the early 2000s. With advancements in wireless networking, micro-electromechanical systems (MEMS), and the Internet, devices began to be designed with built-in sensors that could communicate and interact with the external environment, enabling technology integration on a new level [42].

That said, the use of IoT for technology integration has evolved over time and continues to do so, with applications becoming increasingly sophisticated and diverse. It's a field that's seen continuous innovation and advancement, with new use cases being developed all the time. According to Statista, the total number of installed connected devices is expected to be 75.4 billion globally by 2025 [43]. This would be the fifth time it has increased since 2015. These numbers indicate that the future of IoT promises to be more innovative and revolutionary as compared to the present. The rapid growth of IoT has been supported by advancements in three technologies — Cloud Computing, big data, and artificial intelligence. Furthermore, as more and more data are generated by the IoT systems, product companies would shift towards a service ecosystem.

2.3.1 Communication Protocol

Communication protocols are a set of rules that determine how data is transmitted and received in a network [44]. In the context of IoT, these protocols ensure seamless communication between the different devices and systems integrated through IoT. Given that IoT devices often have different hardware configurations and operational requirements, choosing the right communication protocol is crucial to ensure efficient and reliable data transfer.

Three commonly used communication protocols in IoT are Zigbee (802.15.4), Wi-Fi (802.11), and Bluetooth (802.15.1) [45]. These wireless communications consume little power [46].

2.3.1.1 Zigbee (802.15.4)

Zigbee is a high-level communication protocol used to create personal area networks with small, low-power digital radios. It's especially suited for systems requiring low data rate, long battery life, and secure networking. Zigbee is often used in low-rate private area networks (LR-WPANs) and is designed for applications that require low power consumption and low data rates. It's a popular choice for home automation, medical data collection, and other low-power, low-bandwidth needs.

As of now, Zigbee 3.0 uses cryptographic algorithms for two levels of symmetrical AES-128-CCM encryption: network and APS, which provide both authentication and confidentiality. In addition, Zigbee 3.0 R23, the next version of the Zigbee PRO standard, is expected to further enhance security. A new feature called Zigbee Direct is set to improve consumer experience and simplify automation by bringing together Zigbee and Bluetooth Low Energy, enabling users to interact with their Zigbee networks using a smartphone, tablet, or other Bluetooth enabled device [47].

2.3.1.2 Wi-Fi (802.11)

Wi-Fi is a wireless networking technology that employs radio frequencies to provide high-speed wireless internet and network connections. It supports high data rates and operates over a greater distance than Bluetooth and Zigbee. Wi-Fi is most commonly used for Internet access in homes and offices. Wi-Fi has a higher power consumption than Zigbee and Bluetooth, which can be a problem for battery-powered Internet of Things devices.

Wi-Fi is also becoming a premier option for IoT, with features that improve performance in dense environments and reduce battery consumption, making it an ideal solution for sensor-based devices. Wi-Fi CERTIFIED HaLow delivers long-range, low-power Wi-Fi, enabling power-efficient use cases in areas like smart homes, healthcare, and smart cities. Furthermore, advancements in network optimization technologies have led to Wi-Fi networks reaching new optimization levels, offering greater mobility, enhanced network efficiency, and improved network visibility and management in home and enterprise networks [48].

2.3.1.3 Bluetooth (802.15.1)

Bluetooth is a wireless technology standard used for short-distance data transfer. It is frequently used to attach peripheral devices, such as headphones, mice, and keyboards, to computers, smartphones, and other host devices. Bluetooth Low Energy (BLE), a power-efficient Bluetooth variant, is gaining popularity for Internet of Things (IoT) applications[49]. Bluetooth has a shorter range and slower data rate than Wi-Fi, but it consumes less power and is optimal for short-range device communication.

2.3.1.4 Comparison Between Zigbee, Wi-Fi and Bluetooth

Table 2-3 Summary of Strength and Weakness between Zigbee, Wi-Fi and Bluetooth

Technology	Strengths	Weaknesses
Zigbee	Low power consumption, secure, supports mesh networking, ideal for low-data-rate applications	Shorter range than Wi-Fi, lower data rate
Wi-Fi	High data rate, long range, supports a large number of devices	Higher power consumption, not ideal for battery-powered devices
Bluetooth	Short-range communication, low power consumption (especially BLE), ideal for personal area networks	Not suited for long-range communication, lower data rate compared to Wi-Fi

There are several communication protocols that can be used for energy management systems in residential buildings. ZigBee technology has been shown to be effective in energy management systems for residential buildings [45]. [21], [24], [26], [28], [34], [36] uses the ZigBee technology to abolish the complexity of wiring. Others, such as [12], [22], [27], [29]–[33] use Wi-Fi for communication because they carry out energy management using respective algorithms that reduce user power consumption as well as load and battery management, which requires large amounts of data.

The selection of an optimal communication protocol for an Energy Management System (EMS) is contingent upon several key factors, each of which contributes to the overall efficiency and effectiveness of the system. These factors include the range of the network, power consumption of devices, data transmission rate, data security, potential interference, and the complexity of the network.

The range of the network pertains to the distance between devices within the system. Should devices be dispersed over a large geographical area, a protocol with a longer transmission range may be necessary. In this case, a protocol such as Wi-Fi could potentially be the optimal choice due to its extensive range capabilities.

The power consumption of the devices in the network is another pivotal consideration. For devices that are battery-powered and require extended periods of operation without recharging, a protocol with lower power consumption is preferred. For instance, Zigbee, known for its efficient power usage, could be the suitable choice for such scenarios.

Data transmission rate, or the speed at which data needs to be conveyed, also influences the selection of the communication protocol. If the system necessitates the rapid transmission of large volumes of data, a high-data-rate protocol, such as Wi-Fi, might be the superior choice.

The security of data transmission is an integral component when choosing a communication protocol for an EMS. If the sensitivity of the data being transmitted is of high concern, it is imperative to select a protocol with robust security features. Both Zigbee and Wi-Fi are equipped with substantial security features, however, the specific choice will depend on the unique use case.

The potential for interference from other wireless devices is another important factor. Certain protocols are better equipped to handle interference, thus the selection might be influenced by the presence of other wireless devices within the vicinity.

Lastly, the complexity of the network also plays a role in the selection of the communication protocol. For networks comprising a large number of devices, or in instances where devices need to communicate in complex ways, a protocol that supports mesh networking might be necessary. Zigbee, for example, supports mesh networking and could be a suitable choice for such networks.

In light of these considerations, Zigbee often emerges as a promising choice for EMS. It is designed for low-rate, low-power applications and supports mesh networking, making it ideal for connecting a large number of devices in an energy management system. Additionally, Zigbee's robust security features could be beneficial in contexts where data security is a priority.

However, Wi-Fi may present itself as a more suitable option if the system needs to transmit larger volumes of data over longer distances, or if it needs to integrate with existing Wi-Fi networks.

2.3.2 Sensor Technology

Sensor technology refers to the use of devices or systems capable of detecting changes in the physical or chemical conditions of an environment and converting these changes into signals that can be measured or interpreted. This can include temperature, pressure, light, sound, and many other types of sensors. In its broadest sense, the concept of sensors dates back to the late 1800s, when the first temperature sensor based on a copper resistor was created [50]. Sensors play an important role in a wide range of technological applications, from simple everyday items to complex space exploration equipment. A sensor, in its most basic form, is a device that detects and responds to some type of input from the physical environment. Light, heat, motion, moisture, pressure, or any of hundreds of other environmental phenomena could be the specific input. In general, the output is a signal that is transformed to a human-readable display at the sensor location or electronically transmitted via a network for reading or additional processing.

In terms of construction, sensors can be classified into two broad categories: analog and digital sensors. Analog sensors produce a continuous output signal or data that is generally proportional to the measure of the physical property. Examples of such sensors include thermistors and Light Dependent Resistors (LDRs). On the other hand, digital sensors produce a discrete output signal or data which is typically in the form of binary language. The output data in this case can only be read by a device or viewer designed to interpret the binary data. Examples of digital sensors include digital temperature sensors and optical encoders.

Sensor technology is a critical component of the Internet of Things (IoT). It plays a fundamental role in enabling devices to monitor, track, and measure various aspects of the physical world and convert this data into meaningful information.

Traditionally, sensors have been functionally straightforward devices that convert physical variables into electrical signals or electrical property changes. While this functionality is a necessary starting point, sensors must also possess the following characteristics to function as Internet of Things components [51]:

- Low cost, allowing economical deployment in large quantities.
- Small enough to "disappear" undetected into any environment.
- In most cases, a wired connection is not feasible.
- Identification and validation of oneself
- Very low power consumption, so it can operate for years without a battery replacement or with energy harvesting.
- Robust, to minimize or eradicate maintenance.
- Self-evaluating and -healing
- Self-calibrating or wirelessly accepting calibration commands.
- Data preprocessing to reduce the burden on gateways, PLCs, and cloud resources.

Advancements in sensor technology hold enormous potential to revolutionize various applications within the Energy Management System (EMS). The capacity of a sensor to detect changes in physical or chemical conditions and convert them into measurable signals can be leveraged to optimize energy consumption, enhance the efficiency of energy systems, and contribute to sustainable energy management.

For example, when integrated with an EMS, sensors can monitor energy usage in real-time, providing crucial data to identify inefficiencies and areas for reducing energy consumption. With the aid of the Internet of Things (IoT), these sensors can connect to a network, enabling remote monitoring and control of energy systems, which leads to improved operational efficiency and substantial energy savings.

One practical application of this technology lies within building energy management. In this context, temperature sensors, light sensors, and occupancy sensors could be utilized to automatically adjust heating, ventilation, and air conditioning (HVAC) systems, as well as lighting, based on occupancy and ambient conditions within the building. This automation could drastically reduce energy wastage and enhance the overall energy efficiency of the building.

In the field of renewable energy systems, sensors also play an essential role. For instance, in solar photovoltaic systems, sensors can monitor the efficiency of solar panels, detect faults, and note decreases in output. This data can be used to optimize the performance of the solar system, ensuring it operates at peak efficiency.

2.4 Application of Algorithms in IoT-Based Energy Management Systems

The transformative influence of algorithms on IoT-based Energy Management Systems (EMS) cannot be understated. They serve as the foundation for the development of systems that exhibit enhanced responsiveness, adaptability, and proficiency in energy efficiency. In the ensuing discussion, this literature review will dissect the intricacies of three pivotal classes of algorithms that have indelibly influenced the landscape of IoT-based EMS - namely, rule-based techniques, optimization techniques, and AI-Based Techniques algorithms.

Rule-based algorithms, which provide a preset set of instructions, have greatly improved the responsiveness of these systems. They respond immediately to energy demands depending on predefined criteria. The following are AI-Based algorithms. Their self-learning capabilities have advanced IoT-based EMS to a new level of efficiency. By utilizing historical data to understand and predict future energy consumption trends, these algorithms drive significant energy savings and enhance the overall performance of the system.

Completing the triad are optimization techniques, which include both linear and nonlinear programming, have given IoT-based EMS an unprecedented level of adaptability. They arrange the efficient use of energy resources by maximizing or limiting predetermined objectives. This comprehensive investigation endeavors to shed light on the significant role played by the various categories of algorithms in the realm of Internet of Things (IoT)-based Energy Management Systems (EMS).

2.4.1 Rule-Based Techniques

Rule-based algorithms and rule-based classifiers are both types of artificial intelligence systems that use a set of predefined rules to make decisions. However, they are typically used in different contexts and have different characteristics.

2.4.1.1 Rule-Based Algorithm

These are general algorithms that operate based on a set of predefined rules [26]. In this section, the Rule-Based system proposed by the authors [21] will be examined in detail. As mentioned in the Existing Energy Management System Overview section, The system consists of 3 major modules which are programmable air conditioner remote controls, smart sockets for monitoring appliance power use, and nodes for monitoring room temperature.

The 1st module that isthe programmable air conditioner remote control, is integrated with the proposed HEM (also known as EMS) system so that the system has the flexibility to control the AC units. The air conditioner remote control is able to learn the IR digital code patterns transmitted by the original AC units. The patterns are stored in the microcontroller's memory as digital commands with the various temperature settings.

The 2nd module, the smart sockets for monitoring appliance power use, is used in the EMS system. The smart plug/socket can keep track of how much power a connected device up to 13 A uses and turn it on or off using a built-in relay. Single-phase power line voltage and current are measured by the plug/socket, processed, and transmitted to the controller.

The 3rd and final module consist of temperature monitoring nodes, incorporated into the proposed HEM. This module monitors ambient conditions including room temperature, humidity, illumination, and CO2 concentration. The circuit is also capable of detecting motion in the room via a motion sensor and can therefore provide the inputs required by the decision-making algorithm to the main controller.

All of these three modules are then actuated through the rule-based algorithm. A rule-based system is aimed to reduce energy use on a repeatable basis during a 24-hour period. The algorithm prioritizes user comfort, making it easier to integrate the system into the consumer's daily routine without disrupting their lifestyle. The algorithm gathers data from a variety of sources, including smart sockets that monitor the amount of energy consumed by connected appliances, room condition monitoring circuitry that provides data on ambient conditions such as room temperature, humidity, illuminance, and CO2 concentration, and a scheduling terminal that retrieves day-ahead pricing from the website of the utility service specified by the user. Based on information gathered from the module, control commands are created and delivered to specified smart plugs to turn on/off linked appliances, Zigbee-connected dimmers to reduce/increase light intensity of dimmable LEDs, and Zigbee-connected infrared remote controls to set ACs to specific temperatures.

First the authors developed the algorithm to establish comfort ranges for the measured appliance. Before developing the algorithm, the authors conducted a survey to determine the user's preferences regarding the room's condition, the appliance's on/off status, the brightness of the room's lighting, etc. The preset user comfort ranges algorithm equation is as in Equation (**2-1**(2-1) below:

$$\begin{gathered} T_{REF\_min} \leq T_{REF_t} \leq T_{REF\_max} \\ T_{ROOM\_min} \leq T_{ROOM_t} \leq T_{ROOM\_max} \\ LUX_{min} \leq LUX_{ROOM_t} \leq LUX_{max} \\ T_{W\_min} \leq T_{W_t} \leq T_{W\_max} \end{gathered} \tag{2-1}$$

T_{REF} is the refrigerator temperature, T_{ROOM} is the room temperature, T_W is the water temperature and LUX is the illuminance. The maximum and minimum values of the ranges indicate the optimal parameter values. These parameters are based on setup appliance of the energy management. The author then creates scheduling rules with new parameters like time-of-use (TOU), power consumption, room occupancy (v), and desired appliance performance, where the parameters are acquired from several circuitries form the proposed HEM system. Proposed schedule Equations (2-2),(2-3),(2-4) and (2-5) are as follows:

$$S_{LIGHT} = \begin{bmatrix} 0\%, & LUX_{ROOM\_t} = 0 \Rightarrow \text{ if } \mathrm{v} = 0 \\ 90\%, & \mathrm{LUX}_{\min} \leq LUX_{ROOM\_t} \leq \mathrm{LUX}_{\max} \Rightarrow \text{ if } \mathrm{v} = 1 \,\&\, \mathrm{t} = \mathrm{t}_{\text{nonpeak}} \\ 100\%, & 0 \leq LUX_{ROOM\_t} \leq \mathrm{LUX}_{\min} \Rightarrow \text{ if } \mathrm{v} = 1 \,\&\, \mathrm{t} = \mathrm{t}_{\text{nonpeak}} \\ 50\%, & (\mathrm{LUX}_{\max} + \mathrm{LUX}_{\min})/2 \leq LUX_{ROOM\_t} \Rightarrow \text{ if } \mathrm{v} = 1 \,\&\, \mathrm{t} = \mathrm{t}_{\text{peak}} \end{bmatrix} \tag{2-2}$$

$$S_{AC} = \begin{bmatrix} 0, & T_{ROOM\_\min} \leq T_{ROOM\_t} \leq T_{ROOM\_\max} \Rightarrow \text{ if } \mathrm{V} = 0 \\ 21, & T_{ROOM\_\min} \leq T_{ROOM\_t} \Rightarrow \text{ if } \mathrm{V} = 1 \,\&\, \mathrm{t} = \mathrm{t}_{\min} \\ 23, & T_{ROOM\_\min} \leq T_{ROOM\_t} \leq T_{ROOM\_\max} \Rightarrow \text{ if } \mathrm{V} = 1 \,\&\, \mathrm{t} = \mathrm{t}_{\text{nonpeak}} \\ 25, & T_{ROOM\_t} \leq T_{ROOM\_\max} \Rightarrow \text{ if } \mathrm{V} = 1 \,\&\, \mathrm{t} = \mathrm{t}_{\text{peak}} \\ S_{AC\_t-1}, & T_{ROOM\_\min} \leq T_{ROOM\_t} \leq T_{ROOM\_\max} \Rightarrow \text{ if } \mathrm{V} = 1 \,\&\, \mathrm{t} = \mathrm{t}_{\text{nonpeak}} \end{bmatrix} \tag{2-3}$$

$$S_{REF} = \begin{bmatrix} 1, & T_{REF\_\max} \leq T_{REF\_t} \\ 0, & T_{REF\_t} < T_{REF\_\min} \\ 0, & T_{REF\_\min} \leq T_{REF\_t} \leq T_{REF\_\max} \,\&\, T_{ROOM\_\min} \leq T_{ROOM\_t} \Rightarrow \text{ if } \mathrm{t} = \mathrm{t}_{\text{peak}} \\ S_{REF\_t-1}, & T_{REF\_\min} \leq T_{REF\_t} \leq T_{REF\_\max} \Rightarrow \text{ if } \mathrm{t} = \mathrm{t}_{\text{nonpeak}} \end{bmatrix} \tag{2-4}$$

$$S_{Heater} = \begin{bmatrix} 1, & T_{Heater\_t} \leq T_{Heater\_\min} \\ 0, & T_{Heater\_\max} \leq T_{Heater\_t} \\ 0, & T_{Heater\_\min} \leq T_{Heater\_t} \leq T_{Heater\_\max} \Rightarrow \text{ if } \mathrm{t} = \mathrm{t}_{\text{peak}} \\ S_{REF\_t-1}, & T_{Heater\_\min} \leq T_{Heater\_t} \leq T_{Heater\_\max} \Rightarrow \text{ if } \mathrm{t} = \mathrm{t}_{\text{nonpeak}} \end{bmatrix} \tag{2-5}$$

Equations (2-2) until (2-5) shows that the appliances On and Off will be controlled by the algorithm as long as parameters value are within the range. For example, in equation (2-2), the room's luminance is set to 50 percent when it is vacant [21]. The remaining illumination that remains within the comfort range during peak pricing hours will reduce energy consumption while maintaining user comfort.

In conclusion, the rule-based algorithm presented for Energy Management Systems (EMS) demonstrates a sophisticated approach to energy management, particularly from an integrative standpoint. The success of this algorithm is shown by its ability to control energy use without disrupting the everyday life of the user. The authors' work is an important addition to rule-based algorithms for EMS, as it skillfully combines technology with the needs for human comfort. Even though it's complex to include many different factors, the rule-based system, as described in this review, seems to expertly handle the tricky balance between saving energy and keeping users comfortable. This gives a hopeful direction for future progress in the EMS field.

2.4.1.2 Rule-Based Classifier Techniques

Rule-based Classifier Techniques generate a comprehensive set of "if-then" rules to predict outcomes based on input data [52]. These principles are derived from training data and can provide both classification and an explanation for why a particular instance was classified in a particular manner. The construction of a rule-based classifier entails several key steps: rule generation, rule pruning, and rule ordering [53].

- **Rule Generation**: This involves the creation of rules based on the training data. There are various methods to generate these rules, including separate-and-conquer algorithms, decision tree induction, association rule learning, and genetic algorithms. Each method has its strengths and weaknesses, and the choice depends on the specific use case.
- **Rule Pruning**: This is the process of removing or modifying rules to improve the accuracy and simplicity of the model. Overly complex rules can lead to overfitting, meaning the model becomes too specialized for the training data and performs poorly with unseen data.
- **Rule Ordering**: Rules are ordered by their estimated quality. This means the most reliable rules are placed at the beginning of the list. When classifying a new instance, the system will check each rule in order and assign the class specified by the first rule that matches the instance.

The utilization of rule-based classifiers may play a significant role within Energy Management Systems (EMS), which are responsible for the surveillance, regulation, and optimization of electricity generation and/or consumption.

Authors in [32] uses the Ruled-Based Classifiers at the Demand Side Management (DMS) System. The DSM system collects the total house demand (HD) power consumption data from the smart plug and preprocessed the data before initiate the calculations in the DSM system as shown in Figure 2.1

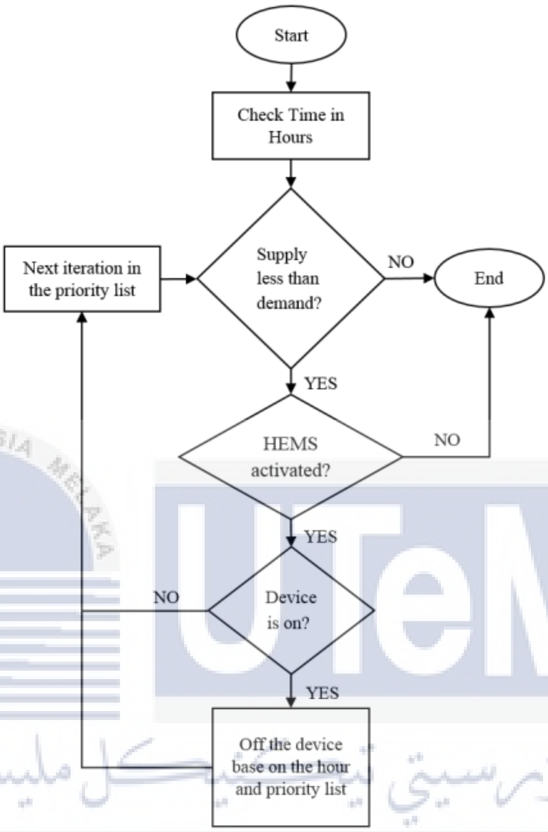


Figure 2.1 DSM System Flowchart [32]

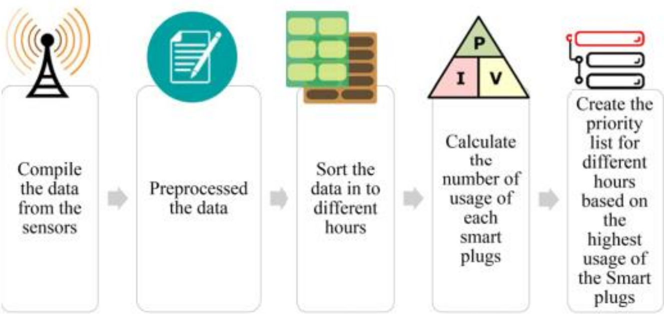


Figure 2.2 Preprocessing procedure [32]

The preprocessed procedure is where the data of the power consumption by the user is identified through the user's actions number of ON and OFF of the plugs. Then this process will sort it out into different hours, calculate the usage of the smart plugs and generate the historical data through a priority list. The priority list will commence with the highest power usage as the top priority and the low usage as the low priority on the list. Then the lowest priority on the list will be OFF when the power demand exceeds the power supply until the power demand is lower than the power supply.

2.4.1.3 Refining Rules in Rule-Based System

If the algorithm is underperforming, several strategies can be implemented [54]:

- **Refine the Rules**: The rules might not accurately reflect the complexity of the system. They could be based on outdated or incomplete information. In such cases, it would be important to revisit the rule-generation process.
- **Incorporate More Data**: If the rule-based classifier is not providing the desired results, it could be that more or different types of data are needed. For example, incorporating weather forecasts or building occupancy data might improve performance.
- **Combine with Other Methods**: Rule-based classifiers can also be combined with other machine-learning techniques to form a hybrid model. For example, a neural network could be used to generate a set of potential rules, which are then pruned and ordered using traditional rule-based techniques.
- **Continuous Monitoring and Adjustment**: An important aspect of using rule-based classifiers, or any machine learning model, in a real-world application like an EMS, is the need for continuous monitoring and adjustment. As the system and environment change over time, so should the rules.
- **Fine-tuning**: The model parameters can be adjusted to get better performance. For example, the threshold of a rule can be adjusted to better fit the specific context of the energy management system.

2.4.2 AI-Based Techniques in EMS

AI-based techniques are important in energy management systems because they use advanced algorithms and computational models to optimize energy use, improve efficiency, and minimize costs. Artificial neural networks (ANNs) and fuzzy logic control (FLC) are two popular AI techniques utilized in energy management systems.

2.4.2.1 Artificial Neural Networks

ANNs (Artificial Neural Networks): ANNs are computer models inspired by the form and operation of biological neural networks, such as the human brain. They are made up of interconnected nodes termed artificial neurons or perceptrons that are arranged in layers. ANNs are trained on a dataset to discover patterns, correlations, and linkages. ANNs that have been trained can make predictions or choices based on new inputs.

When it comes to energy management systems, ANNs can be quite beneficial. Energy management involves planning and operating energy production and consumption units. The goal is to conserve resources, protect the environment, and reduce costs. This involves the use of renewable energy resources, demand forecasting, optimization of energy use, and more. ANNs can help with energy forecasting, which is crucial for efficient energy management. They can learn patterns in energy usage and predict future consumption based on historical data. This can be used to optimize energy production and distribution.

Furthermore, ANNs can be used to control and optimize energy usage in buildings and industry. For instance, an ANN could learn the energy usage patterns of a building and control heating, ventilation, and air conditioning (HVAC) systems to minimize energy use while still meeting occupants' needs.

In [24], an architecture of an Intelligent Smart Energy Management System (ISEMS) that capable of hourly and daily accurate renewable energy (solar energy) forecasting is proposed. Thus, this section will examine [24] proposed architecture in which the authors tested several machine learning prediction models as a technique for forecasting solar energy generation.

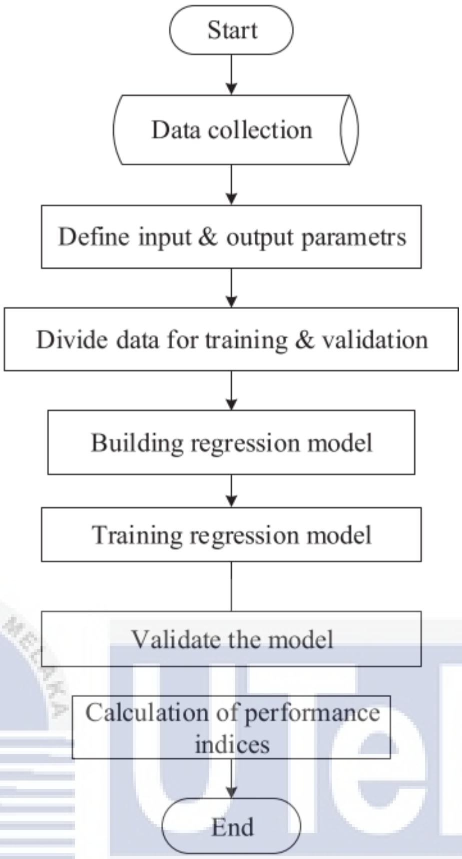


Figure 2.3 The flowchart of the proposed model [24]

Figure 2.3 shows the flow of the proposed prediction model where the data collection is gathered through the ISEMS data collection module. From the data collection, the preprocessing is done before goes into the regression model. The regression model in the flow chart is used to determine a function that approximates the target values accurately using a set of input values. Initial variable, max-depth and coefficients are varied to build an accurate model [24]. Then, the prediction model is trained with the data given. The prediction model or known as the ANN algorithm, will be in the training phase that splits into training and testing sections. The operation of the Artificial Neural Network (ANN) model encompasses the analysis of historical data, integrating a diverse set of input parameters, including temperature, wind speed, time of day, and month. Subsequently, the data is partitioned, with 75% allocated to form the training set, and the remaining 25% designated as the testing set.

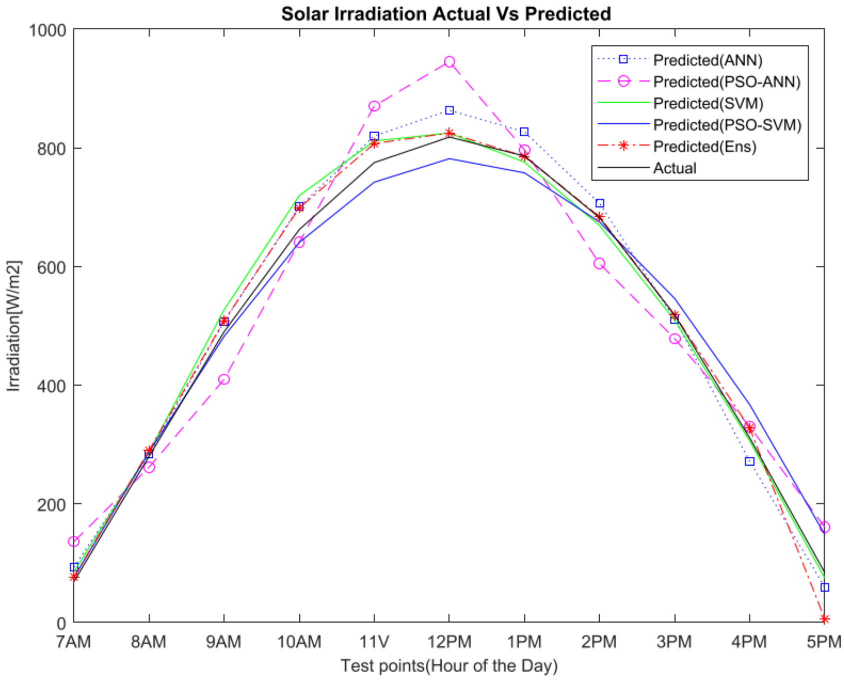


Figure 2.4 Prediction Results using Different Types of Prediction Models [24]

Figure 2.4 illustrates that the ANN models predict solar irradiance less accurately than the other models, but better than PSO-ANN. Other models such as PSO-ANN, SVM, Ens and PSO-SVM show better performance in predicting when the AI Based Techniques are combined with the optimization techniques. In these results, the PSO- SVM model outperforms the rest in terms of accuracy.

In conclusion, Artificial Neural Networks (ANNs), with their ability to analyze and interpret complex datasets, provide significant advantages in energy management systems. They hold considerable potential for optimizing energy production and distribution, controlling energy usage in buildings, and even forecasting future energy demands. However, while ANNs can effectively learn and predict based on historical data and diverse input parameters, the accuracy of their predictions might still be improved.

The exploration of other machine learning models, as discussed in [24], introduces the concept of enhancing ANN performance by integrating it with other techniques. The proposed Intelligent Smart Energy Management System (ISEMS) architecture includes the application of different prediction models, including the

ANN, to forecast solar energy generation. While the ANN model demonstrated commendable predictive capability, it was observed that it was not the most accurate when compared to other models, such as the PSO-ANN, SVM, Ens, and PSO-SVM.

Of note is the superior performance of models like the PSO-SVM, which combines artificial intelligence techniques with optimization methods, signifying that the integration of ANNs with additional strategies can potentially enhance their predictive accuracy. Therefore, the incorporation of ANNs with other techniques in energy management systems is a promising direction for future research and development, with the aim of achieving more accurate and efficient energy forecasting and optimization.

2.4.2.2 Fuzzy Logic Control

Fuzzy logic control systems play a significant role in engineering and are often the natural choice for designing control applications. They are popular and appropriate for controlling home and industrial appliances, and there's constant research by academic and industrial experts to propose innovative and effective fuzzy control systems. Applications of these systems can be found in energy and power systems, navigation systems, imaging, and industrial engineering [55].

A fuzzy control system is based on fuzzy logic, a mathematical system that analyzes analog input values in terms of logical variables that can take on continuous values between 0 and 1, as opposed to classical or digital logic that operates on discrete values of either 1 or 0 (true or false, respectively) [56]. Fuzzy logic allows for the handling of concepts that cannot be strictly classified as true or false, but rather as partially true, making it easier to mechanize tasks that are already successfully performed by humans [56].

Fuzzy logic was proposed by Lotfi A. Zadeh in 1965, and its first industrial application came in 1975 in a cement kiln in Denmark [57]. Japanese engineers subsequently developed a wide range of fuzzy systems for both industrial and consumer applications, such as vacuum cleaners that use microcontrollers running fuzzy algorithms to adjust suction power, and washing machines that use fuzzy controllers to set the wash cycle for the best use of power, water, and detergent [57].

Fuzzy logic control can be used in energy management systems as well. FLC can be used in energy management systems for load balancing, improving energy distribution, and demand response. Fuzzy logic is capable of dealing with the fluctuation and uncertainty associated with energy demand and supply, allowing for better control and decision-making. FLC allows the system to adapt and respond to changing conditions, ensuring energy efficiency while preserving comfort and stability. In the previous section of the literature review, a paper [29] presented their Energy Management System (EMS) which utilizes Fuzzy Logic Control (FLC).

Fuzzy logic control involves three main steps:

- **Fuzzification**: The crisp input values are converted into fuzzy sets. In this process, the membership function is used to determine the degree to which an input belongs to each of the fuzzy sets.
- **Inference**: The fuzzy rule base, which is a collection of IF-THEN rules, is applied to the fuzzy sets from the fuzzification process. The output of the inference step is a fuzzy set.
- **Defuzzification**: The fuzzy output set from the inference step is converted back into a crisp value, which can be used to control the system.

The main advantage of fuzzy logic control is its ability to handle uncertainty and non-linearity. It is also less dependent on precise mathematical models, making it suitable for complex systems where the relationships between variables are not well understood.

2.4.2.3 XGBoost

XGBoost, which stands for eXtreme Gradient Boosting, is an ensemble learning algorithm based on the Gradient Boosting Decision Tree (GBDT) algorithm

[58]. XGBoost is a machine learning framework specialized in tree boosting. It facilitates the creation of a group of decision trees that perform either classification or regression tasks on the input data. These trees are commonly referred to as CART (Classification and Regression Trees) because of their dual capability. In a decision tree, the response can be binary (determining whether the input data falls into a specific category) or numerical, often expressed through a function. XGBoost employs the numerical approach in tree boosting, resulting in an ensemble output that takes a particular form.

$$\hat{y}_i = \sum_{k=1}^{k} f_x(x_i) \tag{2-6}$$

x_i denotes an input pattern to be classified, $f_x(x_i)$ is the function representing each decision tree's output, K is the total number of trees, and $\hat{y}_i$ is the collective response of the entire tree ensemble. The training process involves using pairs of input and desired output (x_i, y_i), where x_i is the pattern to be classified and y_i is its expected result. The goal is to fine-tune the tree structure's parameters by minimizing a cost function in a supervised manner. However, this task is more challenging compared to learning methods used in other machine learning models, like the gradient descent in neural networks, as simultaneously training all trees can be computationally demanding. Therefore, a simplified, iterative approach called "boosting" is employed, where each tree is trained one step at a time.

The training procedure begins by setting the initial prediction value to zero.

$$\hat{y}_i^0 = 0 \tag{2-7}$$

In this formula, the superscript indicates the time step in the process. At this point, a first tree, characterized by its defining function, is incorporated into the tree ensemble, resulting in the following output:

$$\hat{y}_i^1 = \hat{y}_i^0 + f_1(x_i) = f_1(x_i) \tag{2-8}$$

This newly added tree is trained using a portion of the training dataset, after which predictions are made for the entire dataset. Since several of these predictions might differ from their anticipated values, another tree is introduced and trained specifically with the patterns that were previously misclassified. Consequently, the prediction function of the ensemble is updated to:

$$\hat{y}_i^2 = \hat{y}_i^1 + f_2(x_i) = f_1(x_i) + f_2(x_i) \quad (2\text{-}9)$$

This procedure continues until either a specific level of accuracy is attained, or the total number of trees hits a predetermined limit. At this point, the predictive function of the tree ensemble will be:

$$\hat{y}_i^t = \hat{y}_i^{t-1} + f_t(x_i) = \sum_{k=1}^{t} f_k(x_i) \quad (2\text{-}10)$$

The cost function that needs to be minimized during the training of the trees is:

$$\mathcal{L} = \sum_i l\,(\hat{y}_i, y_i) = \sum_{k=1}^{t} \Omega(f_k) \quad (2\text{-}11)$$

'i' represents the count of training patterns, and 'k' signifies the total number of trees. The term $(\hat{y}_i, y_i)$ quantifies the error in each prediction, commonly using the Mean Squared Error as the metric. The term $\Omega(f_k)$ is a regularization component that assesses the complexity of the tree structures. Its purpose is to encourage the formation of as simple a tree structure as possible.

In the context of energy management systems, especially those integrated with IoT, the predictive capability of XGBoost can be leveraged for forecasting energy prices, energy consumption patterns, or CO2 emissions, leading to more efficient energy use and better environmental policy decisions. [59] paper demonstrates the effectiveness of XGBoost in handling time-series data like EUA prices, which are critical in the energy sector. By accurately predicting these prices, IoT-based energy management systems can optimize energy usage and reduce costs, contributing to more sustainable and economically efficient energy practices.

Machine learning is also good for making energy consumption prediction [60]. The XGBoost model is highly applicable for predicting energy consumption in IoT-based energy management systems. By leveraging its strong predictive capabilities, the XGBoost algorithm can analyze large datasets from various IoT devices to forecast energy usage patterns accurately. This is crucial in optimizing energy distribution, reducing waste, and managing demand-supply dynamics effectively. In energy management systems, such detailed and reliable predictions enable smarter decisions, promoting energy efficiency and sustainability. The integration of XGBoost into IoT frameworks therefore represents a significant advancement in the field of energy management.

2.4.3 Optimization Techniques

Energy management systems (EMS) are complex networks that require efficient optimization techniques to ensure optimal energy usage, reduce costs, improve resilience and sustainability, and mitigate environmental impacts.

2.4.3.1 Linear model algorithms

Linear Programming (LP): Linear programming is a method used in optimization where the objective function and constraints are all linear. Particularly, the Arithmetic Optimization Algorithm (AOA) is a linear programming which is used by authors [30]. In other paper, [39] proposed the mix integer linear programming (MILP) to be implemented as it is proven that the application able to deceasing the daily energy electricity bill by 25%.

LP is a widely used optimization technique in energy management systems to manage energy usage and costs, especially when the system under consideration has continuous variables and linear relationships. It's typically used for planning and scheduling of resources, optimal power flow, and energy dispatch among others [61].

This section will examine two different types of linear model algorithms, namely Arithmetic Optimization and mix integer linear programming (MILP) and highlight their differences. Authors in [30] shows the arithmetic optimization algorithm that authors [62] proposed.

Arithmetic is a fundamental component of number theory, and AOA derives its inspiration from the application of arithmetic operators to solve arithmetic issues. Similar to other population-based algorithms, AOA starts its enhancement process with a randomly generated initial population within the problem's search area. The optimal solution found in each cycle is evaluated against the best solution achieved up to that point and will supersede it if found to be better. The matrix below displays the initial population:

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & x_{1,n-1} & x_{1,n} \\ x_{2,1} & \cdots & x_{2,j} & \cdots & x_{2,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & x_{N-1,j} & \cdots & x_{N-1,n} \\ x_{N,1} & \cdots & x_{N,j} & x_{N,n-1} & x_{N,n} \end{bmatrix} \tag{2-12}$$

N signifies the count of available solutions within the initial group, while n represents the dimensions or aspects of the problem at hand. During each cycle of the AOA optimization procedure, the exploration phase is chosen utilizing the Math Optimizer Accelerate (MOA) function. The formulation of the MOA function is given by the ensuing equation:

$$MOA\,(Iter) = Min + Iter(\frac{Max - Min}{M_{Iter}}) \tag{2-13}$$

In this case, MOA $(Iter)$ symbolizes the function's value at the tth iteration, while Iter stands for the present iteration, and M_{Iter} denotes the highest number of iterations possible. Furthermore, Min and Max correspond to the lowest and highest values, respectively.

On the other hand, Mixed-integer linear programming (MILP) is a mathematical optimization approach that can be used to solve problems that require integer solutions. In this case, "mixed" means that some of the variables are required to be integer, while others are allowed to be continuous. This type of problem is more general than a pure integer linear programming problem, where all variables must be integer. The MILP model must have the Decision variables, objective function and constraints. In [39], the objective of the authors is to minimize the total electricity bill where the equation is:

$$Electricity\ Bill = \sum_{t=1}^{NT} \sum_{i=1}^{NA} Tarrif_t \times P_{i,t} \times \Delta t \tag{2-14}$$

2.4.3.2 Particle Swarm Optimization (PSO)

PSO is a computational method that optimizes a problem by iteratively trying to improve a candidate solution. PSO simulates the behaviours of bird flocking and fish schooling. In energy management, PSO has been used in optimizing power generation, load dispatch, and distributed energy resource management. However, it is important to note that PSO performance highly depends on parameter tuning [63].

As mentioned in the section titled 2.2.1, 'the Summarization of the Existing Energy Management', [24] and [26] implemented PSO to predict the solar irradiance and schedule optimally the appliance operation respectively. Authors in [24] are more on experimenting the prediction models where they integrated the AI-Based techniques with PSO to get better accuracy. Thus, this section will examine the implementation of PSO algorithm in [26] that benefits the Energy Management system.

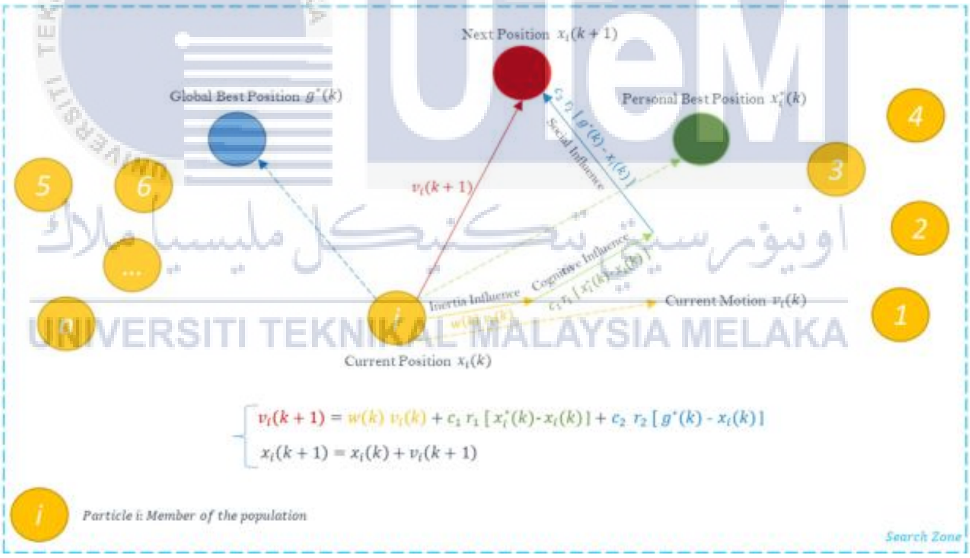


Figure 2.5 Geometric representation of PSO algorithm [26]

Figure 2.5 illustrates an algorithm that showcases the optimization process of a problem through geometric representation. It uses yellow color to depict a population of particles, each of which aims to find the optimal point. As the particles move iteratively, the algorithm updates the search area, along with the individual particle's position x_i, speed v_i and its personal experience (P_{best}^i: representing the best position visited by particle i) and social experience (g_{best}: indicating the best position visited by the entire population). After few iterations, the particles converge towards the optimal zone.

$$\begin{cases} v_i(k+1) = w(k)\, v_i(k) + c_1 r_1 \left[p_{best}^i - x_i(k)\right] + c_2 r_2 \left[g_{best} - x_i(k)\right] \\ x_i(k+1) = x_i(k) + v_i(k+1) \end{cases} \tag{2-15}$$

- $v_i(k+1)$ et $v_i(k)$:designated velocities of particle I in iteration k+1 et k.
- $x_i(k+1)$ et $x_i(k)$: designated position of particle i in iteration k+1 et k.
- c_1 et c_2: cognitive and social learning factor respectively
- r_1 et r_2: two stochastic aleatory variables within the interval [0,1]
- p_{best}^i et g_{best}: best personal position of particle I and best global position of the entire position.

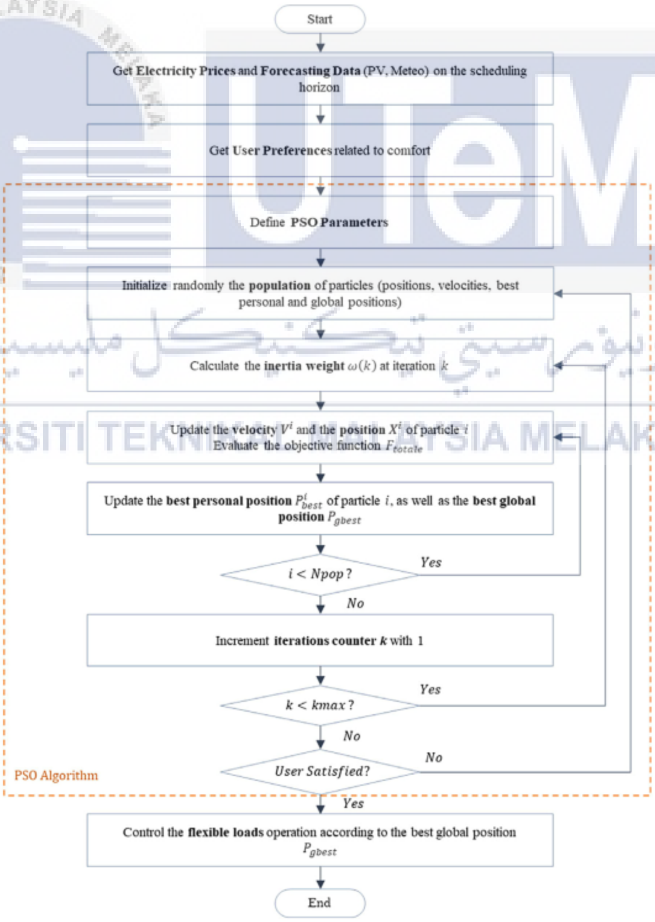


Figure 2.6 PSO-Algorithm for solving the optimization problem [26]

However, the authors proposed the multi objective Particle Swarm Optimization (MOPSO) algorithm to be used in the system as depicted in Figure 2.6.

MOPSO is an extension of PSO that addresses problems with multiple conflicting objectives. In MOPSO, the algorithm aims to find a set of solutions that represent a trade-off between different objectives, instead of a single optimal solution. These solutions are known as the Pareto front or Pareto set. The Pareto front represents a set of solutions where no other solution can improve one objective without sacrificing another. MOPSO algorithms employ techniques such as dominance comparison, crowding distance, or fitness assignment to guide the particles towards the Pareto front and maintain a diverse set of non-dominated solutions [64]. PSO is focused on optimizing a single objective function, while MOPSO is specifically designed to handle problems with multiple conflicting objectives and find a set of solutions that represent the trade-off between these objectives.

2.4.3.3 Stochastic Optimization

Stochastic optimization is a mathematical optimization technique used to solve problems involving randomness or uncertainty. To manage these uncertainties and enhance the system's robustness, stochastic optimization techniques can be employed [65]. Stochastic optimization can be used to address the Stochastic Economic Dispatch (SED) problem in the context of energy management.

In power systems, the Economic Dispatch (ED) problem is a fundamental optimization issue. It entails determining the optimal generation schedule for power plants in order to fulfill anticipated demand at the lowest possible cost while adhering to multiple operational constraints. In practice, however, there are uncertainties in factors such as load demand, renewable energy generation, and fuel prices that can have a significant influence on the efficacy of the power system.

The (SED) problem extends the traditional ED problem by considering these uncertainties explicitly. It incorporates probabilistic models for uncertain variables and aims to find a generation schedule that minimizes the expected cost or maximizes the expected profit, taking into account the stochastic nature of the system. A stochastic economic dispatcher optimizer optimizes the operation of the unbalanced three-phase AC microgrids [25].

Its primary purpose is to determine the optimal day-ahead scheduling of distributed energy resources (DERs) in the microgrid. The EDO employs a mixed-integer linear programming (MILP) model to guarantee the day-ahead dispatch of DERs while adhering to grid constraints such as voltage, current, and power limits. In addition, the optimization module considers security constraints for unplanned islanded operation and stochastic scenarios of local demand and renewable generation. Using an optimal power flow calculation for unbalanced three-phase AC networks, the EDO determines the optimal day-ahead dispatch of DERs.

CHAPTER 3

METHODOLOGY

3.1 Introduction to the Project Methodology

The primary goal of this project is to develop an Internet of Things (IoT) based energy management system, with a particular focus on demand-side management (DSM) for air conditioners and room lighting. This project was conducted in three distinct phases, or experiments, each contributing a specific element to the final system. To achieve this goal, three specific objectives were identified, with each objective corresponding to a distinct experiment in the project.

In the first experiment, a commercially available smart socket was utilized to develop and integrate a real-time monitoring and automation system for these appliances. By capturing real-time power, the system forms a foundation for effective DSM.

In the second experiment, an intelligent algorithm was designed and implemented to manage the energy consumption of these appliances, taking into account the principles of DSM. Algorithms based on machine learning or optimization techniques were considered, aiming to enhance the energy management system's decision-making process about when and how intensively to use energy.

In the third and final experiments, the system's performance was critically analyzed and enhanced based on the output from the implemented algorithm. The central focus was to compare the power consumption efficiency with and without the use of the developed energy management system. Data regarding the energy consumption of the air conditioner and lighting under various conditions were meticulously captured. The accuracy of these measurements, captured by the smart socket, was then evaluated.

All collected data were thoroughly analyzed, and results were represented visually using graphs for better understanding and comparison. The effectiveness of the implemented algorithm in enabling efficient DSM and reducing energy consumption was assessed based on these analyses.

Finally, based on the insights gained from the visual data analysis, enhancement strategies were developed. These could include fine-tuning the algorithm, adding more variables or factors, or improving the hardware for better data accuracy for future research.

3.2 Development Design

This section of this methodology chapter outlines the systematic approach employed in the creation of an IoT-based energy management system. This system aims to enhance energy efficiency by automating the control of air conditioner and lighting systems while incorporating a rule-based algorithm. By integrating cutting-edge technologies and intelligent decision-making processes, this development endeavors to optimize energy consumption and improve overall Energy Management system. This section delves into understanding of the system's development and its potential for achieving significant energy savings.

In addition, this development will involve conducting several experiments to assess the system's performance and identify areas for potential improvement. A detailed explanation of these experiments will be provided in the subsequent section, "Experimental Setup.". These experiments will be carried out to evaluate the effectiveness of various parameters, such as energy consumption, comfort levels, and response times, under different scenarios and conditions. By thoroughly analyzing the results obtained from these experiments, the design can be refined to ensure optimal functionality. This iterative approach will enable the system to evolve and adapt, ultimately leading to an enhanced energy management solution that maximizes efficiency and user satisfaction.

3.2.1 Parameters

Several parameters investigate the vital variables influencing the Energy Management System (EMS) algorithm and the automation of the air conditioning and lighting system. These parameters serve as the basis for developing effective algorithms and automation strategies. The following parameters were considered:

Table 3-1 Parameters Involved for the Development of Energy Management System

Parameters	Description
Power Consumption (W)	The energy consumption of appliances, specifically air conditioners and room lighting, is a crucial parameter in designing an energy management system. By monitoring and optimizing power consumption, it is aimed to reduce energy waste and promote efficient usage.
Illuminance (lux)	Illuminance, measured in lux, plays a significant role in the room lighting automation system. The incorporation of a light sensor and subsequent adjustment of LED bulb brightness facilitate efficient lighting control, contingent on the natural lighting conditions of the room.
Room Temperature (°C)	Room temperature is a critical factor in the automation of air conditioner energy consumption. The process of monitoring and regulating the temperature, in accordance with user preferences and room occupancy, aids in optimizing energy usage without sacrificing comfort.

3.2.2 Equipment

The section outlines the key components and devices utilized in the development and implementation of the energy management system. These components form an integral part of the experimental setup and enable real-time monitoring, automation, and data analysis. The following equipment was employed:

Table 3-2 Equipment Involved for the Development of Energy Management System

Tool / Equipment	Description
Smart Socket	Smart socket served as the central monitoring and control device in the experimental setup. It provided real-time measurements of power, enabling accurate data collection for energy consumption analysis.
Microcontroller	A microcontroller was utilized to integrate and control the various components of the energy management system. It acted as the main controller for receiving data, executing algorithms, and sending commands to appliances for automation purposes.
IR Transmitter and Receiver	The IR transmitter and receiver were essential components in the air conditioner automation system. They facilitated wireless communication between the main controller and the air conditioner, enabling remote control and temperature adjustment.
Light Sensor and Passive Infrared (PIR) Sensor	The light sensor and PIR sensor were incorporated into the room lighting automation system. The light sensor detected ambient illuminance, while the PIR sensor detected human motion, allowing for intelligent lighting control based on occupancy and lighting requirements.
Node-RED and Node-Red Dashboard Visualization	Node-RED, a visual programming tool, was used to extract data from the smart socket Cloud API, process the energy consumption data, and feed it to the main microcontroller. Additionally, Node Red Dashboard visualization was employed to enhance energy awareness and control applications.

3.3 Experimental setup

In order to comprehensively test and evaluate the proposed IoT-based energy management system, a robust experimental setup was designed and implemented. This experimental setup consisted of a variety of components, each fulfilling a specific role in the system.

The heart of the experimental setup is the commercially available smart socket. This device was selected for its ability to provide real-time monitoring of the power, current, and voltage used by the attached appliances, in this case, an air conditioner and room lighting. This smart socket also allows for remote control of the connected devices, enabling automation and intelligent control, which is a core requirement for an effective demand-side management system. Among the many available smart sockets, the TP-Link Tapo P110 smart socket is distinguished by its better specifications.

The experimental setup was centered around the TP-Link Tapo P110 smart socket. This device was selected due to its remote-control capability, scheduling feature, timer function, and its compact design. Most importantly, it allowed for the collection of real-time data on power, which is a critical component for the demand-side management system being developed.

Figure 3.1 TP-Link Tapo P110 Smart Socket [66]

The specifications of the TP-Link Tapo P110 used in the experimental setup are presented in the table below:

Table 3-3 Specification of the TP-Link Tapo P110 [66]

Features	Specification
Network	IEEE 802.11b/g/n, Bluetooth 4.2
Wireless Type	2.4 GHz
System Requirements	Android 4.3 or higher, iOS 9.0 or higher
Dimensions	2.0* 2.8* 1.6 in (5172.040 mm)
Material	PC
Buttons	Power Button, Status LED
Power Requirements	AC 220-240 V~50/60 Hz 13 A
Maximum Load	2990 W, 13 A
Operating Temperature	0 ºC–35 ºC
Operating Humidity	10%–90%RH, Non-condensing

In conducting the project, the well-established model presented by Hussain Shareef et.al in his landmark paper, "Wireless Home Energy Management System with Smart Rule-Based Controller" was followed [21]. The objective was to validate and reinforce Dr. Shareef's pioneering contributions to Energy Management Systems, while primarily focusing on evolving these concepts within the context of the Internet of Things (IoT). This adaptation involved customizing the original methodology to meet the specific requirements of the component specifications under consideration. The vision remained, that is, to develop an innovative Energy Management System, augmented through the integration of IoT technologies.

3.3.1 Experiment 1: Development and Integration of an IoT-Based Real-Time Monitoring and Automation System for Air Conditioners and Room Lighting

Many of the commercial buildings have several air conditioners units and most of the rooms in the commercial building require a lighting system even when there are windows present. Insufficient natural light, daylight variability, task-specific lighting and energy efficiency are influenced the implementation of the lighting system.

Commercial buildings often employ centralized control systems for managing air conditioning and main area lighting [67]. These systems are crucial for maintaining occupant comfort and ensuring energy efficiency. Research has shown that integrated automation systems can effectively manage major loads in commercial buildings, including cooling, lighting, and plug loads, while maintaining occupant environmental preferences. This situation may be inefficient in terms of saving energy consumption. The tracking of the energy consumption is also based on the monthly usage of the electricity bill. Thus, this makes it hard for the user to keep track of energy consumption footprint. Also, some of the energy may be wasted from the overuse of these two systems. For example, if the room has high illumination because of the good lighting from the windows, the room lighting may not necessarily be turned on. The automation system can be implemented to control these two-power eater energy consumption. For this experiment, the development, and the integration of the IoT-based real-time monitoring and automation system for air conditioners and room lighting will be breakdown into several mini experiments which are:

- Development and integration of IOT-based real-time monitoring.
- Automation of air conditioner system
- Automation of room lighting System

3.3.1.1 Experiment 1 A: Development and Integration of an IoT-Based Real-Time Monitoring

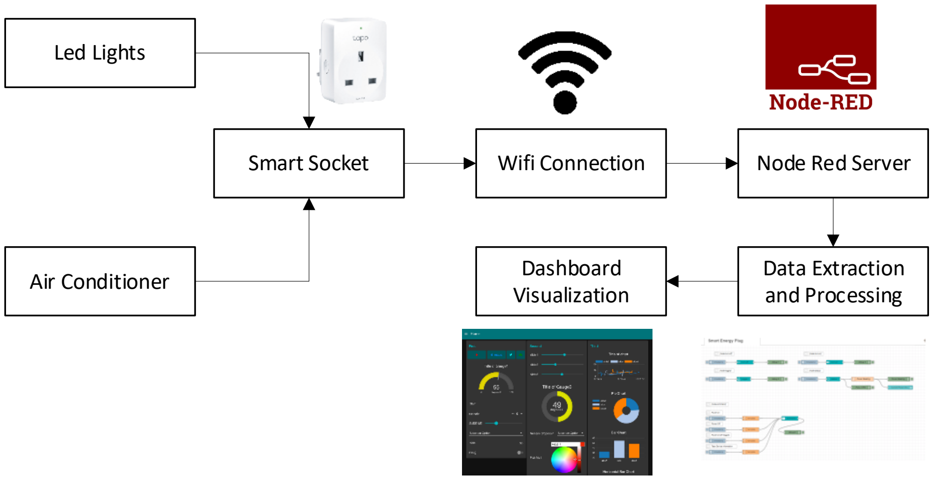


Figure 3.2 Block Diagram of the Real Monitoring of Energy Consumption

Figure 3.2 shows the solution of monitoring system of the energy consumption. The watt (W) energy will be the main parameter that will be monitored in real time. The P110 smart socket is employed to measure the actual energy consumption of both the lighting system and the air conditioner. In order to enhance accuracy, the lighting system exclusively utilizes LED lights, while the number of air conditioners measured is limited to a single unit. When the smart socket captures the energy consumption at that particular time, the Wi-Fi connection will be used to wirelessly send the data to their cloud Api which is Tp-Link cloud. Since the data is sent to their cloud, the data needs to be extracted from the cloud in order to get the energy consumption measurement. Thus, the Node-Red program will be the program that will get the energy data from the TP-Link Cloud Api data.

Node-RED, an invention of IBM, is a tool for visual programming that utilizes a flow-based model. It's specifically designed to interconnect hardware devices, APIs, and various online services in innovative ways [68]. Developed on Node.js, a runtime environment for JavaScript, it features a web-based editor for creating flows. This editor simplifies the process of combining different nodes from a comprehensive palette into functional flows. Once the energy in Watts has been extracted by the node-

Red, the data will be sent to the main microcontroller for the data feeding process to the algorithm of the Energy Management system and also the data will be presented in a graphical representation on the Node-Red dashboard. The Grafana Visualization might as well be considered to improve energy awareness and control applications [33].

3.3.1.2 Experiment 1 B: Automation of Air Conditioner System

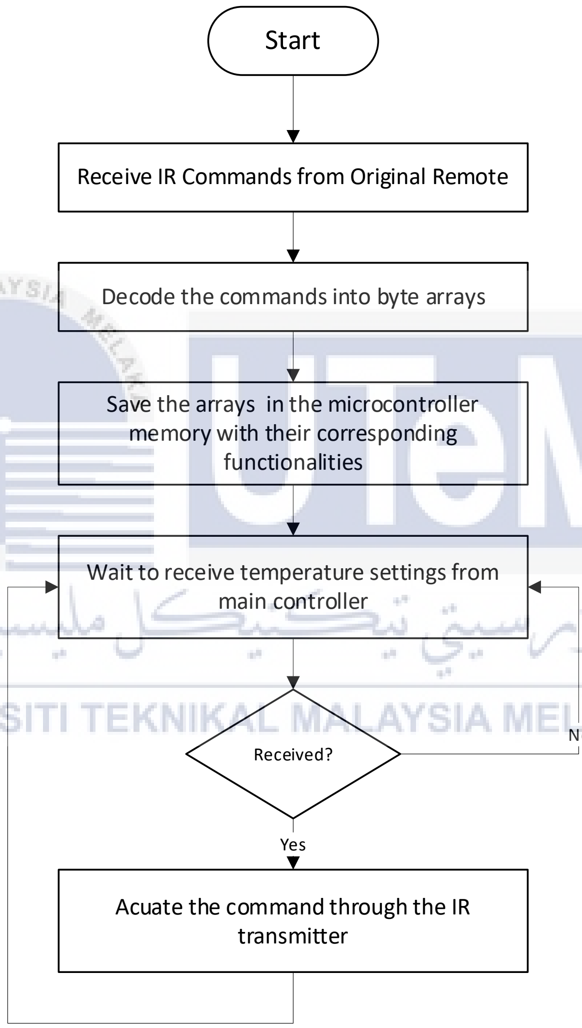


Figure 3.3 Operation of the IR Remote control of Air Conditioner [21]

This section contains automation that controls the temperature of an air conditioner. The air conditioner originally had a remote control that allows the user to manually adjust the modes and desired temperature. First, the air conditioner automation system's hardware must be incorporated with components such as an IR Transmitter, an IR Receiver, a microcontroller (ESP32 or Arduino), and a power supply. These integrated components will autonomously control the room's temperature based on settings made wirelessly by the user from either inside or outside the room. The controlling parameter is the Celsius (°C) temperature.

Figure 3.3 depicts the flow of the automation that controls the temperature level of the room in order to automate the air conditioner. The processes begin by identifying the original air conditioner remote controls by storing the commands within the microcontroller. The IR Receiver will capture the command from the remote control. The commands will be stored in a microcontroller array along with the corresponding functionality command (e.g., Fan Speed, Lowering or Raising Room Temperature, etc.). Once the command is saved, the energy management system's main controller will prepare to send an order to the air conditioner the user controls from the EMS system. The IR Transmitter will be used to actuate the commands received from the main controller.

3.3.1.3 Experiment 1 C: Automation of Room Lighting System

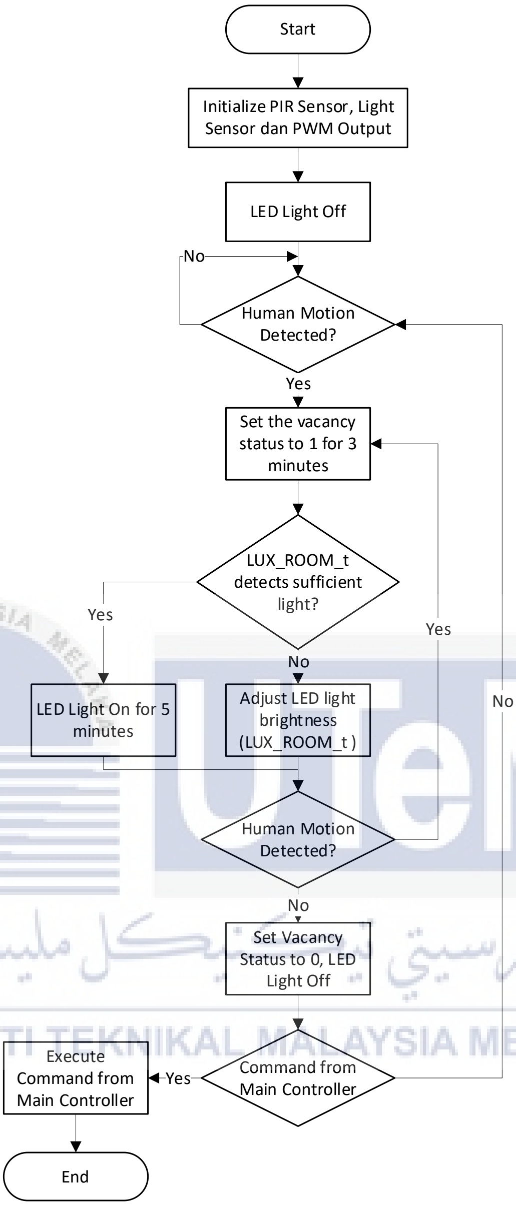


Figure 3.4 Operation of the Room Lighting Automation System

This experiment is designed to regulate a room's lighting system through automation. To ensure consistency and reduce the potential for variable interference, only LED bulbs are utilized throughout the experiment. Traditional lighting systems make use of wall-mounted switches to turn lights ON and OFF. However, in scenarios where a room's illumination is predominantly sourced from windows, the maximal brightness of an LED bulb is deemed unnecessary, leading to an avoidable waste of energy. Consequently, a modified LED bulb, capable of dimming and allowing brightness control, is introduced.

Energy wastage is also recognized in cases where lights remain on in unoccupied rooms. The operational structure of this automated system is described in Figure 3.4. A Passive Infrared (PIR) Sensor and a Light Sensor are deployed to detect environmental changes, producing the appropriate output to regulate room lighting accordingly.

A Pulse Width Modulation (PWM) output, generated from a microcontroller, controls the dimming process. The process commences by initializing the PIR Sensor, Light Sensor, and PWM Output. In the absence of human detection by the PIR Sensor, the LED light remains OFF. Upon detecting human movement, a variable parameter 'v', signifying room occupancy, transitions from 0 to 1 for a period of three minutes. This variable plays a significant role in Experiment 2.

The parameter LUX_ROOM_t records the value provided by the Light Sensor, measuring the room's illuminance. If the sensor detects sufficient light, the LED light is activated. If the room's illuminance is deemed insufficient, the PWM output modifies the bulb's brightness. As human beings typically move when required and stay still when not, the PIR sensor ceases to detect human motion after a span of three minutes. If no movement is detected after this interval, the variable 'v' reverts to 0 and the LED light is switched OFF.

The automated system continues to function in this manner until a stop command is issued by the main controller. At this point, the automation system ceases operations, allowing for the initiation of other processes.

3.3.2 Experiment 2: Implementation of Algorithm for Air Conditioner Energy Consumption and Room Lighting System (Fuzzy Logic Rule Based)

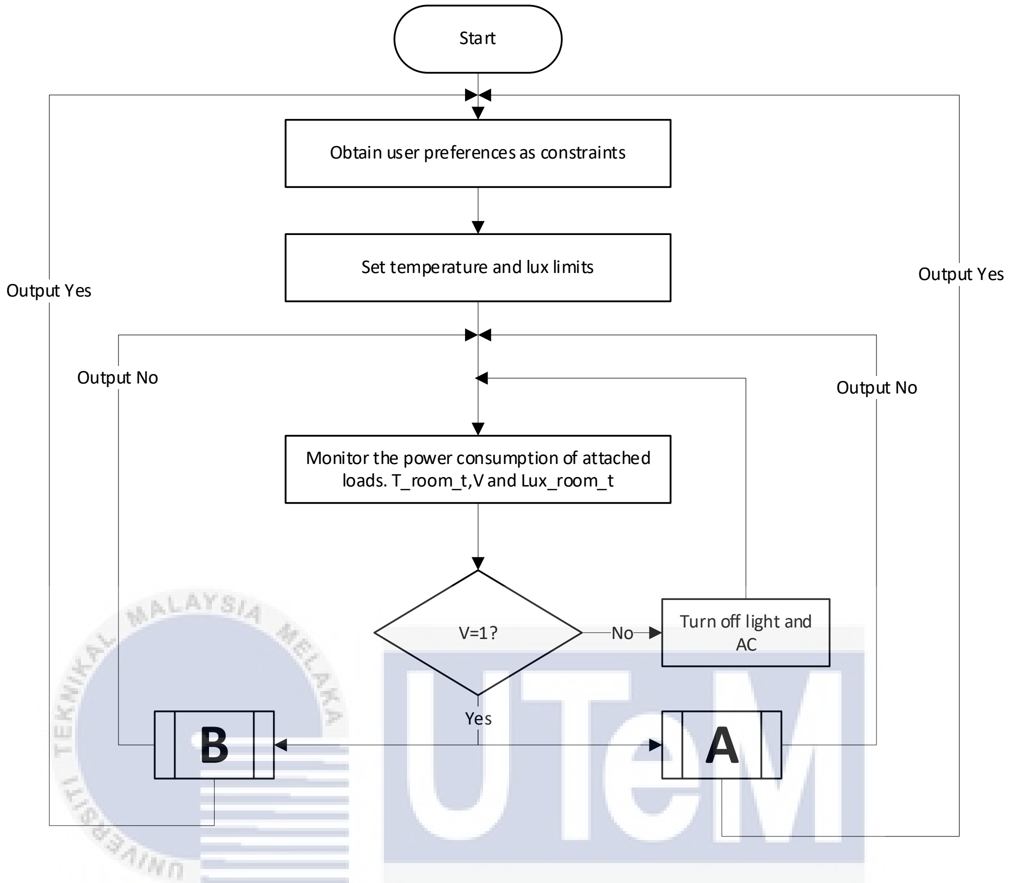


Figure 3.5 Flowchart of the Rule-Based Algorithm

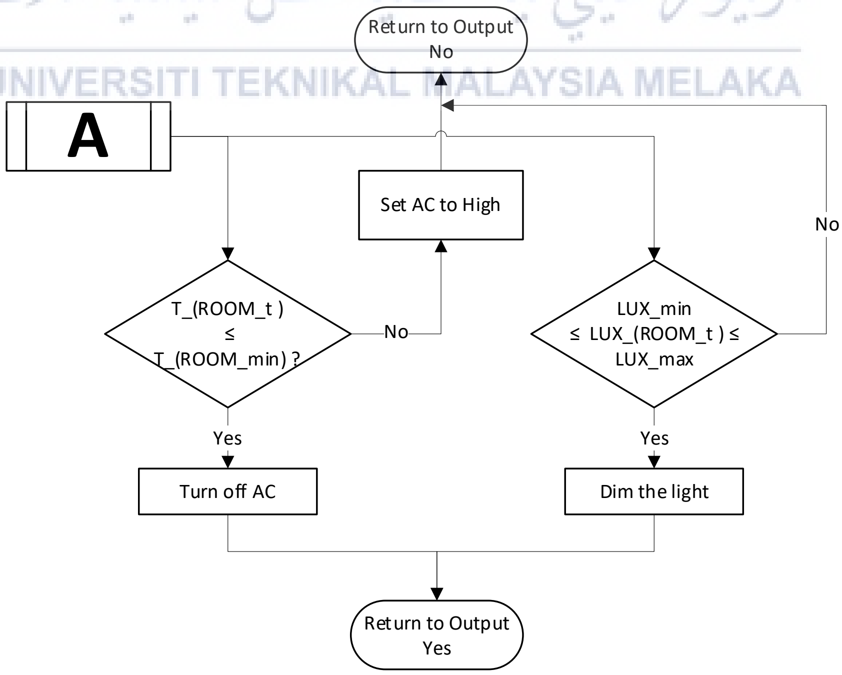


Figure 3.6 Flowchart A Subfunction Process

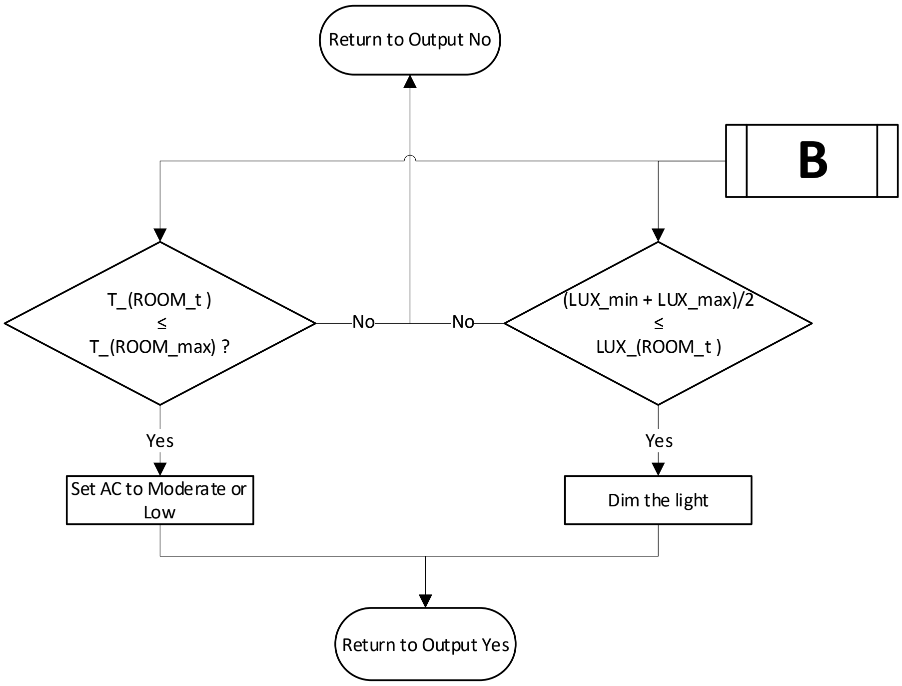


Figure 3.7 Flowchart B Subfunction process

Figure 3.5,Figure 3.6 and Figure 3.7 are modified versions of Energy Management System in [21]. The system harnesses real-time data from room condition monitoring circuits, which are instrumental in tracking variations in key environmental parameters such as illuminance, room temperature and vacancy. [21]emphasized using the rule-based algorithm to reduce the energy consumption where all the rules are listed according to the preference of the user on how they wanted to use their electrical appliances without affecting the performance. Crucially, this system also incorporates the measurement of energy consumption, thus offering a holistic view of the energy efficiency landscape within a given environment.

This IOT based system for energy management system adapts Fuzzy Logic Rule-Based (FLRB) approach. Unlike conventional binary logic systems, the FLRB approach manages the inherent uncertainty and variability in environmental parameters. For instance, room temperature and illuminance do not adhere to binary states but rather exist on a continuum. The FLRB, with its ability to process varying degrees of truth, aligns seamlessly with this reality, thereby enhancing decision-making accuracy. The modified flowchart will be the framework to construct the fuzzy logic base rule algorithm.

The system integrates three key inputs, that are temperature, vacancy (alternatively known as occupancy), and light intensity. The outputs, derived through a fuzzy logic-based decision process, manage two systems,: the air conditioning and the lighting. The fuzzy logic's rule base is initially established not based on customer preferences, but as a test configuration to evaluate daily energy consumption. This test setup considers hypothetical user preferences, including desired light intensity, preferred temperature settings, and specific requirements for a confined space. These test preferences are then utilized to construct the fuzzy sets for each category, enabling the examination of the fuzzy logic system's effectiveness in a controlled environment.

Table 3-4 Fuzzy Logic Input and User Preference

Inputs	Description (Linguistic Variables)
Temperature	Continuous range from 15 to 31 degrees Celsius. Categorized into three sets: • Low (15 to 22 °C) • Moderate (21 to 27 °C) • High (26 to 31 °C)
Light Intensity	Measured in Digital Units ranging from 0 (brightest) to 4095 (darkest). Categorized into three sets: • Low (3000 to 4095) • Moderate (1500 to 3500) • High (0 to 2000)
Vacancy	Binary Input indicating room occupancy. • Vacant • Occupy

For the light intensity, the decision to use digital units instead of lumen or lux for measuring light intensity with an LDR (Light-Dependent Resistor) is primarily due to the inherent characteristics of LDRs and the practical challenges associated with calibrating them for accurate light measurements. First is LDR is known for the nonlinear response to light intensity. Also, to accurately measure light intensity in lux or lumens using an LDR, a detailed calibration process is necessary. This involves using a known light source and precisely measuring the LDR's response to different light levels. Given that LDRs can vary significantly in their characteristics. For this project, having the understanding of light intensity whether it is low, moderate, or high is sufficient. Also, the use of digital units provides a simple and effective means to monitor and respond to changes in light condition without the need for calibrated lux or lumen measurement.

Table 3-5 Fuzzy Logic Output and User Preference

Output	Description (Linguistic Variables)
Air Conditioning	Output Ranging from 0 to 5: • Off (0 to 1) • Low (1 to 3) • High (3 to 5)
Lighting	Output Range from 0 to 5, classified into: • Low (0 to 1) • Dim (1 to 3) • Bright (3 to 5)

Based on Table 3-5, the air conditioning output ranges from 0 to 5 and is categorized into 3 states which are “Off”, “Low” and “High”. When the air conditioning is in “Off” state, the system is turned off. There is no cooling action taking place, and the system remains inactive. This state is typically selected when the room is either adequately cool or not in use. When the system is in the “Low” state, it operates at a reduced capacity. This means that the air conditioning will not be intensely used, and the air temperature will not be as low as in the “High” state. It provides a moderate cooling effect, suitable for maintaining a comfortable environment without excessive energy consumption. This state is ideal for conditions where a mild cooling effect is sufficient or where energy efficiency is a priority. For “High” state, the air conditioning operates at full power. This leads to more intense cooling and a lower air temperature. The system works at its maximum capacity to provide significant cooling, which is particularly useful in situations where the room temperature is considerably high, or rapid cooling is desired. However, this state also corresponds to higher energy consumption. The lighting system also has a range from 0 to 5. When the Lighting is in “Off” state, The lighting is completely turned off. This is typically used when the room is not in use or sufficient natural light is available. When the lighting is set to dim level (“Dim State”), it provides moderate illumination. This setting is suitable for when full brightness is not necessary. Lastly, when the lighting system is in “Bright” state, the lighting is at its brightest level. It is used when maximum illumination is required, such as for work or reading purposes. The Fuzzy Logic rules are then tailored to satisfy the input as well as the output.

Table 3-6 Fuzzy Logic Rules for Air Conditioning System

No	Rules Condition
1	If temperature is **high** and vacancy is **occupied**, then air conditioning is **high**.
2	If temperature is **moderate** and vacancy is **occupied**, then air conditioning is **low**.
3	If temperature is **low**, light intensity is **low**, and vacancy is **occupied**, then air conditioning is **off**.
4	If temperature is **low**, light intensity is **moderate**, and vacancy is **occupied**, then air conditioning is **off**.
5	If temperature is **low**, light intensity is **high**, and vacancy is **occupied**, then air conditioning is **off**.
6	If temperature is **moderate**, light intensity is **high**, and vacancy is **occupied**, then air conditioning is **low**.
7	If temperature is **high**, light intensity is **moderate**, and vacancy is **occupied**, then air conditioning is **high**.
8	If temperature is **high**, light intensity is **high**, and vacancy is **occupied**, then air conditioning is **high**.
9	If vacancy is **vacant**, temperature is **low**, and light intensity is **low**, then air conditioning is **off**.
10	If vacancy is **vacant**, temperature is **moderate**, and light intensity is **low**, then air conditioning is **off**.
11	If vacancy is **vacant**, temperature is **high**, and light intensity is **low**, then air conditioning is **off**.
12	If vacancy is **vacant**, temperature is **low**, and light intensity is **moderate**, then air conditioning is **off**.
13	If vacancy is **vacant**, temperature is **moderate**, and light intensity is **moderate**, then air conditioning is **off**.
14	If vacancy is **vacant**, temperature is **high**, and light intensity is **moderate**, then air conditioning is **off**.
15	If vacancy is **vacant**, temperature is **low**, and light intensity is **high**, then air conditioning is **off**.
16	If vacancy is **vacant**, temperature is **moderate**, and light intensity is high, then air conditioning is **off**.
17	If vacancy is **vacant**, temperature is **high**, and light intensity is **high**, then air conditioning is **off**.
18	If temperature is **moderate**, light intensity is **moderate**, and vacancy is **occupied**, then air conditioning is **low**.
19	If temperature is **moderate**, light intensity is **low**, and vacancy is **occupied**, then air conditioning is **low**.
20	If temperature is **high**, light intensity is **low**, and vacancy is **occupied**, then air conditioning is **high**.

Table 3-7 Fuzzy Logic Rules for Lighting System

No	Rules Condition
1	If temperature is **low**, light intensity is **low**, and vacancy is **occupied**, then lighting is **bright**.
2	If temperature is **moderate**, light intensity is **low**, and vacancy is **occupied**, then lighting is **bright**.
3	If temperature is **high**, light intensity is **low**, and vacancy is **occupied**, then lighting is **bright**.
4	If temperature is **low**, light intensity is **moderate**, and vacancy is **occupied**, then lighting is **dim**.
5	If temperature is **moderate**, light intensity is **moderate**, and vacancy is **occupied**, then lighting is **dim**.
6	If temperature is **high**, light intensity is **moderate**, and vacancy is **occupied**, then lighting is **dim**.
7	If temperature is **low**, light intensity is **high**, and vacancy is **occupied**, then lighting is **off**.
8	If temperature is **moderate**, light intensity is **high**, and vacancy is **occupied**, then lighting is **off**.
9	If temperature is **high**, light intensity is **high**, and vacancy is **occupied**, then lighting is **off**.
10	If temperature is **low**, light intensity is **low**, and vacancy is **vacant**, then lighting is **off**.
11	If temperature is **moderate**, light intensity is **low**, and vacancy is **vacant**, then lighting is **off**.
12	If temperature is **high**, light intensity is **low**, and vacancy is **vacant**, then lighting is **off**.
13	If temperature is **low**, light intensity is **moderate**, and vacancy is **vacant**, then lighting is **off**.
14	If temperature is **moderate**, light intensity is **moderate**, and vacancy is **vacant**, then lighting is **off**.
15	If temperature is **high**, light intensity is **moderate**, and vacancy is **vacant**, then lighting is **off**.
16	If temperature is **low**, light intensity is **high**, and vacancy is **vacant**, then lighting is **off**.
17	If temperature is **moderate**, light intensity is **high**, and vacancy is **vacant**, then lighting is **off**.
18	If temperature is **high**, light intensity is **high**, and vacancy is **vacant**, then lighting is **off**.

The EMS employs Mamdani fuzzy rule-based approach to control the output of two systems: air conditioning and lighting. The table below shows the process of how the Fuzzy Rules Algorithm will work.

Process	Process Explained
Fuzzification of Inputs	• Inputs such as temperature, light intensity, and vacancy are first converted into fuzzy values using membership functions. These functions categorize each input into fuzzy sets (like 'low', 'moderate', 'high' for temperature).
Applying Mamdani Fuzzy Rules	• The system then applies the Mamdani fuzzy rules[69]. These rules are "IF-THEN" statements that dictate how the output should be adjusted based on the input conditions. • For example, a rule like "IF temperature is high AND vacancy is occupied, THEN set air conditioning to high" dictates that when the room is occupied and the temperature is high, the air conditioning should work at a high setting.
Inference	• The inference engine processes these rules collectively and determines what the outputs should be in fuzzy terms. It considers all the relevant rules based on the current input conditions and combines their effects.
Defuzzification	• Finally, the fuzzy output for air conditioning and lighting is converted into a crisp value. This is the actual, actionable output that the system will implement. For instance, it will determine the exact setting level for the air conditioning and lighting.

During the implementation of the Fuzzy Logic Rule-based system, fine-tuning is carried out to ensure that the rule conditions align with user preferences according to the specified settings. The Fuzzy Logic Rule Based system is implemented in Python Environment and connected to the central hardware where the computation of the algorithm will take place. Raspberry Pi is the main choice for the system as it is as powerful as a laptop. Due to its small and compact design, it is ideal for this project. The Wi-Fi capabilities offered by the Raspberry Pi are essential for IoT applications and for updating fuzzy logic parameters or algorithms remotely.

3.3.3 Experiment 3: Analysis the Efficiency of Algorithm Output for the Developed Energy Management System

Experiment 3 in the project is meticulously designed to analyze the efficiency of a Fuzzy Logic Rule-Based of Energy Management System (EMS). This experiment is structured to provide a comprehensive understanding of how the EMS algorithm behaves in a real-world scenario and its potential impact on energy consumption patterns. The first phase of the experiment involves a detailed 24-hour data capture of energy consumption under normal conditions. During this period, crucial data parameters such as room vacancy (occupancy status), light intensity, temperature, and the energy consumption of various appliances are meticulously recorded. This phase is vital as it establishes a baseline of typical energy usage and power consumption patterns in the absence of the EMS. This data capture provides a realistic view of how appliances typically consume power in a standard environment, serving as a critical reference point for later stages of the experiment.

Following the initial data capture, the experiment transitions into a simulation phase. In this phase, the raw data collected - encompassing temperature, light intensity, vacancy, and energy consumption - are used as inputs in a Python-based simulation environment. This is where the core of the experiment takes place. The Fuzzy Logic Rule-Based EMS algorithm is applied to these inputs, replicating the conditions of the 24-hour monitoring period. The focus of this simulation is to observe and analyze the behavior of the fuzzy logic system throughout the same 24-hour period. This involves assessing how the system interprets and responds to changes in the environmental conditions and how these responses impact the overall energy consumption.

The simulation results are then thoroughly analyzed to gauge the effectiveness of the Fuzzy Logic Rule-Based EMS. By comparing the simulated energy consumption with the actual data captured during the initial monitoring, the experiment aims to ascertain the potential of the EMS in optimizing energy usage. This comparative analysis is critical as it highlights the efficiency of the EMS in real-world settings and its responsiveness to varying environmental conditions.

The ultimate goal of this experiment is to evaluate whether the implementation of a Fuzzy Logic Rule-Based system can lead to more efficient energy management practices. It seeks to determine if such a system can significantly reduce energy waste and optimize energy usage, thereby contributing to more sustainable energy consumption patterns. Through this experiment, the project aims to provide a clearer understanding of the practical applicability and benefits of employing a Fuzzy Logic Rule-Based EMS in real-world scenarios, bridging the gap between theoretical algorithms and their actual impact on energy management.

For the simulation, it is important to recognize that energy consumption varies depending on the settings of the lighting and air conditioning systems. Therefore, before commencing the simulation, it is crucial to understand these varying power values. The first step in the simulation process involves mapping the outputs of the fuzzy logic system to specific energy consumption values. This mapping correlates the brightness levels of lighting and the operational modes of the air conditioning system to their respective energy consumption figures, which have been predetermined. By doing so, we can accurately simulate and estimate the energy usage based on the different settings of these systems.

Table 3-8 Energy Usage of Lighting System based on Brightness.

<table>
<tr><th colspan="3">For Lighting</th></tr>
<tr><th>Brightness
(%)</th><th>Power
(W)</th><th>Mean
(W)</th></tr>
<tr><td rowspan="2">100</td><td>12.9000</td><td rowspan="2">12.9123</td></tr>
<tr><td>12.9170</td></tr>
</table>

	12.9200	
75	10.1500	10.2527
	10.4830	
	10.1250	
50	7.2500	7.2777
	7.2840	
	7.2990	
30	5.0000	5.0050
	5.0660	
	4.9490	
10	2.8390	2.7633
	2.7320	
	2.7190	
Off	1.3290	1.3743
	1.3690	
	1.4250	

Table 3-9 Energy Usage of Air Conditioning based on Brightness.

For Air Conditioner				
Mode	Power (W)		Mean(W)	
	Cold Mode OFF	Cold Mode On	Cold Mode OFF	Cold Mode On
High	59.1170	62.5050	59.0827	62.6360
	59.1220	62.6460		
	59.0090	62.7570		
Med	53.6800	57.7330	53.7267	57.4893
	53.7850	57.5860		
	53.7150	57.1490		
Low	47.2330	51.0200	47.6710	51.1773
	48.1480	50.7690		
	47.6320	51.7430		
Off	0.7190		0.6887	

	0.6670	
	0.6800	

In order to ensure the accuracy and reliability of the simulation, the energy consumption for each mode of the air conditioning and lighting systems was meticulously recorded three times, with a 10-second interval between each measurement. This methodical approach was employed to ascertain the average power consumption for each setting, thereby providing a robust and representative dataset for the simulation.

The energy consumption values for different brightness levels of the lighting system and operational modes of the air conditioning system are detailed in Table 3-8 and Table 3-9. These tables present the power usage in watts for each setting, providing an average figure that encapsulates the typical energy consumption for each state. For instance, the lighting system's energy usage varies from a low of 1.3743W when off to a high of 12.9123W at 100% brightness. Similarly, the air conditioning system's energy consumption ranges from 0.6887W when off to 62.6360W in high mode with the cold feature activated.

With this comprehensive dataset, functions were developed in Python to translate the fuzzy logic system's output values into corresponding energy consumption figures. These functions play a pivotal role in the simulation, allowing for a dynamic and accurate estimation of energy usage based on the varying settings of the lighting and air conditioning systems. By employing this approach, the simulation can effectively mirror the real-world energy consumption patterns, facilitating a more accurate assessment of the fuzzy logic system's efficiency in managing energy usage. This method not only provides a quantitative measure of energy savings but also highlights the potential of fuzzy logic in optimizing energy consumption in various operational scenarios. Then, the energy consumption value is taken to create functions in Python to take the output value from the fuzzy logic system and return the corresponding energy consumption values.

```python
import pandas as pd

# Define the functions for calculating energy consumption
def lighting_energy_consumption(brightness):
    if 0 <= brightness < 0.5:
        return 1.3743  # Off
    elif 0.5 <= brightness < 1:
        return 2.7633  # 10%
    elif 1 <= brightness < 2:
        return 5.0050  # 30%
    elif 2 <= brightness < 3:
        return 7.2777  # 50%
    elif 3 <= brightness < 4:
        return 10.2527  # 75%
    else:
        return 12.9123  # 100%

def air_conditioning_energy_consumption(mode):
    if 0 <= mode < 1:
        return 0.6887  # Off
    elif 1 <= mode < (5/3):
        return 47.6710  # Low
    elif (5/3) <= mode < (7/3):
        return 51.1773  # Low (Cold Mode On)
    elif (7/3) <= mode < 3:
        return 53.7267  # Med
    elif 3 <= mode < (11/3):
        return 57.4893  # Med (Cold Mode On)
    elif (11/3) <= mode < (13/3):
        return 59.0827  # High
    else:
        return 62.6360  # High (Cold Mode On)
```

Figure 3.8 Assigned Energy Consumption Values Corresponding to Each Range of Fuzzy Logic Output

3.4 Limitation of the proposed methodology

The proposed methodology for developing an IoT-based energy management system has several limitations that should be taken into consideration. Firstly, the scope and generalizability of the methodology are limited. It focuses specifically on the development of an energy management system for air conditioners and room lighting. While the methodology proves effective within this specific context, its applicability to other types of energy management systems or different appliances may be constrained.

The experimental setup used in the methodology is another factor that introduces limitations. The setup employed in the experiments may not fully replicate real-world conditions. Factors such as room size, environmental conditions, and the specific models of appliances and equipment utilized may differ in practical applications. These variations could potentially impact the performance and outcomes of the energy management system when deployed in real-world scenarios. Therefore, the results obtained from the experiments should be interpreted with caution and validated in diverse settings to assess their generalizability.

The selection of the algorithm for energy consumption management is also a limitation of the methodology. While a rule-based algorithm has been chosen based on previous research, there may be alternative algorithms or optimization techniques that could yield different results. The choice of algorithm introduces a subjective element into the methodology and may affect the overall performance of the energy management system. Further research and exploration of alternative algorithms could enhance the system's efficiency and effectiveness.

Data accuracy and reliability are critical aspects to consider when interpreting the results of the methodology. The accuracy and reliability of the data collected during the experiments are essential for drawing valid conclusions. However, limitations in data collection methods, sensor precision, and potential measurement errors may impact the reliability and accuracy of the collected data. These limitations can introduce uncertainties and potential biases into the analysis and subsequent refinement of the system. It is crucial to address these challenges and ensure the data used for analysis is as accurate and reliable as possible.

Practical implementation challenges pose additional limitations to the proposed methodology. While the methodology demonstrates success within a controlled experimental environment, real-world implementation may present practical challenges. Interoperability with existing infrastructure, compatibility with different appliances and building systems, and user acceptance are factors that need to be considered. These challenges may affect the feasibility and effectiveness of deploying the energy management system on a larger scale. A comprehensive evaluation of these practical implementation challenges is necessary to ensure the system's successful integration into real-world settings.

Lastly, cost and complexity are important limitations to consider. Implementing an IoT-based energy management system can involve significant costs associated with acquiring and integrating the necessary equipment and technologies. Additionally, the complexity of the system and its associated algorithms may require expertise in IoT, data analysis, and programming. These factors can limit the accessibility and affordability of implementing the proposed methodology in certain contexts. Cost-effective solutions and simplified implementation approaches should be explored to make the system more feasible and widely applicable.

3.5 Summary

The development of an IoT-based Energy Management System (EMS) with a focus on demand-side management for air conditioners and lighting is structured into three distinct experiments, each contributing uniquely to the system's overall functionality.

Experiment 1 centers on the development and integration of a real-time monitoring and automation system. A commercially available smart socket is used to capture real-time power consumption data, laying the groundwork for effective demand-side management. This initial phase establishes a baseline of energy usage for air conditioners and lighting systems under typical conditions.

Experiment 2 involves the design and implementation of an intelligent algorithm for energy consumption management. This phase explores machine learning or optimization techniques to refine the EMS's decision-making process. The goal is to enhance the system's ability to regulate energy use efficiently, balancing consumption with user needs.

Experiment 3 is dedicated to analyzing and enhancing the system's performance based on the algorithm's output. This involves a detailed 24-hour energy consumption capture to establish a baseline, followed by a simulation phase in a Python environment. The simulation uses the collected data to replicate real-world conditions and applies the Fuzzy Logic Rule-Based EMS algorithm to assess its behavior and impact on energy consumption. The focus here is on comparing the simulated energy consumption patterns against the actual data to evaluate the EMS's efficiency in optimizing energy usage.

However, the methodology has certain limitations. It is limited in scope to air conditioners and room lighting, and its applicability to other energy management systems or appliances may be constrained. The experimental setup may not fully replicate real-world conditions, impacting the system's performance in practical applications. The choice of the algorithm is subjective, and alternative algorithms could yield different results. Data accuracy and reliability, as well as practical implementation challenges, should be considered. The cost and complexity of implementing the system may also limit its feasibility in certain contexts.

Key parameters such as power consumption, illuminance, and room temperature are carefully considered in the system's development. Equipment like smart sockets, microcontrollers, IR transmitters and receivers, light and PIR sensors, and Node-RED for data visualization play crucial roles in the experimental setup. This experimental setup involves various tests to evaluate the system’s performance, focusing on energy consumption, comfort levels, and response times, guiding an iterative design process for an enhanced energy management solution.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Results and Discussions Introduction

This section delves into the results of the study on the IoT-based Energy Management System (EMS), with a particular focus on air conditioners and lighting. Several experiments were conducted to evaluate the effectiveness of the EMS, especially its Fuzzy Logic Rule-Based approach in controlling energy usage. The findings from these experiments are examined to gauge how well the EMS conserves energy and manages the operation of air conditioners and lights. The discussion includes an analysis of how the EMS influences energy usage patterns, the system's accuracy, and the implications for smarter and more sustainable energy use. This analysis is crucial for appreciating the advantages of smart energy management systems and their contribution to energy conservation.

4.2 Experimental Results

4.2.1 Development and Integration of an IoT-Based Real-Time Monitoring and Automation System for Air Conditioners and Room Lighting

The development and integration of an IoT-Based Real-Time Monitoring and Automation System for air conditioners and room lighting marks a significant advancement in the field of energy management. This section provides an in-depth exploration of the system's architecture, designed to enable seamless interaction between various components for optimal control of environmental conditions within a room.

Breaking down the system's setup into focused sections, first the development of the User Interface using Node-Red, a visual tool for wiring together hardware devices, APIs, and online services. This interface is pivotal in providing users with an engaging and easy-to-navigate control panel for managing the room's temperature and lighting settings. Following this, the implementation of specific subsystems, the Air Conditioner Automation System, and the Room Lighting Automation System. Each of these subsystems is engineered to not only respond intelligently to environmental changes and occupancy but also to allow for user customization, thereby striking a balance between automated efficiency and personal preference.

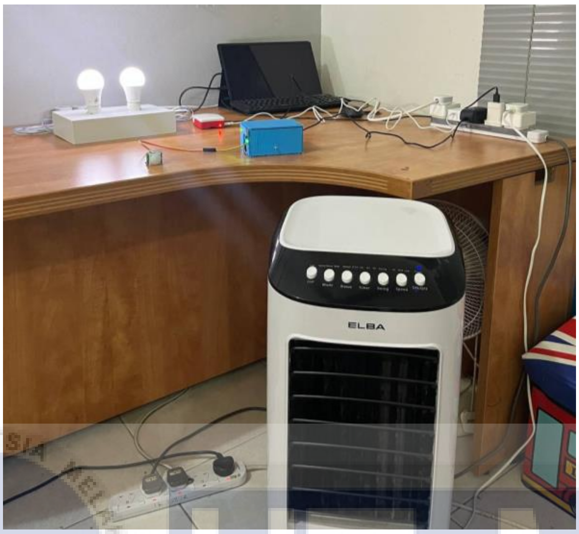


Figure 4.1 Energy Management System Setup

4.2.1.1 Development and Integration of an IoT-Based Real-Time Monitoring

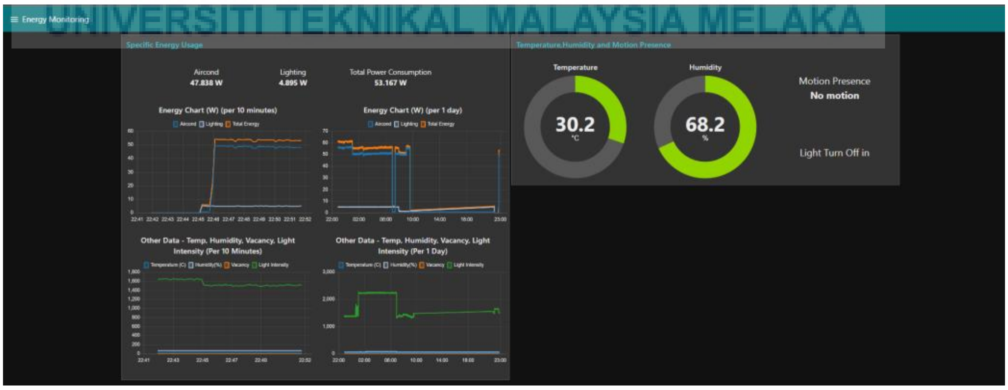


Figure 4.2 Node-Red Dashboard User Interface Tab 1 – Energy Monitoring

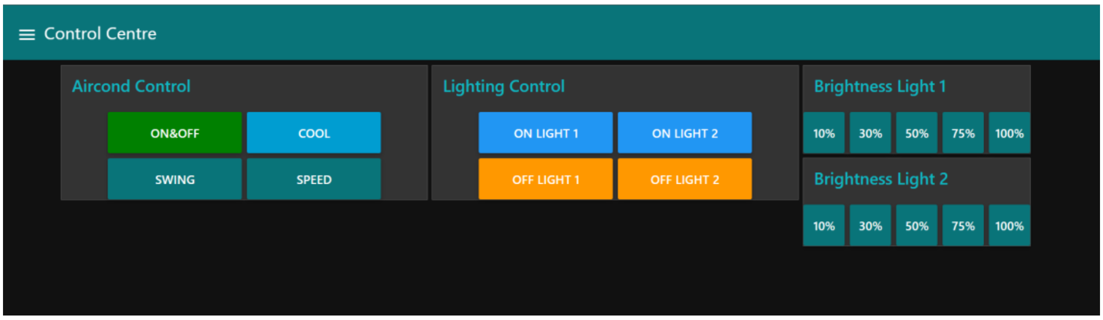


Figure 4.3 Node-Red Dashboard User Interface Tab 2 – Control Centre

Figure 4.4 Node-Red Dashboard User Interface Tab 3 – Control Centre (Main Power)

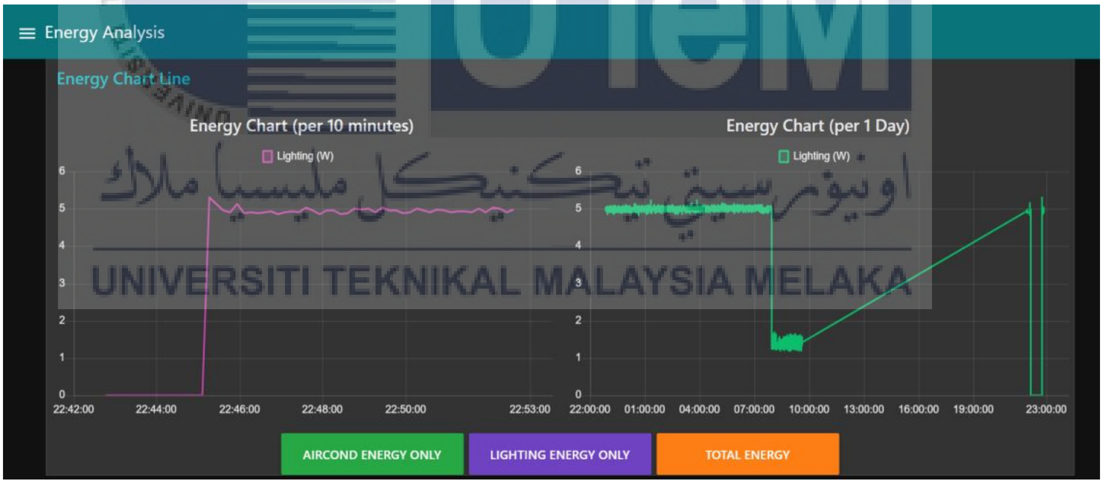


Figure 4.5 Node-Red Dashboard User Interface Tab 4 – Energy Analysis

Figure 4.2,Figure 4.3,Figure 4.4 and Figure 4.5 showcase the User Interface developed using the Node-Red Dashboard. The interface is organized into four distinct tabs to prevent cluttering of information and controls in a single space. The first tab, the 'Energy Monitoring Tab,' offers insights into the energy consumption of both the Air Conditioning and Lighting systems. It also displays information on Temperature, Vacancy, and Light Intensity. This tab features a linguistic vacancy indicator which promptly shows the room's occupancy status. Additionally, there is a countdown timer that informs users when the lights will turn off if no motion is detected within 3 minutes. More detailed visuals of the UI can be found in the appendix section.

The second tab, named 'Control Section,' is designed for manual adjustments of the air conditioner and lighting. Users can easily operate the air conditioning using the provided buttons. As the lighting brightness is adjustable, there are five brightness levels to choose from, catering to user preferences. The third tab, also a part of the Control Centre, primarily focuses on Main Power. This refers to the Smart Socket, which is the primary gateway for electricity from the grid to the appliances like the air conditioner and lighting system. Users can shut off the main electricity source using a toggle button in this tab.

The fourth tab is dedicated to energy analysis. Here, users can alter the display of the graph plots through various options. They can switch from viewing three plots simultaneously to focusing on a single plot for a more detailed analysis of energy usage. The interface also allows for the comparison of two different graph plots by selecting the respective buttons.

The results demonstrate that the user interface significantly enhances user awareness of energy consumption. It also enables wireless control of the air conditioner and lighting systems, eliminating the need to physically interact with appliance switches or remotes.

4.2.1.2 Air Conditioner Automation System

Figure 4.6 IR Transmitter

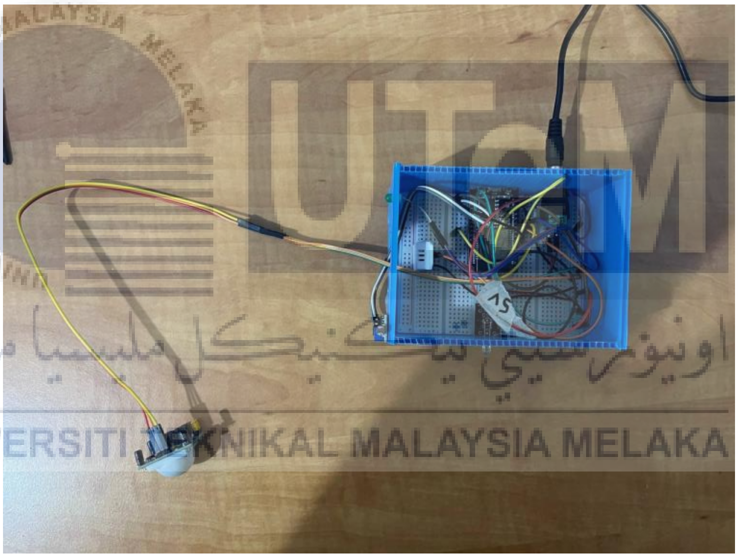


Figure 4.7 Hardware Connection ESP 32 with several sensors

The Air Conditioning System is operated through an ESP 32 microcontroller, which stores all the remote-control commands for the air conditioner. Upon receiving a command from Node Red, the IR Transmitter wirelessly transmits the signal to activate or modify the air conditioner's mode. Additionally, the system integrates a PIR Sensor, along with temperature and light intensity sensors (LDR), all of which are housed within a dedicated box. These sensors are essential for capturing ambient temperature and light intensity data, enabling the system to make informed decisions about air conditioning adjustments based on the current room conditions.

4.2.1.3 Room Lighting Automation System

Figure 4.8 Lighting System Setup

The Lighting System depicted in the figure demonstrates successful illumination and dimming capabilities of the bulbs, controlled via the Node Red Dashboard. At its brightest setting, the system draws a maximum power of 13W for both bulbs, while at its dimmest (10% brightness), the power consumption is minimized to approximately 2.76W. This system is adept at providing artificial lighting in rooms or confined spaces, particularly useful when natural light is insufficient. Furthermore, the Lighting System offers the flexibility of manual adjustments. Users can wirelessly control the brightness levels and switch the bulbs on or off through the Node Red Dashboard User Interface, ensuring convenience and user-centric operation.

In essence, this development represents a significant enhancement in lighting control, offering both energy efficiency and user-friendly interaction. The ability to adjust lighting conditions in response to the room's ambiance not only contributes to energy savings but also enhances the overall comfort of the space.

4.2.2 Implementation of Algorithm for Air Conditioner Energy Consumption and Room Lighting System

Figure 4.9 Successful Computation and Transmission of Data by the Fuzzy Logic Rule-Based Algorithm to Node-Red

Figure 4.9 illustrates the successful computation performed on the Raspberry Pi, where the output from the fuzzy logic algorithm is transferred to Node-Red. Given Node-Red's adoption of flow-based visual programming, the detailed flow and nodes are presented in the Appendix section. Additionally, the Figure 4.10 includes a specific node flow that displays how the fuzzy logic output controls the air conditioning settings.

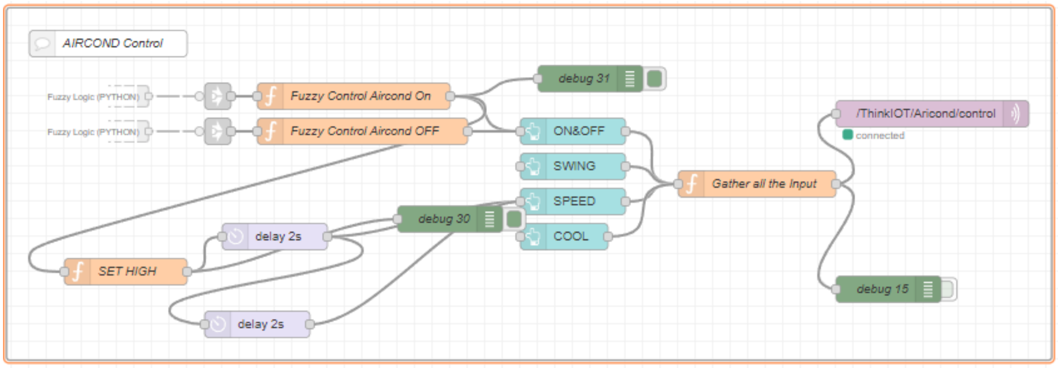


Figure 4.10 The Fuzzy Logic Output Goes to Function Node, to the Input Air Conditioner Button Settings

During the development of the fuzzy logic algorithm using Python coding, multiple tests were conducted to observe the behavior of the fuzzy rules when specific values for primary inputs were inputted. The first test involved examining the shapes of the fuzzy sets. The chosen shapes for the fuzzy sets were Triangular and Trapezoidal. While Singleton Membership Functions were also considered, it could not be directly implemented in Python as the library lacked the necessary module. Therefore, sigmoid functions were utilized as an alternative to represent the singletons. After implementing and fine-tuning these shapes, Figure 4.11, Figure 4.12, Figure 4.13, Figure 4.14 and Figure 4.15 show the membership function that successful met the conditions outlined by the user preferences.

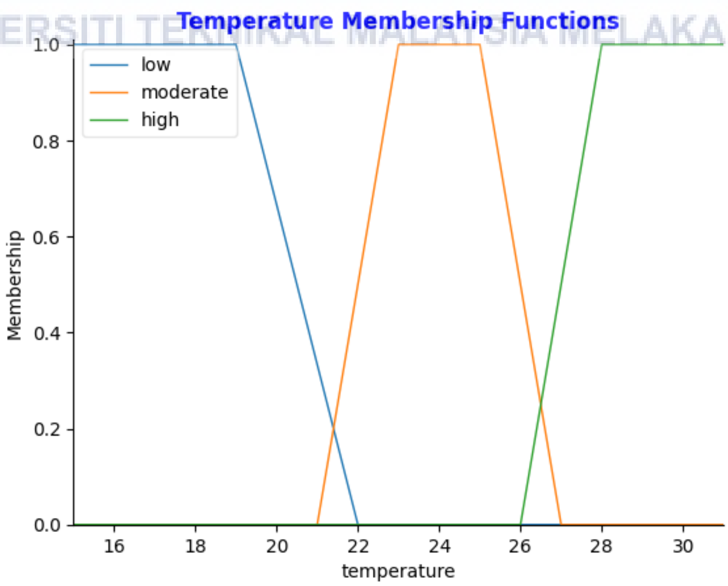


Figure 4.11 Temperature Membership Functions Fuzzy Sets - Input

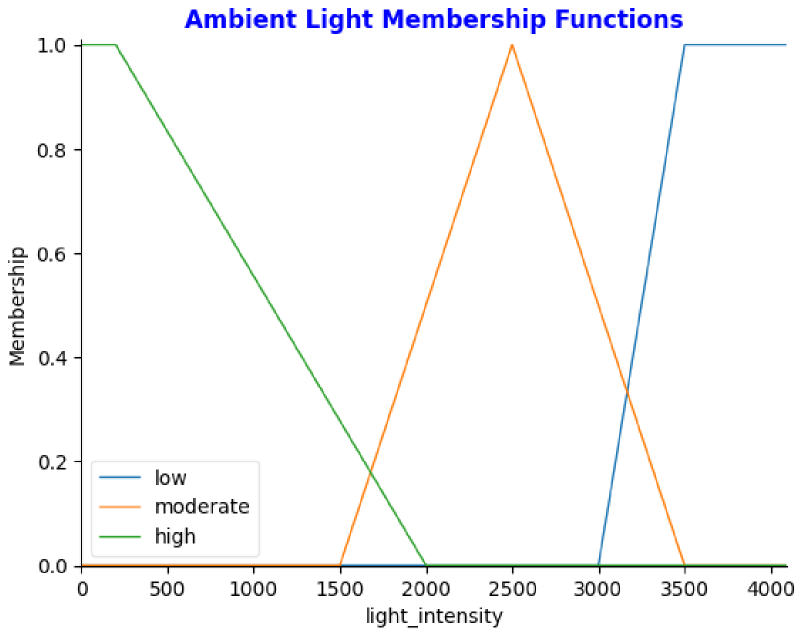


Figure 4.12 Ambient Light Membership Function Fuzzy Sets - Input

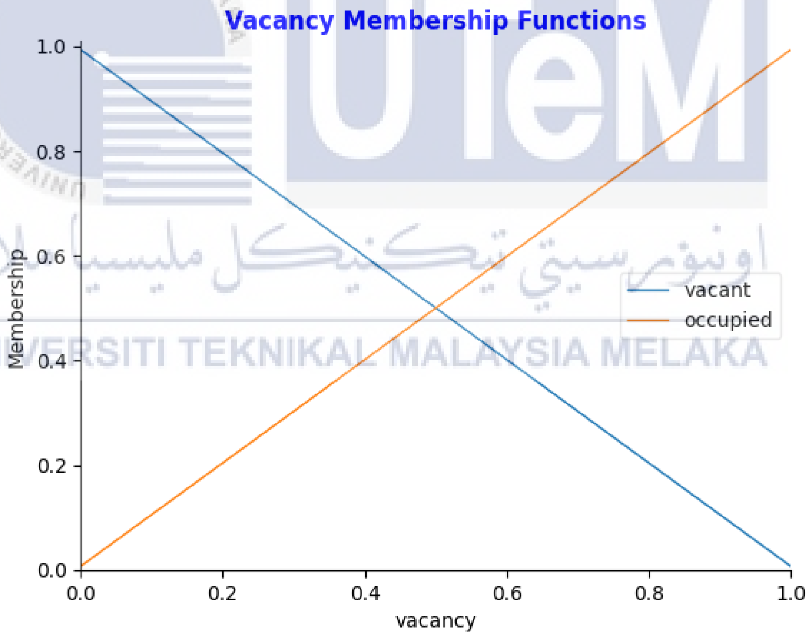


Figure 4.13 Vacancy Membership Functions Fuzzy Sets - Input

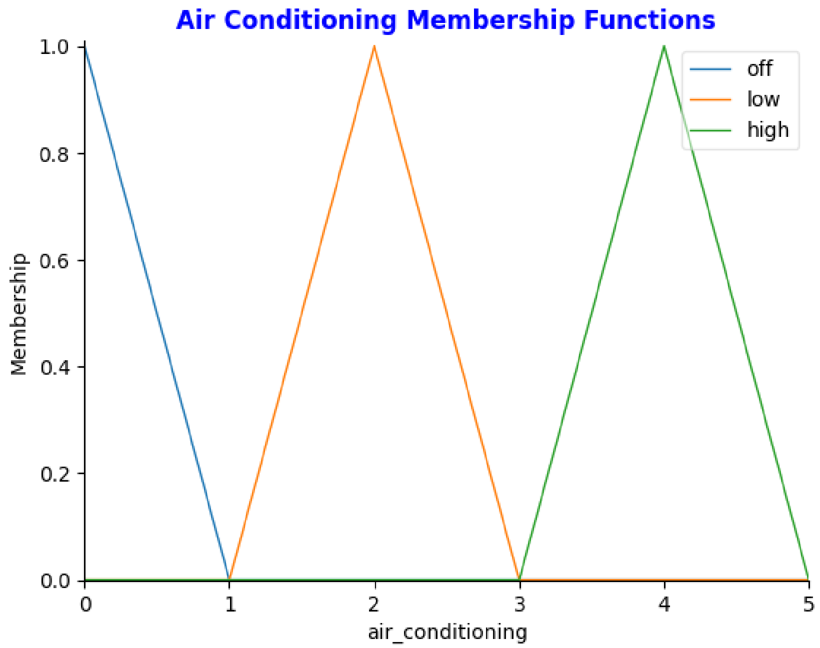


Figure 4.14 Air Conditioning Membership Functions Fuzzy Sets - Output

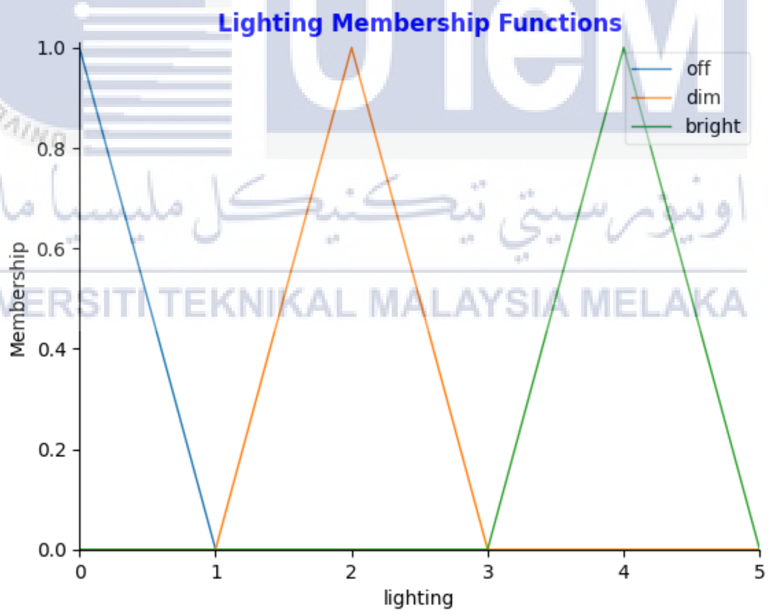


Figure 4.15 Lighting System Membership Functions Fuzzy Sets - Output

To assess the effectiveness and adaptability of the Fuzzy Logic Rule-Based Algorithm, three distinct input conditions were tested. This was done to observe how the algorithm responds and adjusts its output based on varying environmental parameters. The conditions tested are as follows:

- 1st Condition: Temperature at **30.3°C**, Light Intensity at **1608**, and Vacancy at **0 (unoccupied)**.
- 2nd Condition: Temperature at **25°C**, Light Intensity at **1000**, and Vacancy at **1 (occupied)**.
- 3rd Condition: Temperature at 30.3°C, Light Intensity at **3200**, and Vacancy at **1 (occupied)**.

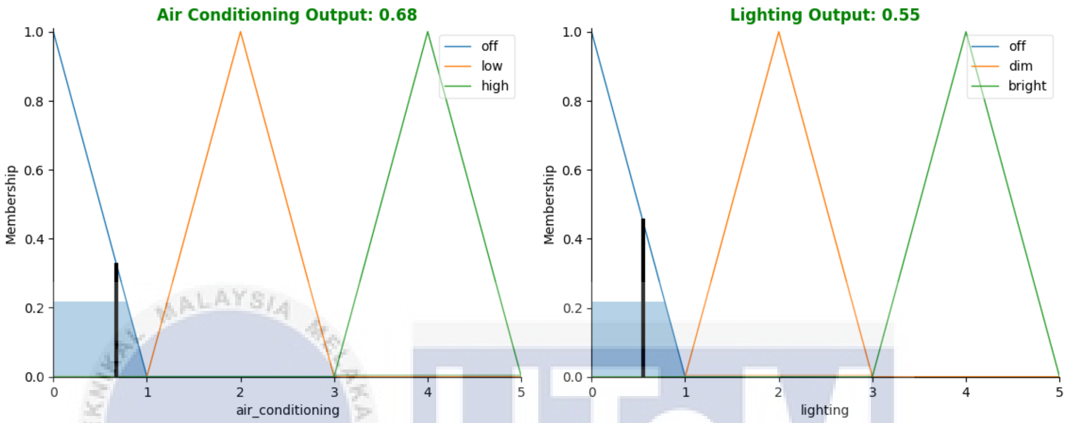


Figure 4.16 1st Condition Test Results

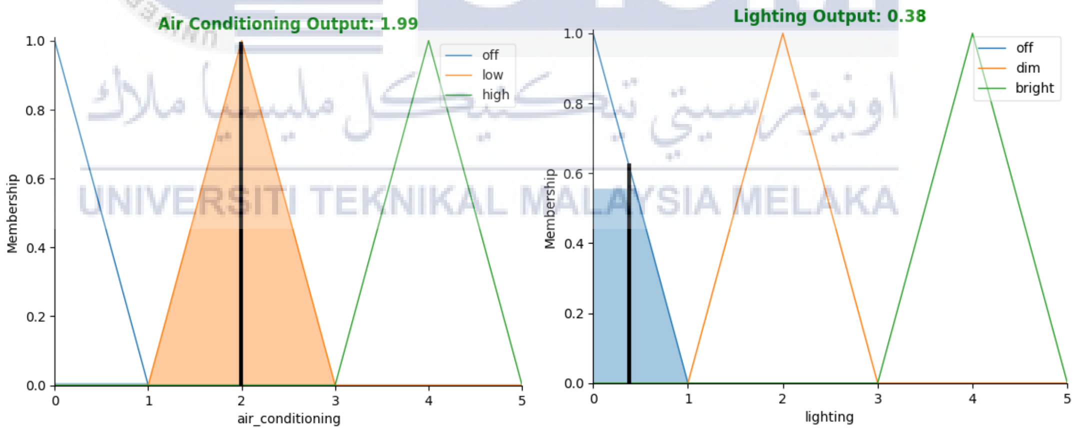


Figure 4.17 2nd Condition Test Results

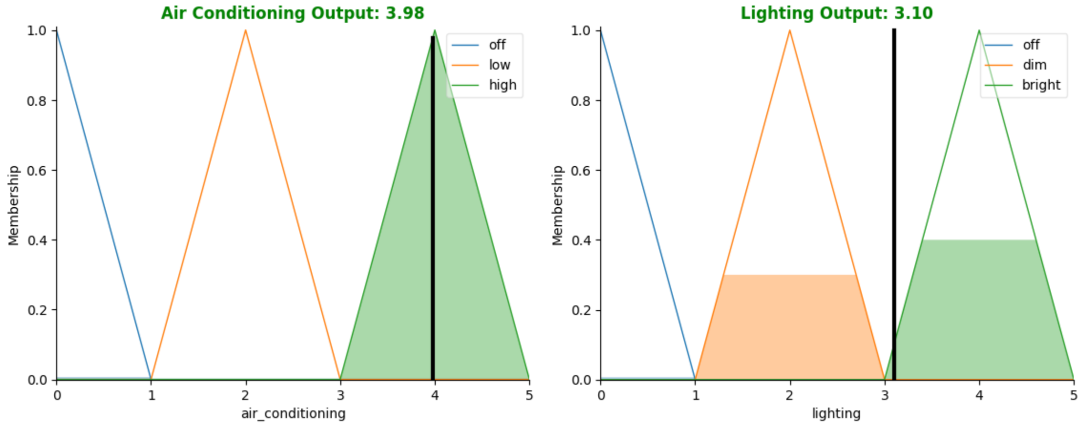


Figure 4.18 3rd Condition Test Results

The corresponding results are illustrated in Figure 4.16, Figure 4.17 and Figure 4.18. These figures demonstrate the algorithm's accurate and context-sensitive responses to the input values. For instance, in 1st Condition, the environment is characterized by a high temperature and bright light, with the room being unoccupied. The algorithm intelligently turns off both the air conditioning and lighting systems, acknowledging their redundancy in the absence of people. This decision not only ensures energy efficiency but also demonstrates the system's ability to recognize and react to the lack of human presence. Next, the 2nd Condition presents a slightly cooler environment at 25°C and bright light conditions, with the room being occupied. Here, the air conditioning is set to a low level, a decision that aligns well with the already comfortable ambient temperature. The lighting system is switched off, recognizing that the natural light suffices for the occupied space. This scenario exemplifies the algorithm's nuanced understanding of occupant comfort and natural lighting conditions. Lastly, in the 3rd Condition, the room's temperature remains high at 30.3°C, but with a significant decrease in light intensity (3200), indicating a darker environment. The algorithm responds by activating both the air conditioning and artificial lighting. This adjustment is crucial for maintaining a comfortable temperature and adequate lighting in the room, particularly important in an occupied setting.

These observations collectively affirm the algorithm's proficiency in intelligently adapting its outputs to diverse environmental states. This adaptability ensures the maintenance of comfort and energy efficiency within a room, highlighting the practical utility of the Fuzzy Logic Rule-Based Algorithm in real-world applications.

4.2.3 Analysis the Efficiency of Algorithm Output for the Developed Energy Management System

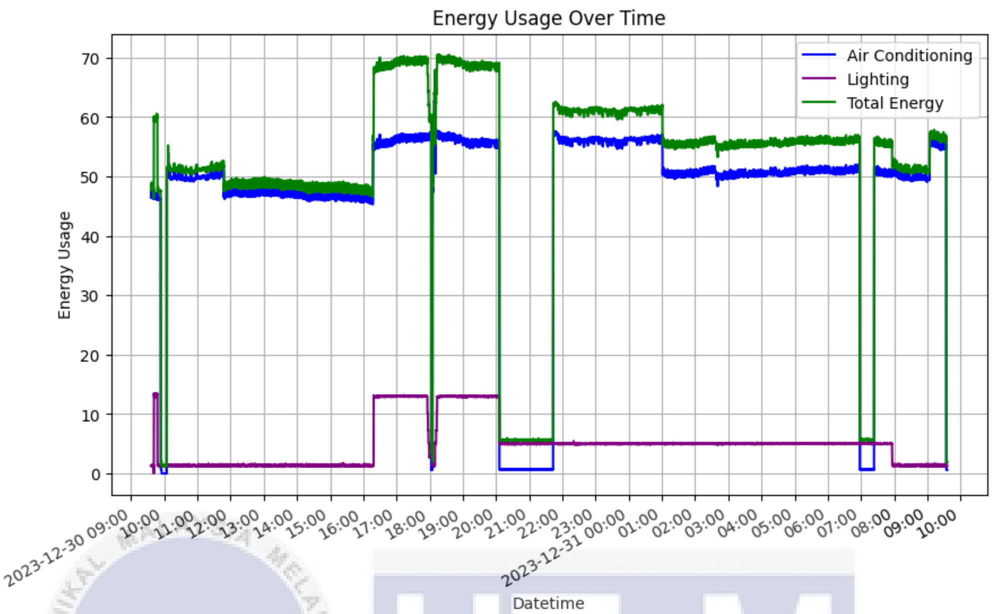


Figure 4.19 Energy Usage Over Time

Figure 4.19 illustrates the energy consumption of both the air conditioning and lighting systems as recorded by the developed hardware of the Energy Management System. This data represents a successful 24-hour period of energy monitoring. The graph reveals that the air conditioning system accounts for the majority of the energy consumption, while the lighting system utilizes approximately 13W. When operating the air conditioning at its highest capacity in conjunction with the lighting system at its brightest setting, the combined energy consumption can reach up to approximately 70.5W. This demonstrates the effectiveness of the system in capturing and quantifying energy usage across different modes and intensities of these utilities.

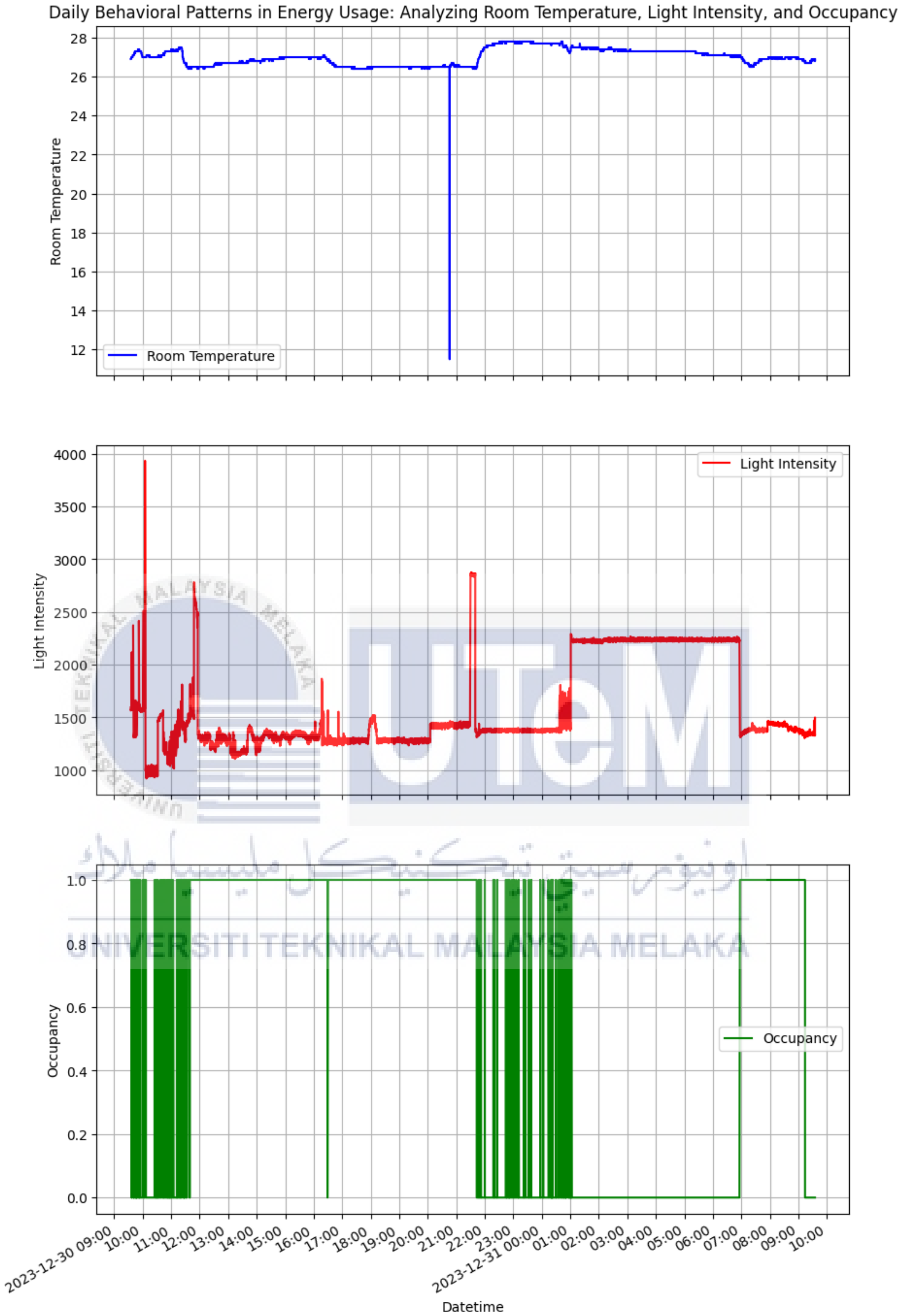


Figure 4.20 Daily Behavioral Pattern in Energy Usage – Room Temperature, Occupancy and Light Intensity

Not only the energy consumption is recorded but also the behavioral pattern in energy usage is also taken into account. The Figure 4.20 provides a comprehensive visualization of the daily patterns in a room's environment, crucial for understanding energy usage. This graph displays data over a 24-hour period, capturing fluctuations in room temperature, occupancy status, and light intensity.

The importance of this data lies in its role as a foundational input for the subsequent fuzzy logic simulation. By analyzing these parameters, we gain insights into how environmental conditions and human presence interact and influence energy consumption within space.

The room temperature graph displayed on Figure 4.20 recorded over a full 24-hour period, illustrating how the temperature is influenced by both the air conditioning system and the day's weather conditions. Notably, there is a minor fluctuation in the data on December 30, 2023, between 20:00. and 21:00., which can be attributed to a temporary disruption in Wi-Fi connectivity. Despite this brief anomaly, the integrity of the temperature readings remains unaffected for the remainder of the period, ensuring a consistent and reliable dataset.

The light intensity graph depicts the daily variation of intensity, measured in digital units rather than Lux or Lumen, as outlined in the methodology section. For the purposes of this project, categorizing light intensity into low, moderate, or high levels suffices. The use of digital units offers a straightforward and efficient method to monitor and adapt to changes in light conditions. This approach bypasses the complexity of calibration required for precise Lux or Lumen measurements, focusing instead on the practicality of understanding relative light intensity. The graph thus reflects the simplicity and effectiveness of using digital units to gauge environmental lighting, aligning with the project's requirements for assessing light intensity without the intricacies of calibrated measurements.

Next, the recorded occupancy data effectively indicates the presence of people within the room. Typically, when individuals are present, they exhibit movement, especially during interactions with one another. In this context, the PIR (Passive Infrared) Sensor proves to be an ideal component for detecting motion within the room. The sensor's output is binary: a value of 1 signifies that the room is occupied, while 0 indicates it is vacant. This parameter plays a crucial role in managing energy consumption. For instance, when the room or building is no longer in use, the information about its vacancy is vital. It can influence the fuzzy logic algorithm to issue commands to both the air conditioner and lighting system to power down, thereby conserving energy. This mechanism ensures that energy usage is optimized based on actual occupancy, leading to more efficient and sustainable operations.

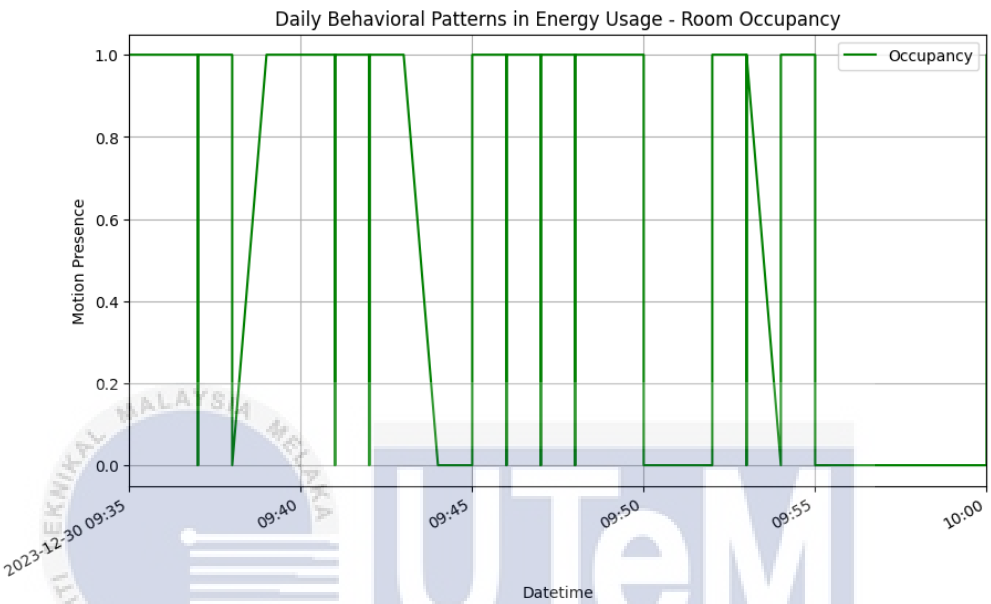


Figure 4.21 Daily Behavioral Pattern in Energy Usage – Room Occupancy from 9:35 to 10:10 on 30 Dec 2023

Given the challenges in discerning the 24-hour Room Occupancy data due to its rapidly fluctuating values between 0 and 1, Figure 4.21 provides a zoomed-in view of the graph covering the period from 9:35 to 10:10 on 30th December 2023. This specific timeframe captures data related to the movement of people entering and exiting the room, without focusing on the duration of their stay. This particular set of data offers valuable insights into the behavior of the Fuzzy Logic Rule-Based Algorithm under such conditions. It allows for a clearer observation of how the algorithm responds to frequent changes in room occupancy, highlighting its adaptability and effectiveness in managing energy consumption in response to real-time human presence.

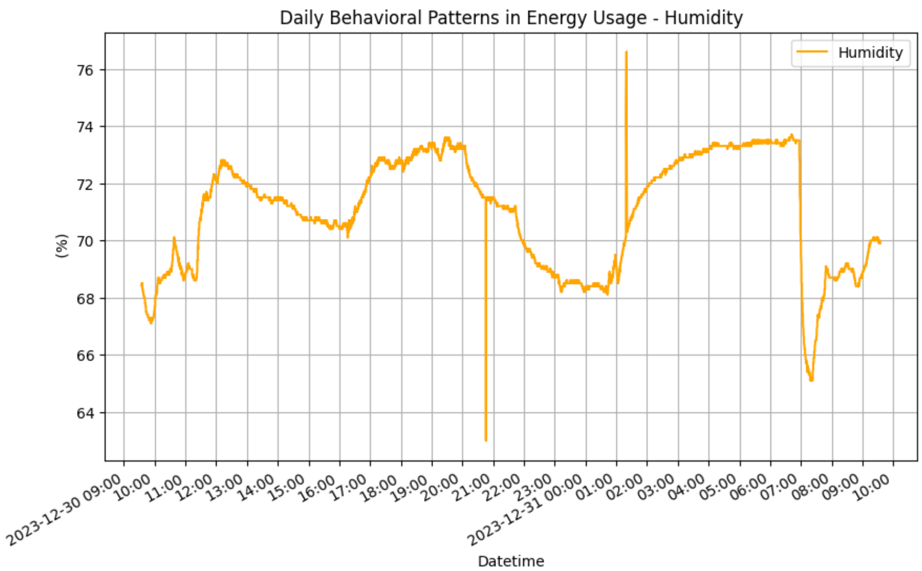


Figure 4.22 Daily Behavioral Pattern in Energy Usage – Humidity

As for the Humidity, the graph presents the humidity data over a 24-hour period. While humidity levels do not directly influence the fuzzy logic output of the energy management system, they provide useful contextual information. Understanding humidity trends is beneficial for a comprehensive analysis of environmental conditions, although this aspect is not a primary factor in the system's operational context. As such, the graph serves an informative purpose and is not integral to the core functionality of the energy management system development.

4.2.3.1 Fuzzy Logic Rule Based Algorithm Output Results

This section delves into the Output Results of the Fuzzy Logic Based Algorithm, focusing on how the algorithm's outputs are simulated within a Python Environment. The simulation is designed to showcase the effectiveness of the algorithm in optimizing energy consumption under various conditions. By analyzing both raw data and the results post-simulation, this section offers a comprehensive view of the algorithm's performance in real-world scenarios.

The graphs presented here provide a visual representation of energy consumption patterns, both before and after the application of the fuzzy logic system. The initial data reflects the daily energy usage under standard operating conditions, serving as a baseline for comparison. Subsequently, the simulated results demonstrate the impact of the fuzzy logic algorithm, highlighting its potential in reducing energy consumption and enhancing efficiency.

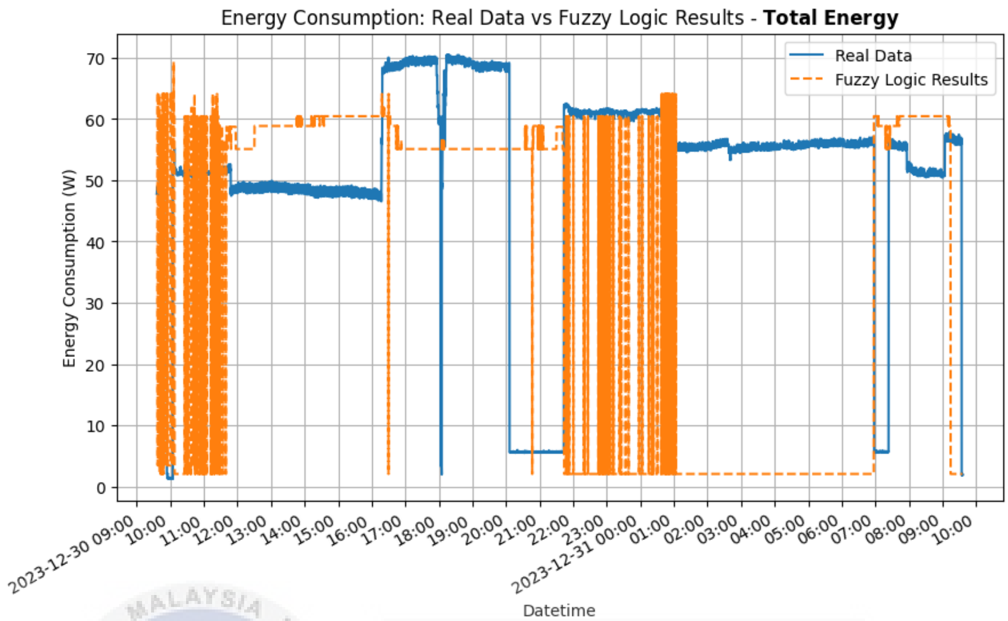


Figure 4.23 Energy Consumption – Real Data Vs Fuzzy Logic Results

The graph compares two distinct data sets, the real data, representing energy usage without any algorithmic intervention, and the output from the fuzzy logic algorithm. The fuzzy logic consists of cumulative energy consumption of both the air conditioning and lighting systems over a 24-hour period. In the initial phase of the simulation, specifically at the 3-hour mark, the fuzzy logic algorithm actively modulates both the air conditioning and lighting. At this point, the energy consumption is observed to be higher than that recorded in the real data. From the 4-hour to the 8-hour mark, although the energy usage remains consistent, it is still slightly above the real data levels. However, a significant change is observed between the 8-hour and 12-hour marks. During this period, the fuzzy logic simulation yields a notably lower energy consumption compared to the real data, indicating effective energy management by the algorithm. This trend of reduced energy usage continues from the 12-hour mark onwards, showcasing notable energy conservation. The simulated energy consumption is substantially lower than the actual energy usage, aligning with the observed behavioral patterns in energy usage. A particularly striking result is seen on 31 December 2023, from 1:00 to 7:00., where the energy consumption in the simulation reaches its lowest point. This reduction is attributed to the lack of motion detected in the room during these hours, implying an absence of occupants. It can be the seen the Figure 4.20 showing no presence of people in the room during these times.

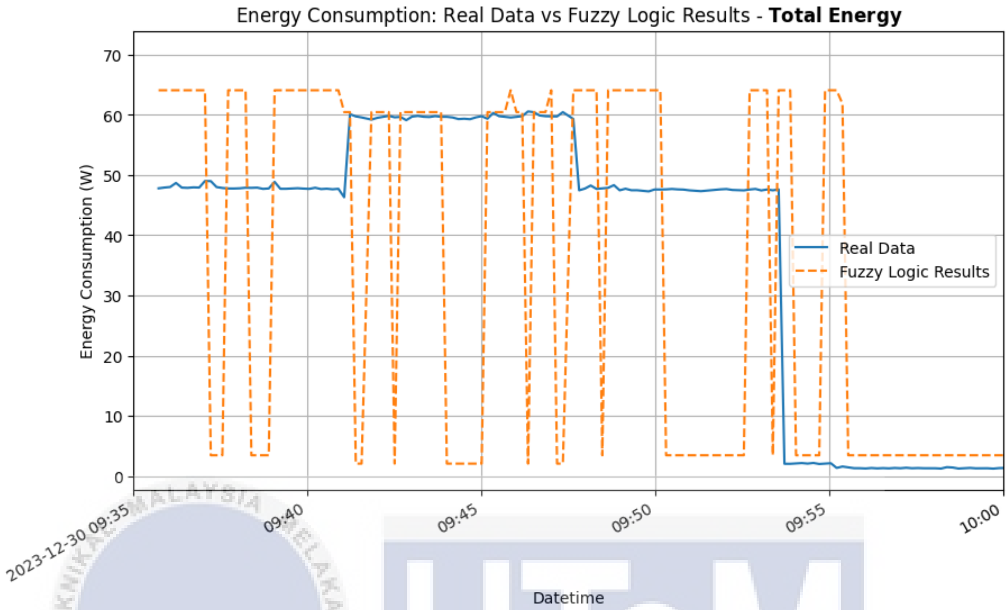


Figure 4.24 Energy Consumption Real Data vs Fuzzy Logic Result 9:35 to 10:00 on 30 Dec 2023

Figure 4.24 focuses on the timeframe from 9:35 to 10:00 on 30 December 2023, it becomes evident that the results from the fuzzy logic exhibit a modulating shape. This pattern is attributed to the Behavioral Pattern indicating that the Temperature, Light Intensity, and Vacancy levels are sufficient to meet user preferences without the need for air conditioning or lighting systems, leading to their deactivation. This is observable through the transition from high power consumption to a lower level, as depicted in the graph. Additionally, the occupancy input significantly influences the results.

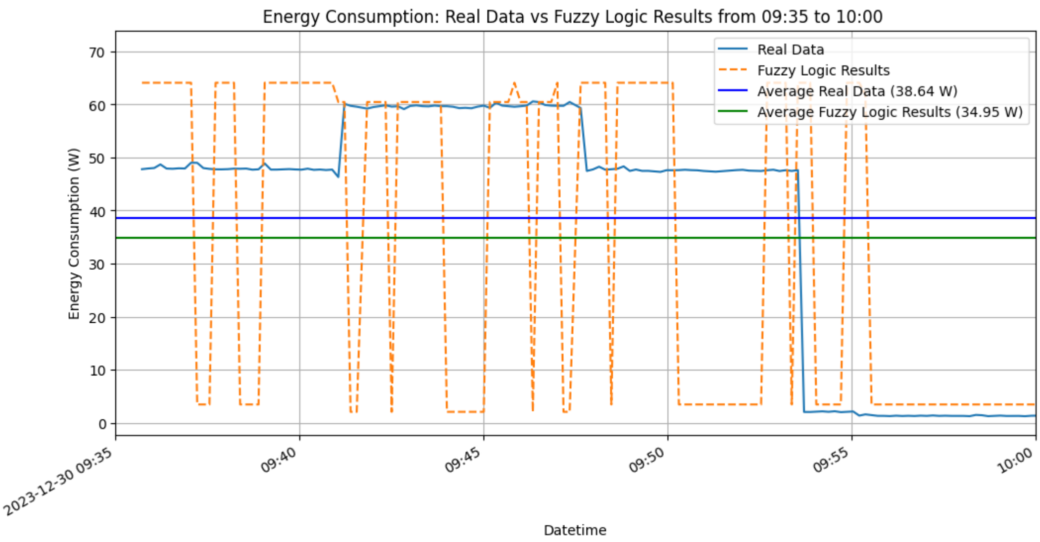


Figure 4.25 Average Energy Consumption of Real Data and Fuzzy Logic Results 9:35 to 10:00 on 30 Dec 2023

The average energy consumption without the fuzzy logic algorithm is 38.64W, in contrast with the implementation of fuzzy logic of 34.94 W, as shown in Figure 4.28. This demonstrates the effectiveness of Fuzzy Logic in reducing energy consumption while still preserving user comfort. The Fuzzy Logic algorithm operates by controlling both the air conditioning and lighting systems, ensuring that these appliances are turned off when the room conditions fall within the 'comfort zone' region.

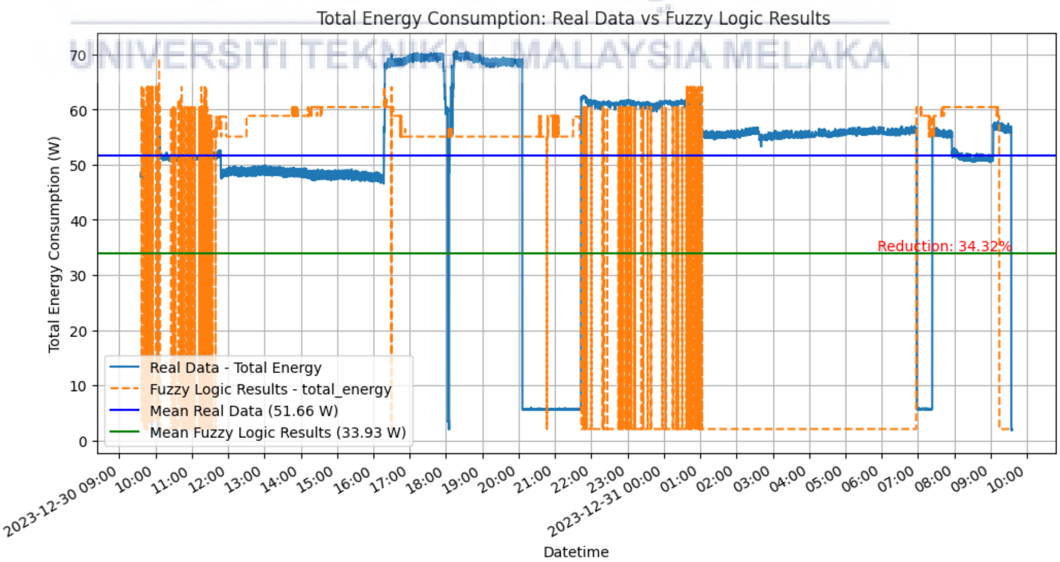


Figure 4.26 Average Energy Consumption of Total Energy: Real Data vs Fuzzy Logic Results

The data presented in Figure 4.26, underscores the significant benefits of employing a fuzzy logic-based rule algorithm in energy management. The graph compares the average total energy consumption captured in the real data with that of the fuzzy logic results. Notably, the total energy consumption without the fuzzy logic algorithm stands at 51.66W, whereas it drops to 33.93W when the fuzzy logic algorithm is implemented. This equates to a substantial 34.32% reduction in energy usage. Such a notable decrease in energy consumption highlights the efficiency of the fuzzy logic algorithm in optimizing energy use. The algorithm's ability to dynamically adjust to varying conditions and requirements allows for more precise control of energy-consuming devices, such as air conditioning and lighting systems. By intelligently modulating these systems based on real-time data, the algorithm ensures that energy is used only, when necessary, thereby preventing wastage. The fuzzy logic approach adds a layer of sophistication to traditional rule-based systems. It allows for more nuanced decisions that take into account the complexity and variability of real-world scenarios. This adaptability is key to achieving significant energy savings.

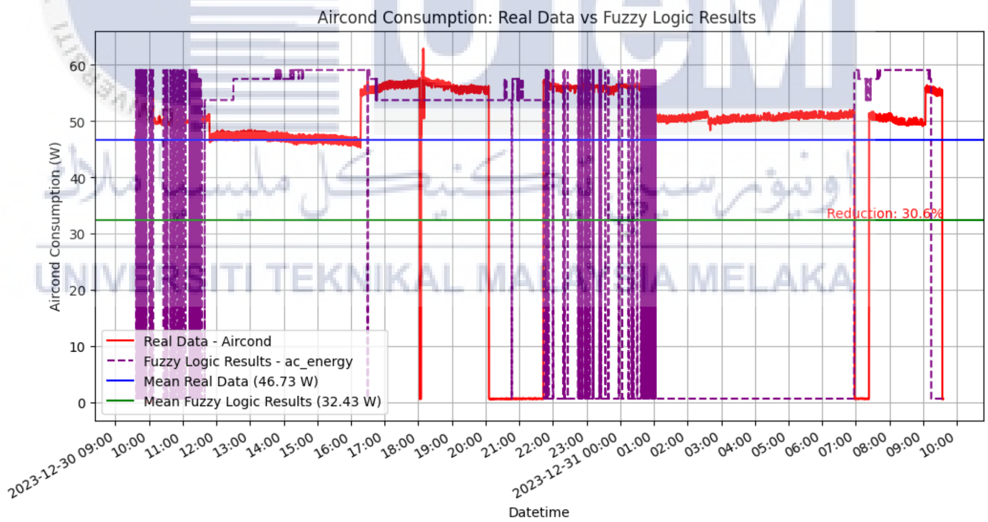


Figure 4.27 Average Energy Consumption of Air Conditioner: Real Data vs Fuzzy Logic Results

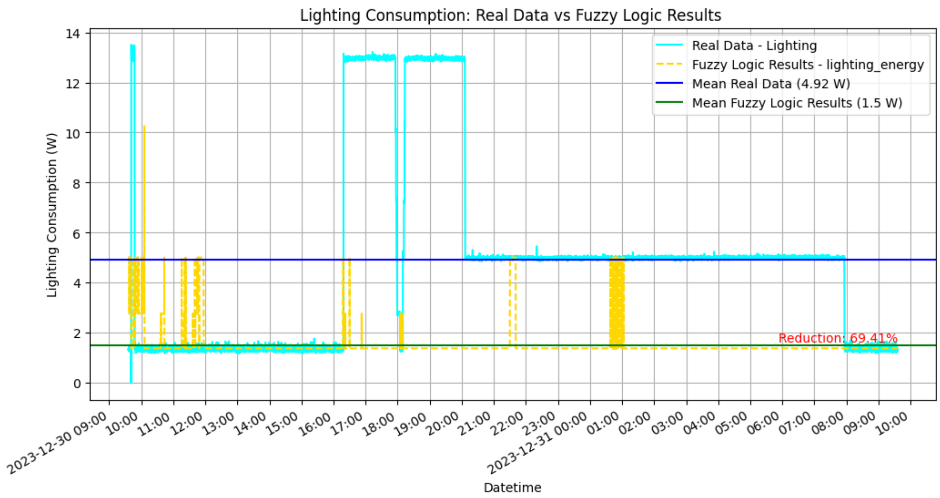


Figure 4.28 Average Energy Consumption of Lighting System: Real Data vs Fuzzy Logic Results

Figure 4.27 and Figure 4.28 provide a detailed analysis of the average energy consumption specifically for the air conditioning and lighting systems, respectively. The results not only delineate the energy usage patterns but also highlight the significant reductions achieved through the implementation of the fuzzy logic algorithm. In Figure 4.27, which focuses on the air conditioning system, the data reveals a 30.6% reduction in energy consumption when the fuzzy logic algorithm is applied. This reduction is indicative of the algorithm's ability to intelligently control the air conditioning system, optimizing its operation based on real-time environmental conditions and occupancy patterns. By adjusting the cooling output in response to actual need, the system avoids unnecessary energy expenditure, leading to this notable decrease in usage. Similarly, Figure 4.28, dedicated to the lighting system, shows an even more striking reduction of 69.41% in energy consumption. This substantial decrease underscores the efficiency of the fuzzy logic algorithm in managing lighting needs. The system dynamically adjusts the lighting intensity and operation, capitalizing on natural light when available and reducing artificial lighting to the minimum necessary level. This approach not only conserves energy but also adapts to the varying light requirements throughout the day.

Together, these figures illustrate the profound impact that intelligent, fuzzy logic-based control systems can have on energy consumption in key areas of a building's operation. The reductions of 30.6% and 69.41% in the air conditioning and lighting systems, respectively, demonstrate the potential of such technology to significantly lower energy usage, thereby contributing to operational costs. These findings reinforce the importance of integrating advanced algorithms in building management systems to achieve greater energy efficiency.

Table 4-1 Tabulated Average Energy Consumption: Real Data vs Fuzzy Logic Results

	Average Power (W)		Percentage Reduction (%)
	Real Data	Fuzzy Logic Rule Based	
Total Energy	55.66	33.93	34.32
Air Conditioning	46.73	32.43	30.6
Lighting System	4.92	1.5	69.41

4.2.3.2 Prediction Model

This section is dedicated to developing a prediction model for energy consumption data, leveraging the principles and techniques outlined in [70]. Utilizing the advanced capabilities of the XGBoost framework, known for its powerful performance in complex time-series analysis, the model aims to predict future trends in energy usage [70]. The development of a prediction model for energy consumption data is crucial for optimizing energy distribution, reducing costs, and promoting sustainable energy utilization. Leveraging the advanced capabilities of the XGBoost framework, known for its powerful performance in complex time-series analysis, the model aims to predict future trends in energy usage. This predictive analysis is key to improving energy management, providing insights for forward-thinking and strategic planning. The ability to accurately project future energy needs is vital for the reliability and effectiveness of any energy management system.

The use of advanced techniques such as XGBoost for time-series analysis is well supported in the literature. For instance, [60] discuss a feature-based prediction method for energy consumption and its application, demonstrating the relevance of advanced prediction methods in energy consumption [60]. Additionally, highlight the accuracy and reliability of artificial intelligence forecasting models, including XGBoost, for time series forecasting of economic and energetic variables, further supporting the use of advanced techniques in energy consumption prediction [59].

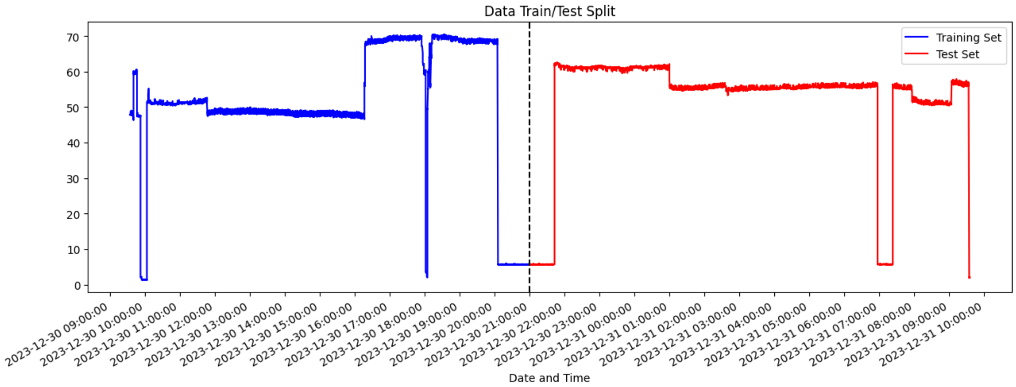


Figure 4.29 Train Test Split Total Energy Consumption Data

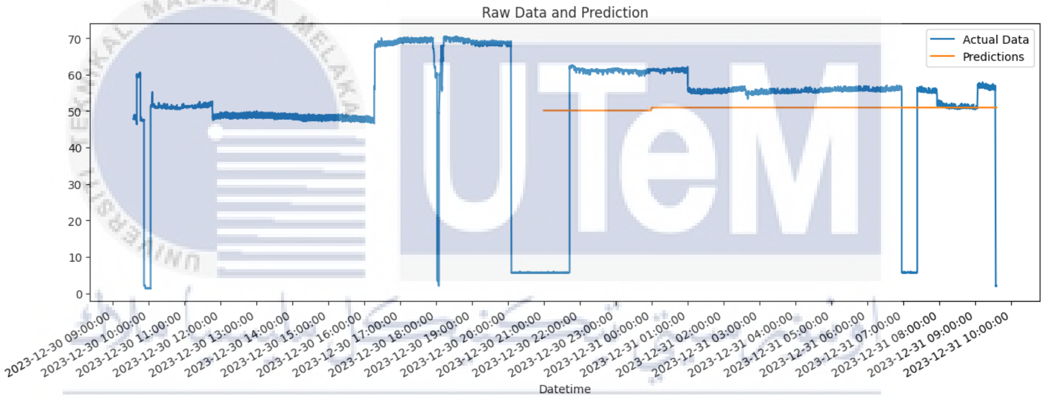


Figure 4.30 Train Test Split Total Energy Consumption Data Results

From the final plot, it can be seen that the model performance is quite good as the differences between the actual data and the predictor are quite accurate. The close correlation between the model's predictions and the actual data, as seen in the final plot, indicates a high degree of accuracy. This precision is crucial for the reliability and effectiveness of any energy management system.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This project focused on the development of an IoT-based Energy Management System (EMS) for air conditioners and lighting systems. It has been successfully demonstrated the potential of integrating IoT with a Fuzzy Logic Rule-Based approach to optimize energy consumption. The EMS, characterized by its advanced real-time monitoring and automation capabilities, has shown a significant advancement in energy management. The implementation of the system encompasses a user-friendly interface developed using Node-Red, enabling effective control over room temperature and lighting settings. The Air Conditioner Automation System, operated through an ESP 32 microcontroller, and the Room Lighting Automation System both exhibited responsive adjustments based on environmental variables and occupancy, highlighting the system's adaptability and user-centric design. Crucially, the application of the Fuzzy Logic Rule-Based Algorithm has been a cornerstone of this study. It has proven to be highly effective in adapting to various environmental conditions, thereby ensuring optimal comfort and energy efficiency. The experimental results underscore the algorithm's precision in managing energy usage, as evidenced by a notable reduction in energy consumption, particularly in the lighting system which observed a 69.41% decrease, and a 30.6% decrease in the air conditioning system. These findings firmly establish the efficacy of the IoT-based EMS in not only enhancing energy efficiency but also in contributing to sustainable energy management practices. The project paves the way for further exploration and implementation of such technologies in the realm of energy conservation, marking a significant step towards achieving sustainable development goals.

5.2 Future Works

In the realm of energy management, the advancement of predictive models represents a crucial frontier. Building on the foundation laid by this project, future work should focus on the development and refinement of a prediction model for energy consumption data. This model, inspired by the principles in the article "Leveraging XGBoost for Time-Series Forecasting" from KDnuggets, should utilize the XGBoost framework. Renowned for its robustness in complex time-series analysis, XGBoost can significantly enhance the predictive capabilities of the energy management system.

The primary objective for future endeavors will be to refine and validate the model, ensuring it can accurately predict future trends in energy usage. This accuracy is not just a technical achievement but a cornerstone for effective energy management. It allows for forward-thinking strategies and informed decision-making, ultimately leading to optimized energy distribution, cost reduction, and efficient energy utilization.

Figures like 4.29 and 4.30, depicting the Train Test Split for Total Energy Consumption Data, demonstrate the model's promising performance, with a high correlation between the model's predictions and actual data. Future work should aim to further improve this accuracy, making the model an invaluable tool for both short-term and long-term energy management planning.

In conclusion, the development of this predictive model is not just an extension of the current project but a significant leap towards a more intelligent, efficient, and proactive approach to energy management. By accurately forecasting energy needs, we can ensure more sustainable energy practices, ultimately leading to a better balance between energy consumption and environmental impact. The future work will be pivotal in elevating the efficacy of IoT-based Energy Management Systems to new heights, ensuring they remain at the forefront of technological innovation in energy conservation and management.

REFERENCES

[1] National Geographic Society, "Nonrenewable Resources," National Geographic. Accessed: May 26, 2023. [Online]. Available: https://education.nationalgeographic.org/resource/nonrenewable-resources/

[2] C. Nunez, "Fossil fuels—facts and information," National Geographic. Accessed: May 26, 2023. [Online]. Available: https://www.nationalgeographic.com/environment/article/fossil-fuels

[3] H. Ritchie, M. Roser, and P. Rasado, "Energy mix - Our World in Data," OurWorldInData.org. Accessed: May 26, 2023. [Online]. Available: https://ourworldindata.org/energy-mix

[4] J. McCurry, "South Korea to shut a quarter of its coal-fired plants over winter to cut pollution | South Korea | The Guardian," The Guardian. Accessed: May 11, 2023. [Online]. Available: https://www.theguardian.com/world/2019/nov/28/south-korea-to-shut-a-quarter-of-its-coal-fired-plants-over-winter-to-cut-pollution

[5] N. H. Mohd Salleh, "Is Malaysia ready to leave coal behind in renewable energy push? | MalaysiaNow." Accessed: May 11, 2023. [Online]. Available: https://www.malaysianow.com/news/2022/04/13/is-malaysia-ready-to-leave-coal-behind-in-renewable-energy-push

[6] Engie, "Thermal power station | Thermal energy | ENGIE." Accessed: May 11, 2023. [Online]. Available: https://www.engie.com/en/activities/thermal-energy/thermal-power-stations

[7] EPA, "Sources of Greenhouse Gas Emissions | US EPA," United States Environmental Protection Agency. Accessed: May 11, 2023. [Online]. Available: https://www.epa.gov/ghgemissions/sources-greenhouse-gas-emissions

[8] NST Business, "No escape from soaring energy prices." Accessed: May 11, 2023. [Online]. Available: https://www.nst.com.my/business/2022/03/781312/no-escape-soaring-energy-prices

[9] O. V. G. Swathika and K. T. M. U. Hemapala, "IOT Based Energy Management System for Standalone PV Systems," *Journal of Electrical Engineering & Technology*, vol. 14, no. 5, pp. 1811–1821, Sep. 2019, doi: 10.1007/s42835-019-00193-y.

[10] Codibly, "What is an Energy Management System? How an EMS Works?," Codibly. Accessed: May 12, 2023. [Online]. Available: https://codibly.com/news-insights/what-is-an-energy-management-system/

[11] N. Hossein Motlagh, M. Mohammadrezaei, J. Hunt, and B. Zakeri, "Internet of Things (IoT) and the Energy Sector," *Energies (Basel)*, vol. 13, no. 2, p. 494, Jan. 2020, doi: 10.3390/en13020494.

[12] S. R. Paveethra, B. Barathi, M. Geethapriya, M. Arthi, and V. Ahasthiya, "Theoretical modelling and implementation of home energy management system using IoT based automation system," in *Materials Today: Proceedings*, Elsevier Ltd, 2021, pp. 1790–1793. doi: 10.1016/j.matpr.2020.08.644.

[13] S. K. Rathor and D. Saxena, “Energy management system for smart grid: An overview and key issues,” *Int J Energy Res*, vol. 44, no. 6, pp. 4067–4109, May 2020, doi: 10.1002/er.4883.
[14] T. De Vass, H. Shee, and S. J. Miah, “The effect of ‘Internet of Things’ on supply chain integration and performance: An organisational capability perspective,” *Australasian Journal of Information Systems*, vol. 22, Jun. 2018, doi: 10.3127/ajis.v22i0.1734.
[15] Y. Banothe, Thakur.Roshni, and A. Banothe, “VIRTUALLY DESIGN COLLEGE CAMPUS NETWORK USING CISCO PACKET TRACER,” *International Research Journal of Modernization in Engineering Technology and Science*, Mar. 2023, doi: 10.56726/IRJMETS34934.
[16] A. Dawod, D. Georgakopoulos, P. P. Jayaraman, A. Nirmalathas, and U. Parampalli, “IoT Device Integration and Payment via an Autonomic Blockchain-Based Service for IoT Device Sharing,” *Sensors*, vol. 22, no. 4, p. 1344, Feb. 2022, doi: 10.3390/s22041344.
[17] Helmy, Fenny Rahmasari, Arif Nursyahid, Thomas Agung Setyawan, and Ari Sriyanto Nugroho, “Performance Analysis of Android-Based Smart Agriculture Monitor and Control Applications,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 11, no. 1, pp. 23–30, Feb. 2022, doi: 10.22146/jnteti.v11i1.3379.
[18] E. Patti and A. Acquaviva, “IoT platform for Smart Cities: Requirements and implementation case studies,” in *2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI)*, IEEE, Sep. 2016, pp. 1–6. doi: 10.1109/RTSI.2016.7740618.
[19] A. Salam, “Internet of Things for Sustainability: Perspectives in Privacy, Cybersecurity, and Future Trends,” 2020, pp. 299–327. doi: 10.1007/978-3-030-35291-2_10.
[20] H. Elahi, K. Munir, M. Eugeni, S. Atek, and P. Gaudenzi, “Energy Harvesting towards Self-Powered IoT Devices,” *Energies (Basel)*, vol. 13, no. 21, p. 5528, Oct. 2020, doi: 10.3390/en13215528.
[21] H. Shareef, E. Al-Hassan, and R. Sirjani, “Wireless home energy management system with smart rule-based controller,” *Applied Sciences (Switzerland)*, vol. 10, no. 13, Jul. 2020, doi: 10.3390/app10134533.
[22] H. Hosseinian and H. Damghani, “Smart home energy management, using IoT system,” in *2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI)*, IEEE, Feb. 2019, pp. 905–910. doi: 10.1109/KBEI.2019.8734979.
[23] A. Srinivasan, K. Baskaran, and G. Yann, “IoT Based Smart Plug-Load Energy Conservation and Management System,” in *2019 IEEE 2nd International Conference on Power and Energy Applications (ICPEA)*, IEEE, Apr. 2019, pp. 155–158. doi: 10.1109/ICPEA.2019.8818534.
[24] P. Pawar, M. TarunKumar, and P. Vittal K., “An IoT based Intelligent Smart Energy Management System with accurate forecasting and load strategy for renewable generation,” *Measurement (Lond)*, vol. 152, Feb. 2020, doi: 10.1016/j.measurement.2019.107187.
[25] J. A. A. Silva, J. C. López, C. P. Guzman, N. B. Arias, M. J. Rider, and L. C. P. da Silva, “An IoT-based energy management system for AC microgrids with grid and security constraints,” *Appl Energy*, vol. 337, May 2023, doi: 10.1016/j.apenergy.2023.120904.

[26] A. Rochd *et al.*, "Design and implementation of an AI-based & IoT-enabled Home Energy Management System: A case study in Benguerir — Morocco," *Energy Reports*, vol. 7, pp. 699–719, Nov. 2021, doi: 10.1016/j.egyr.2021.07.084.

[27] L. Raju, V. Gurunath, R. Darran, and S. V, "IOT based Energy Management System using STM32 and AWS IOT," in *2022 IEEE 2nd International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC)*, IEEE, Dec. 2022, pp. 1–5. doi: 10.1109/iSSSC56467.2022.10051258.

[28] E. J. Palacios-Garcia, B. Arbab-Zavar, J. C. Vasquez, and J. M. Guerrero, "Open IoT infrastructures for in-home energy management and control," in *IEEE International Conference on Consumer Electronics - Berlin, ICCE-Berlin*, IEEE Computer Society, Sep. 2019, pp. 376–379. doi: 10.1109/ICCE-Berlin47944.2019.8966225.

[29] P. N. Krishna, S. R. Gupta, P. V. Shankaranarayanan, S. Sidharth, and M. Sirphi, "Fuzzy Logic Based Smart Home Energy Management System," in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, Jul. 2018, pp. 1–5. doi: 10.1109/ICCCNT.2018.8493744.

[30] D. Bahmanyar, N. Razmjooy, and S. Mirjalili, "Multi-objective scheduling of IoT-enabled smart homes for energy management based on Arithmetic Optimization Algorithm: A Node-RED and NodeMCU module-based technique," *Knowl Based Syst*, vol. 247, p. 108762, Jul. 2022, doi: 10.1016/j.knosys.2022.108762.

[31] H. Shaikh, A. M. Khan, M. Rauf, A. Nadeem, M. T. Jilani, and M. T. Khan, "IoT based Linear Models Analysis for Demand-Side Management of Energy in Residential Buildings," in *2020 Global Conference on Wireless and Optical Technologies, GCWOT 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020. doi: 10.1109/GCWOT49901.2020.9391627.

[32] W. Li, T. Logenthiran, V. T. Phan, and W. L. Woo, "Implemented IoT-based self-learning home management system (SHMS) for Singapore," *IEEE Internet Things J*, vol. 5, no. 3, pp. 2212–2219, Jun. 2018, doi: 10.1109/JIOT.2018.2828144.

[33] T. Anand, S. Upare, S. Jain, M. Andhare, and K. Bhangale, "Deployment of Real-Time Energy Monitoring System Using IoT," in *2022 3rd International Conference for Emerging Technology, INCET 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/INCET54531.2022.9824394.

[34] N. S. Madhuri, K. Shailaja, D. Saha, R. P, K. B. Glory, and M. Sumithra, "IOT integrated smart grid management system for effective energy management," *Measurement: Sensors*, vol. 24, Dec. 2022, doi: 10.1016/j.measen.2022.100488.

[35] T.-Y. Ku, W.-K. Park, and H. Choi, "IoT energy management platform for microgrid," in *2017 IEEE 7th International Conference on Power and Energy Systems (ICPES)*, IEEE, Nov. 2017, pp. 106–110. doi: 10.1109/ICPESYS.2017.8215930.

[36] A. Kelati and H. Gaber, "IoT for Home Energy Management (HEM) Using FPGA," in *2021 9th IEEE International Conference on Smart Energy Grid Engineering, SEGE 2021*, Institute of Electrical and Electronics Engineers Inc., Aug. 2021, pp. 54–57. doi: 10.1109/SEGE52446.2021.9534986.

[37] M. U. Saleem, M. R. Usman, M. A. Usman, and C. Politis, "Design, Deployment and Performance Evaluation of an IoT Based Smart Energy

Management System for Demand Side Management in Smart Grid," *IEEE Access*, vol. 10, pp. 15261–15278, 2022, doi: 10.1109/ACCESS.2022.3147484.

[38] M. U. Saleem, M. R. Usman, and M. Shakir, "Design, Implementation, and Deployment of an IoT Based Smart Energy Management System," *IEEE Access*, vol. 9, pp. 59649–59664, 2021, doi: 10.1109/ACCESS.2021.3070960.

[39] A. E. Nezhad, P. H. J. Nardelli, S. Sahoo, F. Ghanavati, and G. J. Osorio, "A Centralized Home Energy Management System to Minimize Consumer's Electricity Bill," in *2022 IEEE International Conference on Environment and Electrical Engineering and 2022 IEEE Industrial and Commercial Power Systems Europe, EEEIC / I and CPS Europe 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/EEEIC/ICPSEurope54979.2022.9854599.

[40] J. Teicher, "The little-known story of the first IoT device - IBM Blog," IBM. Accessed: May 28, 2023. [Online]. Available: https://www.ibm.com/blog/little-known-story-first-iot-device/

[41] A. Silica, "Interview with Kevin Ashton – inventor of IoT: Is driven by the users | Avnet Silica." Accessed: May 28, 2023. [Online]. Available: https://www.avnet.com/wps/portal/silica/resources/article/interview-with-iot-inventor-kevin-ashton-iot-is-driven-by-the-users/

[42] K. D. Foote, "A Brief History of the Internet of Things - DATAVERSITY," DATAVERSITY. Accessed: May 28, 2023. [Online]. Available: https://www.dataversity.net/brief-history-internet-things/

[43] techahead, "Evolution of Internet of Things (IoT): Past, present and future," techahead. Accessed: May 28, 2023. [Online]. Available: https://www.techaheadcorp.com/knowledge-center/evolution-of-iot/

[44] S. N. Swamy and S. R. Kota, "An Empirical Study on System Level Aspects of Internet of Things (IoT)," *IEEE Access*, vol. 8, pp. 188082–188134, 2020, doi: 10.1109/ACCESS.2020.3029847.

[45] S. MODI, R. NAIYA, and S. SHABANA, "INTELLIGENT ENERGY MANAGEMENT SYSTEM FOR RESIDENTIAL BUILDINGS BASED ON ZIGBEE TECHNOLOGY," *International Journal of Power System Operation and Energy Management*, pp. 181–186, Jul. 2013, doi: 10.47893/IJPSOEM.2013.1085.

[46] J.-S. Lee, Y.-W. Su, and C.-C. Shen, "A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi," in *IECON 2007 - 33rd Annual Conference of the IEEE Industrial Electronics Society*, IEEE, 2007, pp. 46–51. doi: 10.1109/IECON.2007.4460126.

[47] CSA, "Looking Ahead - Industry Trends and New Zigbee Features - CSA-IOT." Accessed: May 29, 2023. [Online]. Available: https://csa-iot.org/newsroom/looking-ahead-industry-trends-and-new-zigbee-features/

[48] Austin, "Wi-Fi Alliance® 2022 Wi-Fi® trends | Wi-Fi Alliance." Accessed: May 29, 2023. [Online]. Available: https://www.wi-fi.org/news-events/newsroom/wi-fi-alliance-2022-wi-fi-trends

[49] M. Nikodem, M. Slabicki, and M. Bawiec, "Efficient Communication Scheme for Bluetooth Low Energy in Large Scale Applications," *Sensors*, vol. 20, no. 21, p. 6371, Nov. 2020, doi: 10.3390/s20216371.

[50] Arrow, "Sensor technology: What is a sensor & uses | Arrow.com." Accessed: May 29, 2023. [Online]. Available: https://www.arrow.com/en/research-and-events/articles/sensor-technologies

[51] Tech Briefs, "Smart Sensor Technology for the IoT - Tech Briefs." Accessed: May 29, 2023. [Online]. Available: https://www.techbriefs.com/component/content/article/tb/pub/features/articles/33212

[52] P. Williams, C. Soares, and J. E. Gilbert, "A clustering rule-based approach to predictive modeling," in *Proceedings of the 48th Annual Southeast Regional Conference*, New York, NY, USA: ACM, Apr. 2010, pp. 1–5. doi: 10.1145/1900008.1900071.

[53] M.-L. Antonie, D. Chodos, and O. Zaïane, "Variations on Associative Classifiers and Classification Results Analyses," in *Post-Mining of Association Rules*, IGI Global, 2009, pp. 150–172. doi: 10.4018/978-1-60566-404-0.ch009.

[54] L. H. Khôi, "Algorithms for refining the rules set and building the regular rule-based system of the expert system.," *Journal of Computer Science and Cybernetics*, vol. 17, no. 2, Aug. 2012, doi: 10.15625/1813-9663/17/2/2439.

[55] S. Ramakrishnan, Ed., *Modern Fuzzy Control Systems and Its Applications*. InTech, 2017. doi: 10.5772/65834.

[56] W. Pedrycz, "Fuzzy control and fuzzy systems," p. 350, 1993.

[57] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965, doi: 10.1016/S0019-9958(65)90241-X.

[58] Y. Huang, Y. Mao, L. Xu, J. Wen, and G. Chen, "Exploring risk factors for cervical lymph node metastasis in papillary thyroid microcarcinoma: construction of a novel population-based predictive model," *BMC Endocr Disord*, vol. 22, no. 1, p. 269, Nov. 2022, doi: 10.1186/s12902-022-01186-1.

[59] M. A. Jaramillo-Morán, D. Fernández-Martínez, A. García-García, and D. Carmona-Fernández, "Improving Artificial Intelligence Forecasting Models Performance with Data Preprocessing: European Union Allowance Prices Case Study," *Energies (Basel)*, vol. 14, no. 23, p. 7845, Nov. 2021, doi: 10.3390/en14237845.

[60] W. Yan, H. Zhang, and F. Ma, "A feature-based prediction method for energy consumption of machining process and its application," *IOP Conf Ser Mater Sci Eng*, vol. 382, p. 032048, Jul. 2018, doi: 10.1088/1757-899X/382/3/032048.

[61] Y. Wang, B. Wang, Tianyang Zhang, H. Nazaripouya, C.-C. Chu, and R. Gadh, "Optimal energy management for Microgrid with stationary and mobile storages," in *2016 IEEE/PES Transmission and Distribution Conference and Exposition (T&D)*, IEEE, May 2016, pp. 1–5. doi: 10.1109/TDC.2016.7520004.

[62] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The Arithmetic Optimization Algorithm," *Comput Methods Appl Mech Eng*, vol. 376, p. 113609, Apr. 2021, doi: 10.1016/j.cma.2020.113609.

[63] C. Blum and D. Merkle, Eds., *Swarm Intelligence*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. doi: 10.1007/978-3-540-74089-6.

[64] Z. Li, Z. Zhu, Y. Song, and Z. Wei, "A multi-objective particle swarm optimizer with distance ranking and its applications to air compressor design optimization," *Transactions of the Institute of Measurement and Control*, vol. 34, no. 5, pp. 546–556, Jul. 2012, doi: 10.1177/0142331211406603.

[65] J. R. Birge and F. Louveaux, *Introduction to Stochastic Programming*. New York, NY: Springer New York, 2011. doi: 10.1007/978-1-4614-0237-4.

[66] tp-link, “Tapo P110 | Mini Smart Wi-Fi Socket, Energy Monitoring | TP-Link Malaysia.” Accessed: Jun. 04, 2023. [Online]. Available: https://www.tp-link.com/my/home-networking/smart-plug/tapo-p110/

[67] F. Sehar, M. Pipattanasomporn, and S. Rahman, “Integrated automation for optimal demand management in commercial buildings considering occupant comfort,” *Sustain Cities Soc*, vol. 28, pp. 16–29, Jan. 2017, doi: 10.1016/j.scs.2016.08.016.

[68] OpenJS Foundation, “About : Node-RED.” Accessed: Jun. 14, 2023. [Online]. Available: https://nodered.org/about/

[69] Y. S. Al-Nahhas, L. A. Hadidi, M. S. Islam, M. Skitmore, and Z. Abunada, “Modified Mamdani-fuzzy inference system for predicting the cost overrun of construction projects,” *Appl Soft Comput*, vol. 151, p. 111152, Jan. 2024, doi: 10.1016/j.asoc.2023.111152.

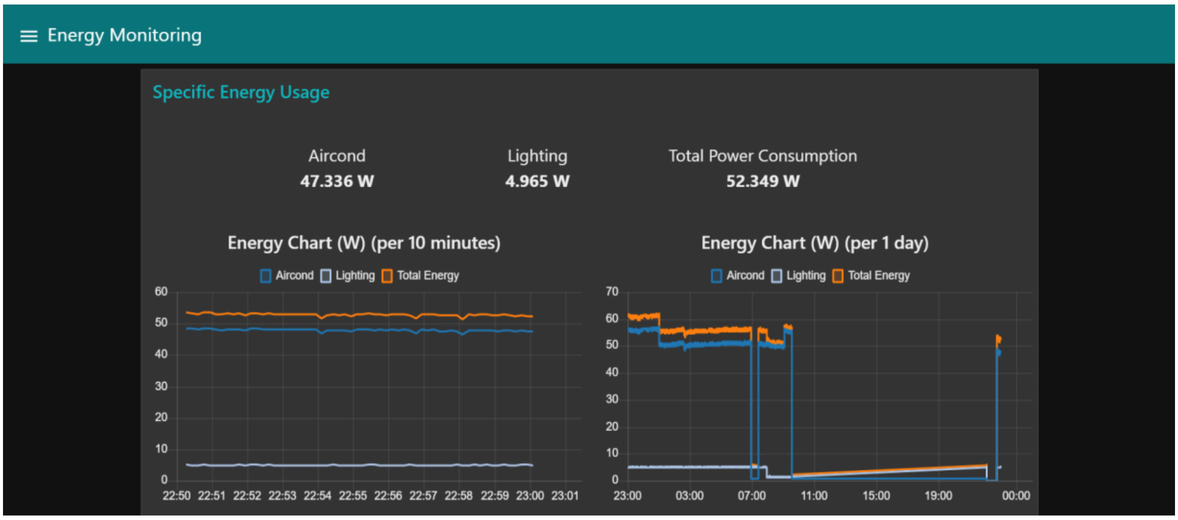
[70] “Leveraging XGBoost for Time-Series Forecasting - KDnuggets.” Accessed: Jan. 08, 2024. [Online]. Available: https://www.kdnuggets.com/2023/08/leveraging-xgboost-timeseries-forecasting.html

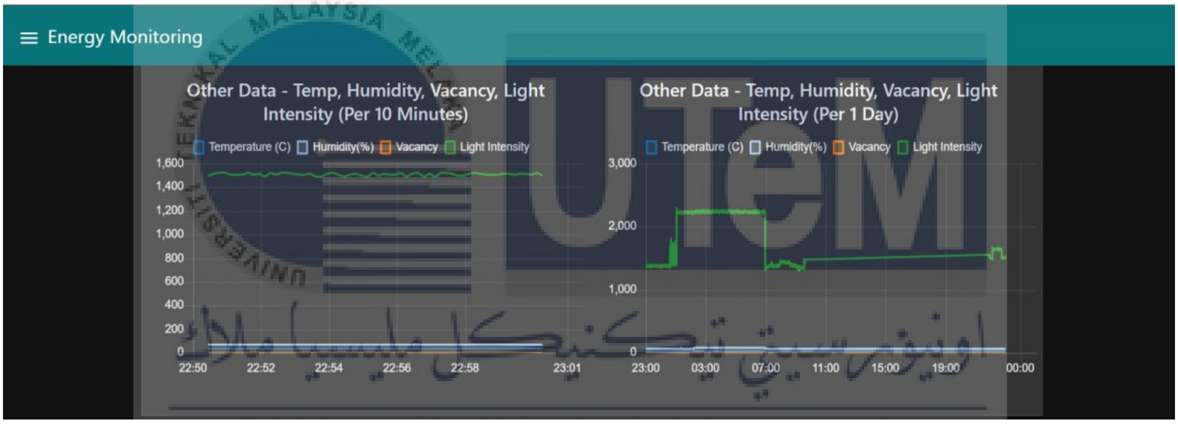
APPENDICES

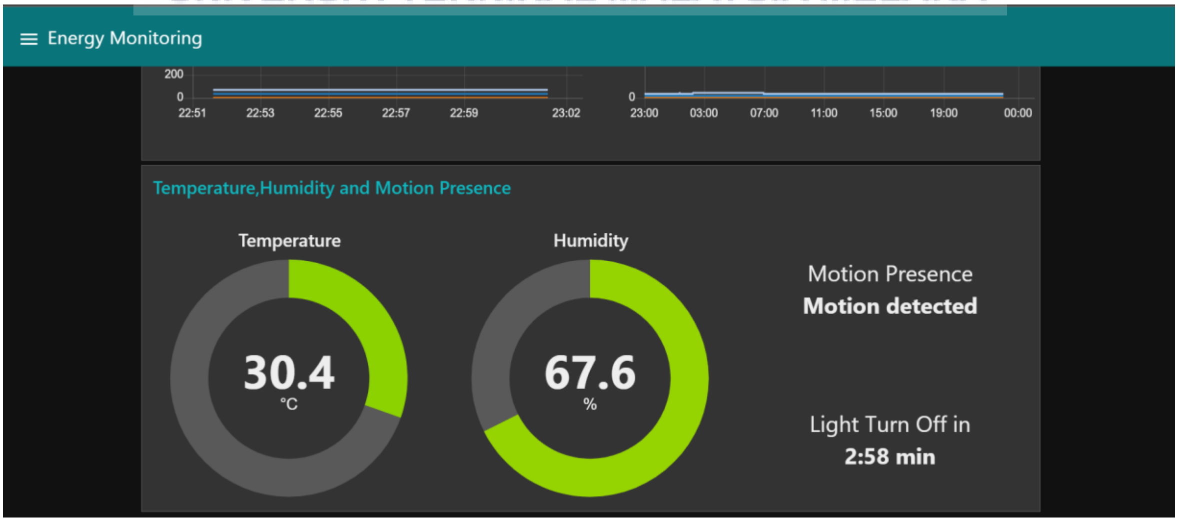
APPENDIX A: Close Up Detailed Design of the Energy Management System Hardware

Figure 5.1 Close Up Detailed View on Lighting System and the IOT Based System Component

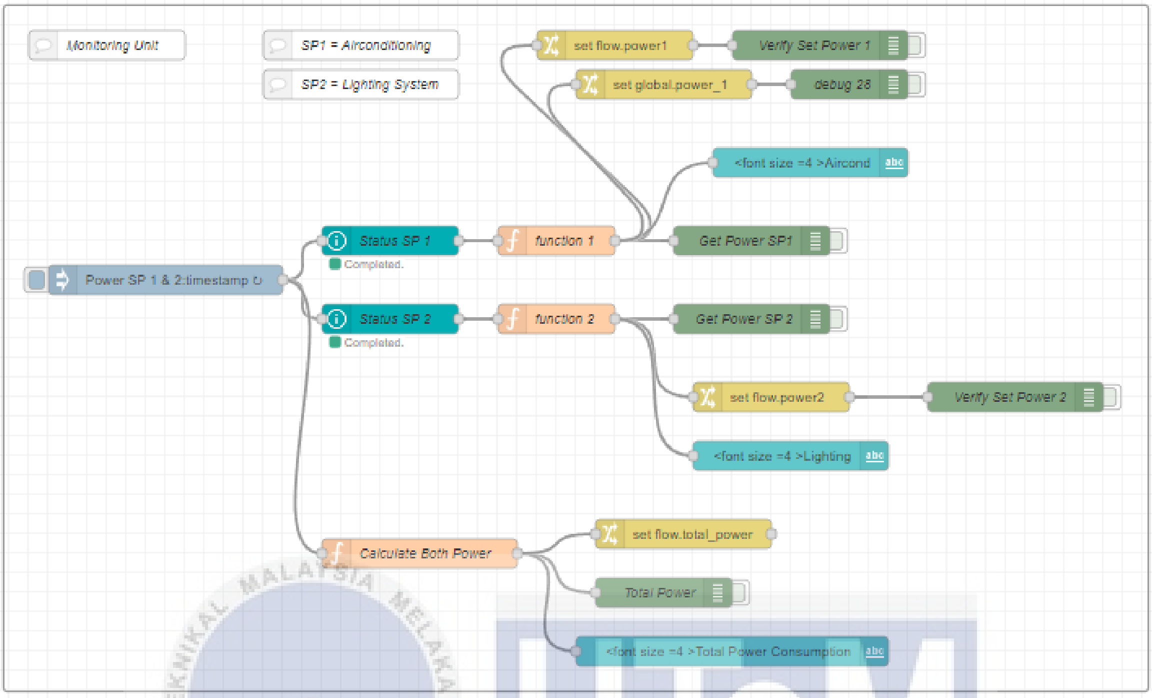
APPENDIX B: Close Up Detailed Design of the Energy Management System User Interface

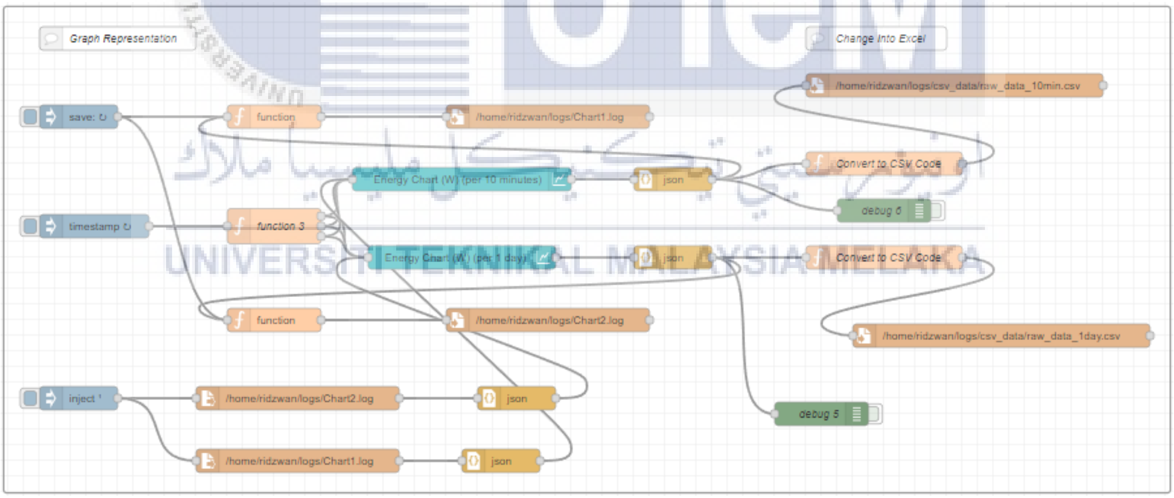


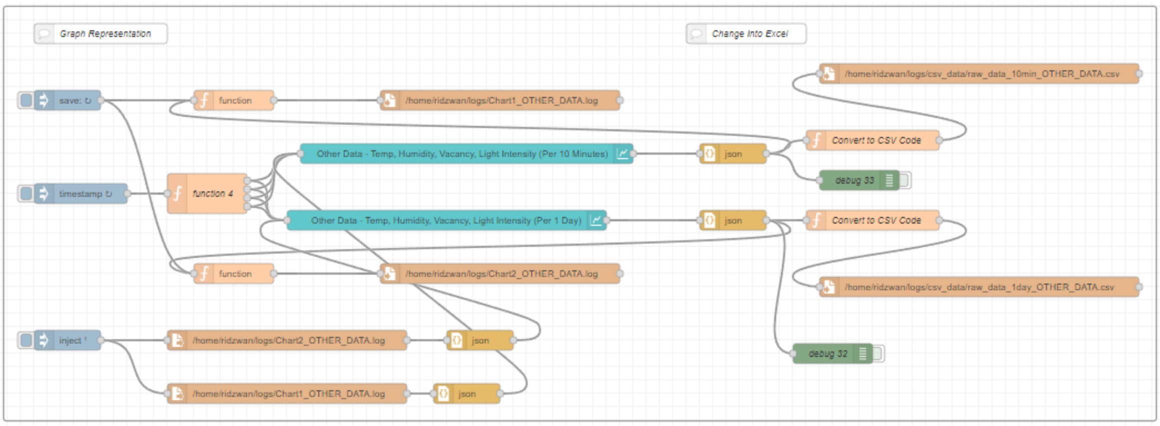


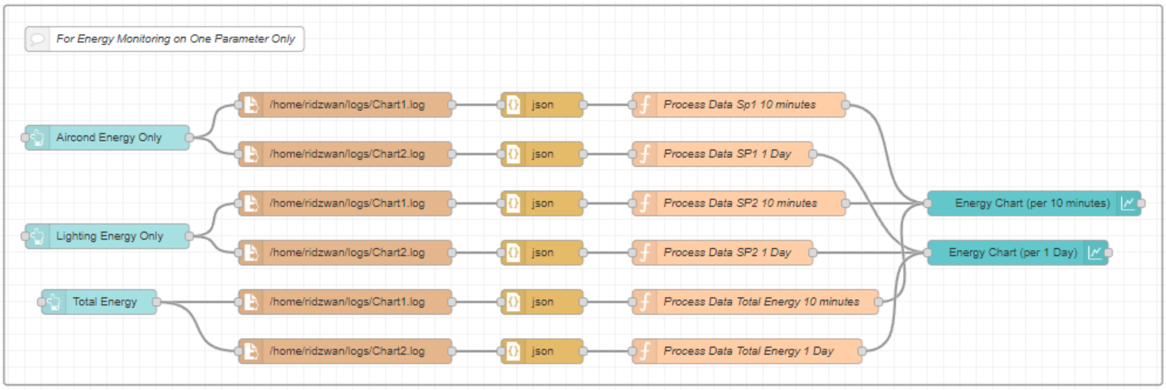


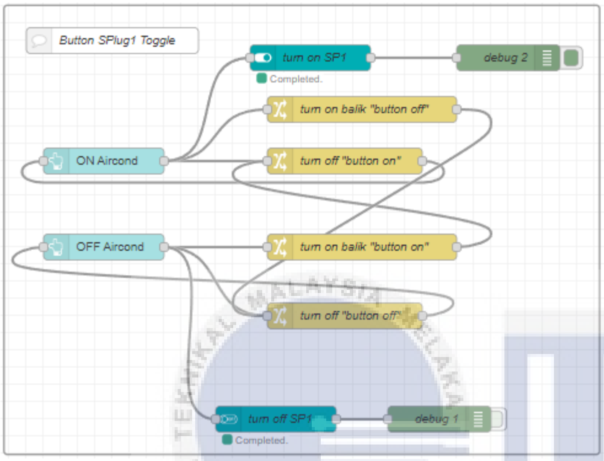
APPENDIX C: Full Node Red Flow

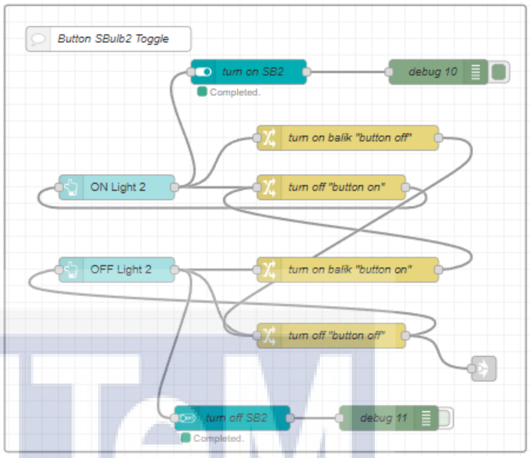


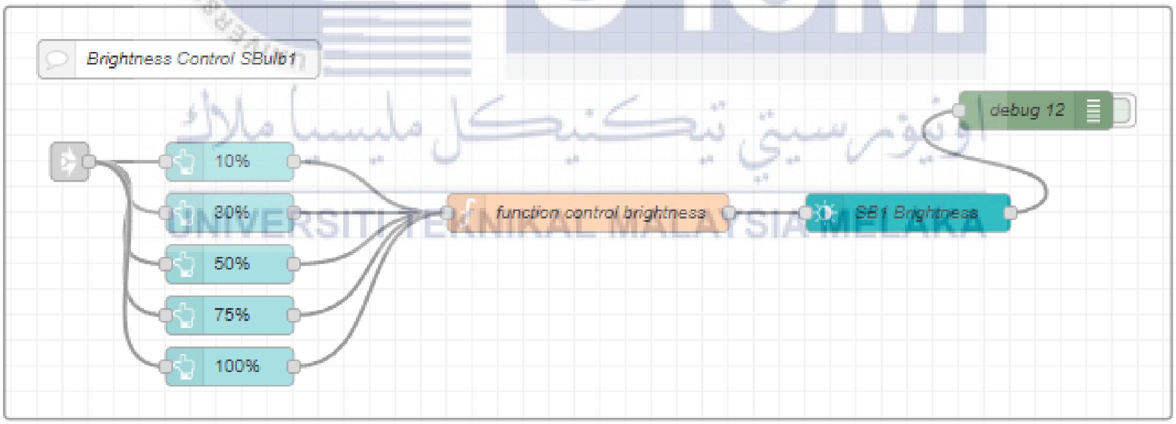


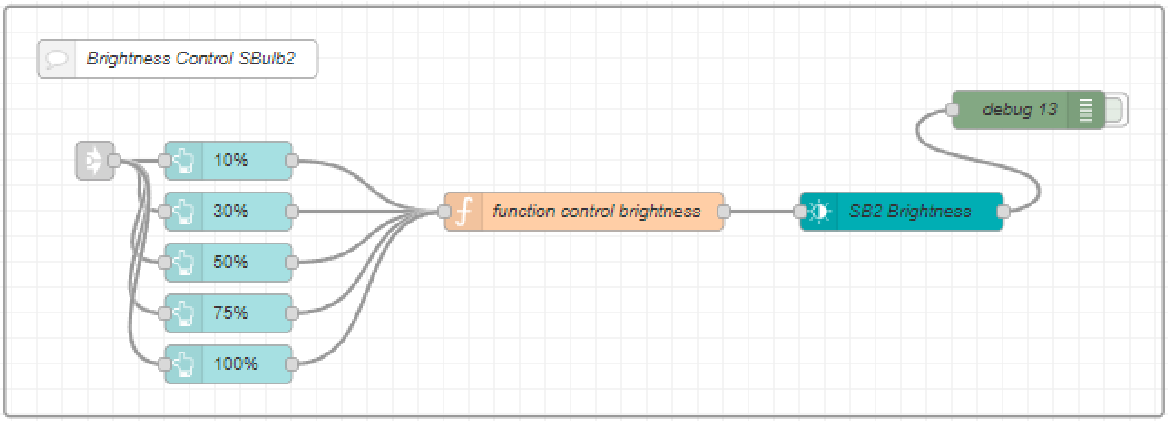


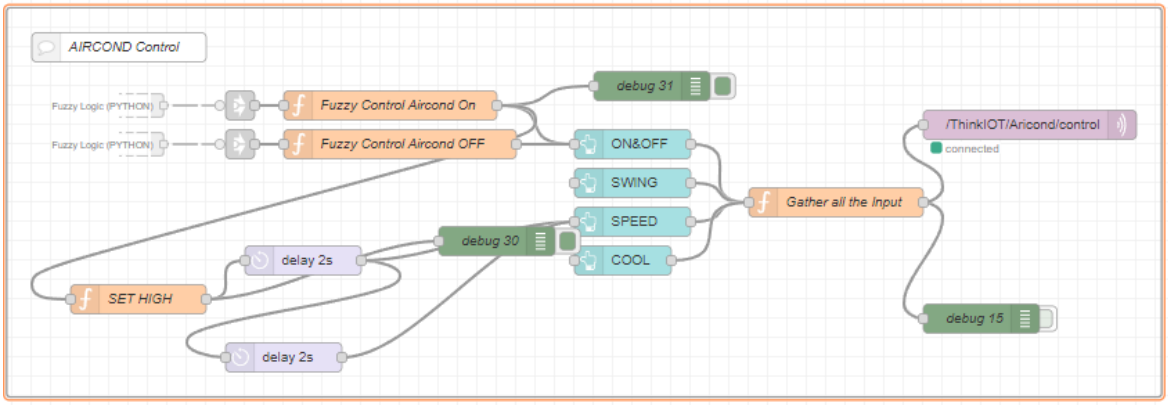

For Energy Monitoring on One Parameter Only
Aircond Energy Only
Lighting Energy Only
Total Energy
/home/ridzwan/logs/Chart1.log
/home/ridzwan/logs/Chart2.log
json
Process Data Sp1 10 minutes
Process Data SP1 1 Day
Process Data SP2 10 minutes
Process Data SP2 1 Day
Process Data Total Energy 10 minutes
Process Data Total Energy 1 Day
Energy Chart (per 10 minutes)
Energy Chart (per 1 Day)

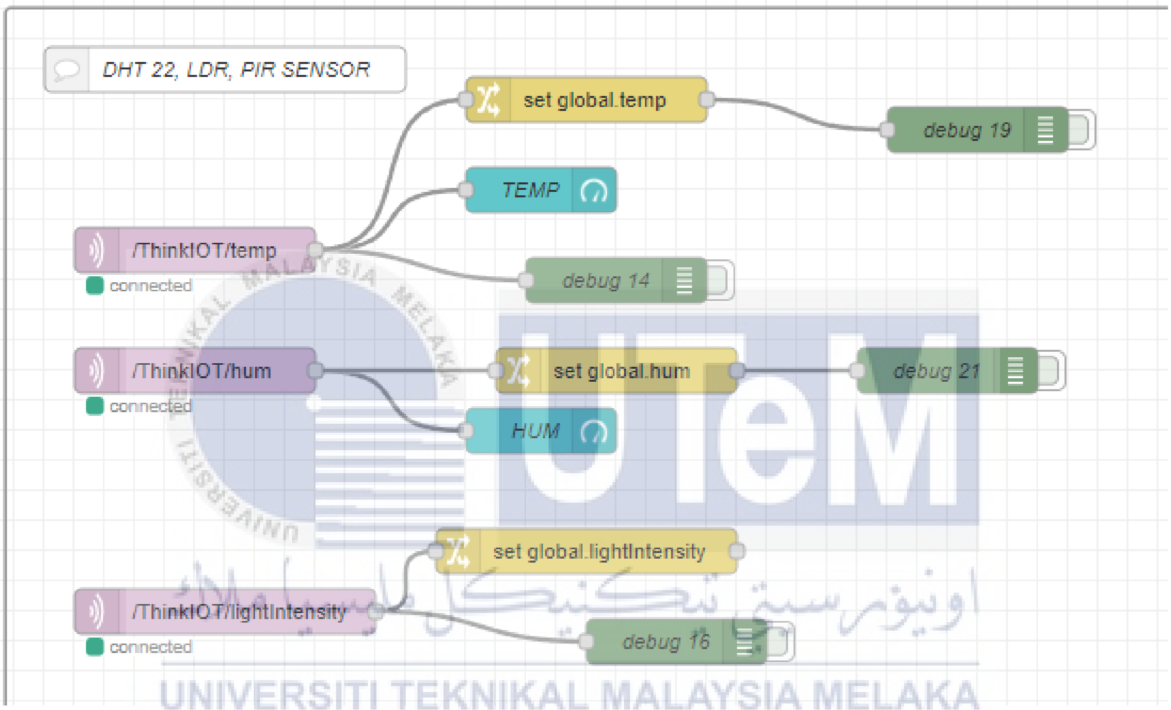

Button SPlug1 Toggle
turn on SP1
Completed.
debug 2
turn on balik "button off"
ON Aircond
turn off "button on"
OFF Aircond
turn on balik "button on"
turn off "button off"
turn off SP1
debug 1
Completed.

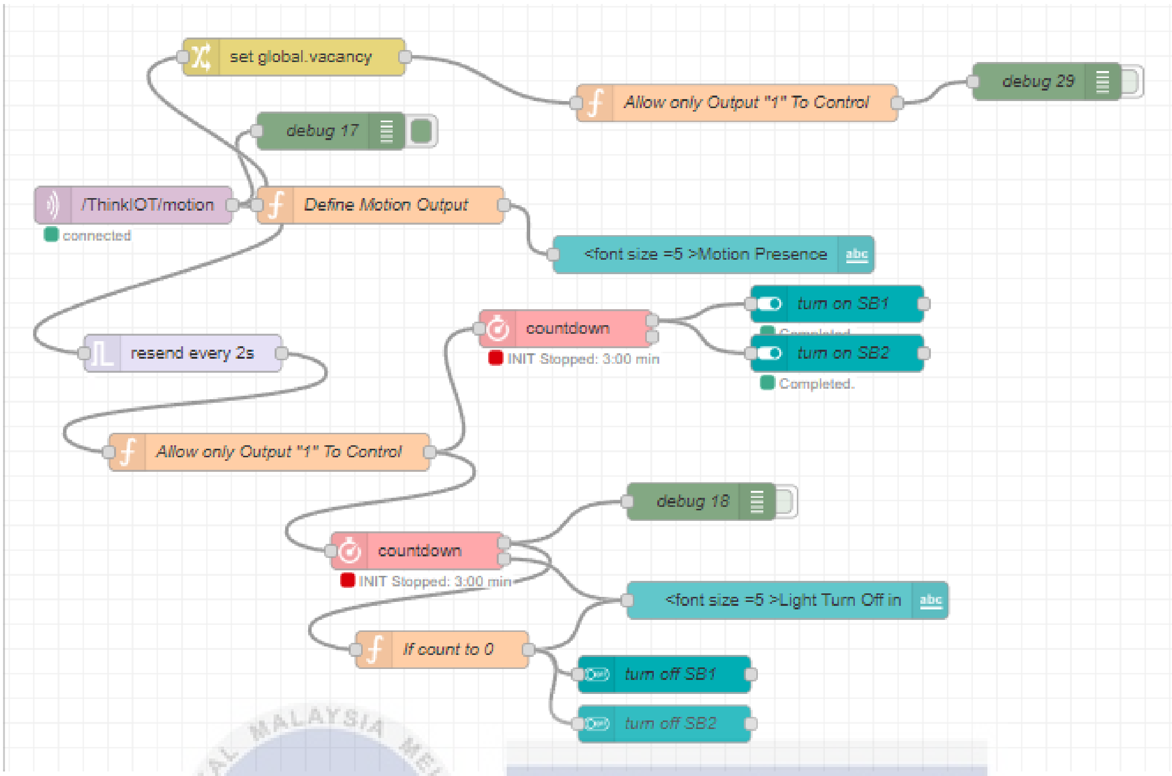

Button SBulb2 Toggle
turn on SB2
Completed.
debug 10
turn on balik "button off"
ON Light 2
turn off "button on"
OFF Light 2
turn on balik "button on"
turn off "button off"
turn off SB2
debug 11
Completed.

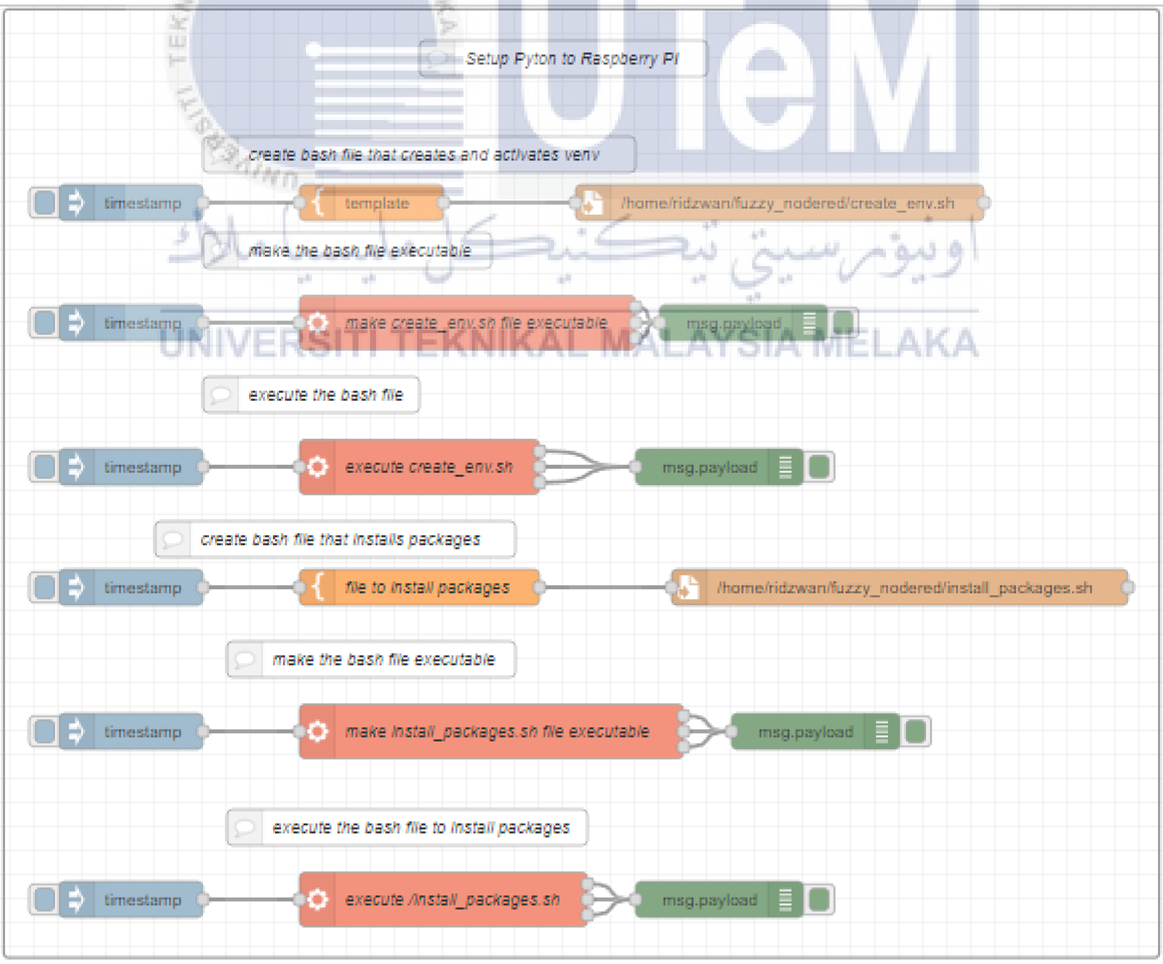

Brightness Control SBulb1
debug 12
10%
30%
50%
75%
100%
function control brightness
SB1 Brightness

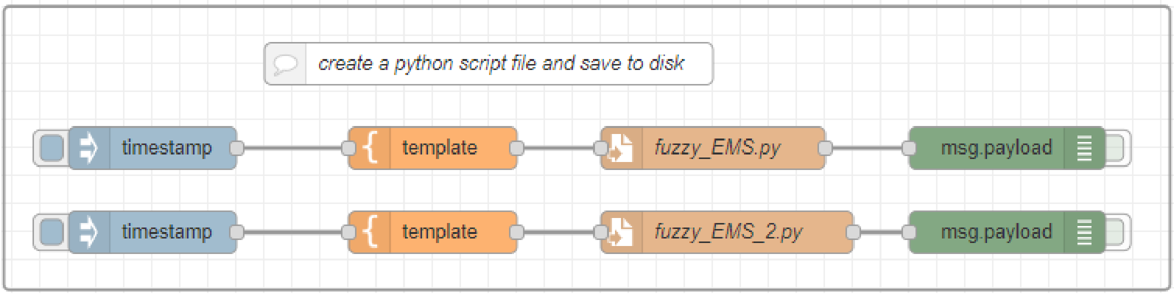

Brightness Control SBulb2
debug 13
10%
30%
50%
75%
100%
function control brightness
SB2 Brightness

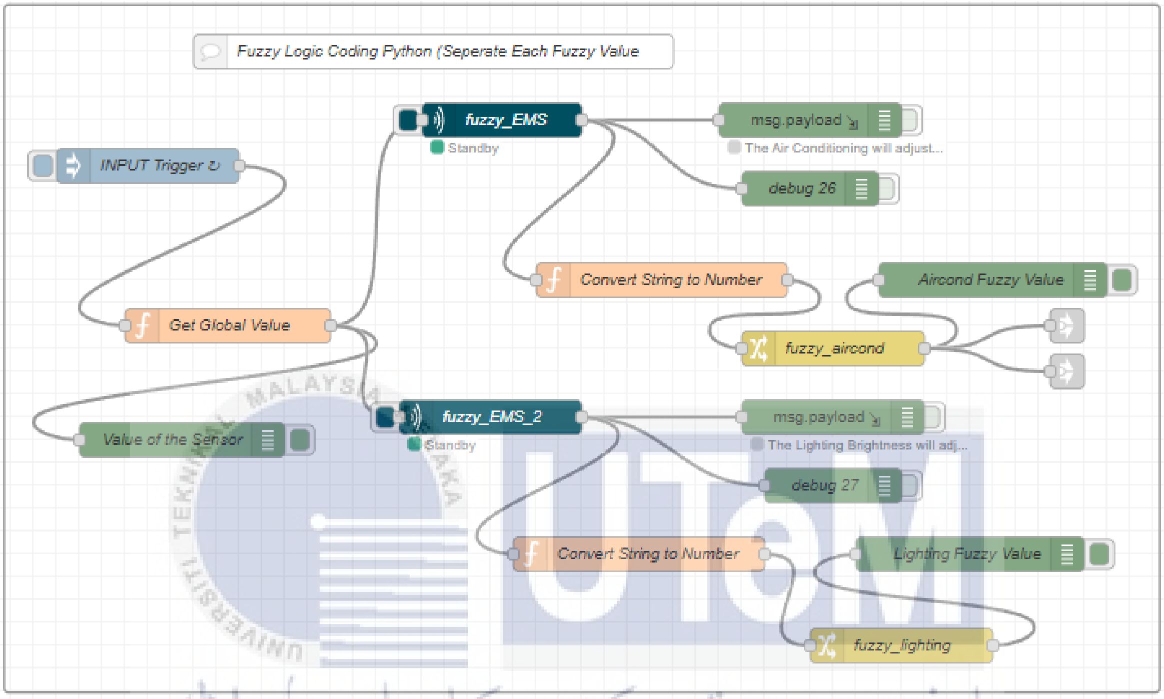

AIRCOND Control
Fuzzy Control Aircond On
Fuzzy Control Aircond OFF
debug 31
ON&OFF
SWING
SPEED
COOL
Gather all the Input
/ThinkIOT/Aricond/control
connected
debug 30
delay 2s
SET HIGH
delay 2s
debug 15


DHT 22, LDR, PIR SENSOR
set global.temp
debug 19
TEMP
/ThinkIOT/temp
connected
debug 14
/ThinkIOT/hum
connected
set global.hum
debug 21
HUM
set global.lightIntensity
/ThinkIOT/lightIntensity
connected
debug 16


set global.vacancy
Allow only Output "1" To Control
debug 29
debug 17
/ThinkIOT/motion
connected
Define Motion Output
<font size =5 >Motion Presence
abc
countdown
INIT Stopped: 3:00 min
turn on SB1
turn on SB2
Completed.
resend every 2s
Allow only Output "1" To Control
debug 18
countdown
INIT Stopped: 3:00 min
<font size =5 >Light Turn Off in
abc
If count to 0
turn off SB1
turn off SB2


Setup Pyton to Raspberry PI
create bash file that creates and activates venv
timestamp
template
/home/ridzwan/fuzzy_nodered/create_env.sh
make the bash file executable
timestamp
make create_env.sh file executable
msg.payload
execute the bash file
timestamp
execute create_env.sh
msg.payload
create bash file that installs packages
timestamp
file to install packages
/home/ridzwan/fuzzy_nodered/install_packages.sh
make the bash file executable
timestamp
make install_packages.sh file executable
msg.payload
execute the bash file to install packages
timestamp
execute /install_packages.sh
msg.payload


create a python script file and save to disk
timestamp
template
fuzzy_EMS.py
msg.payload
timestamp
template
fuzzy_EMS_2.py
msg.payload


Fuzzy Logic Coding Python (Seperate Each Fuzzy Value
fuzzy_EMS
Standby
msg.payload
The Air Conditioning will adjust...
INPUT Trigger
debug 26
Convert String to Number
Aircond Fuzzy Value
Get Global Value
fuzzy_aircond
fuzzy_EMS_2
Standby
msg.payload
The Lighting Brightness will adj...
Value of the Sensor
debug 27
Convert String to Number
Lighting Fuzzy Value
fuzzy_lighting

APPENDIX D: Developing ESP32 Firmware with the Arduino IDE for Integration with Node-RED

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <DHTesp.h>
#include <IRremoteESP8266.h>
#include <IRsend.h>
// Calibration time for the sensor (10-60 secs according to the datasheet)
int calibrationTime = 30;
// DHT Sensor
const int DHT_PIN = 25;
DHTesp dht;
// LDR and PIR Sensor
const int LDR_PIN = 32;
const int pirPin = 26; // PIR sensor pin
const int ledPin = 27; // LED pin for physical notification
unsigned long lastMsg = 0;
boolean motionDetected = false;
unsigned long motionTimer = 0;
const unsigned long motionDelay = 10000; // 10 seconds delay for motion detection

// WiFi and MQTT settings
const char* ssid = "XXXXXXXXXX"; // WiFi SSID
const char* password = "XXXXXXXXX"; // WiFi Password
const char* mqtt_server = "test.mosquitto.org"; // Mosquitto server URL

WiFiClient espClient;
PubSubClient client(espClient);

// IR Transmitter setup
const uint16_t IrPin = 18;  // GPIO pin for the IR transmitter
IRsend irsend(IrPin);
uint16_t powerOn[] = {8930, 4550,  632, 1618,  634, 522,  586, 544,  608, 520,  610, 520,
588, 544,  610, 522,
                    610, 520,  588, 544,  610, 1622,  608, 1644,  608, 1644,  634, 1622,  626,
1628,  608, 1644,  610, 1640,
                    634, 1622,  612, 542,  586, 1644,  632, 1620,  632, 1622,  608, 546,  608,
522,  608, 522,  584, 546,  590, 1638,
                    632, 522,  586, 546,  608, 522,  594, 1636,  608, 1646,  628, 1626,  606};  //
NEC 807FB847

uint16_t Swing[] = {8950, 4512,  558, 1714,  516, 594,  552, 580,  536, 596,  538, 590,
562, 568,  562, 570,  538, 594,  558, 572,
                    552, 1722,  526, 1726,  538, 1714,  518, 1736,  528, 1724,  516, 1740,  520,
1732,  534, 578,  536, 594,  558, 574,
                     558, 1714,  518, 1736,  540, 574,  546, 582,  560, 570,  560, 1714,  540,
1714,  522, 1732,  538, 570,  560, 570,
```

```
                    560, 1714,  528, 1724,  540, 1714,  538, 40128,  8972, 2284,  522};  //
NEC 807F18E7

uint16_t Speed[] = {8966, 4510,  560, 1714,  538, 570,  556, 574,  560, 570,  562, 570,
560, 570,  560, 570,  560, 570,  562, 570,  560,
                1714,  524, 1730,  540, 1714,  538, 1716,  538, 1714,  520, 1732,  522, 1730,
518, 592,  562, 1712,  540, 570,  564,
                 1712,  540, 1714,  540, 572,  544, 588,  560, 568,  562, 1710,  542, 572,  558,
1714,  538, 572,  560, 570,  560, 1714,
                  540, 1714,  538, 1714,  518};  // NEC 807F58A7

uint16_t Mode[] = {8940, 4542,  538, 1714,  538, 574,  556, 574,  536, 596,  534, 596,
558, 574,  558, 574,  534, 596,  560, 572,  558, 1716,
                538, 1714,  538, 1718,  538, 1714,  540, 1714,  514, 1738,  540, 1716,  538,
572,  560, 572,  546, 586,  560, 1714,  538, 572,
                 540, 592,  550, 582,  560, 570,  556, 1718,  516, 1736,  514, 1738,  540, 572,
558, 1716,  538, 1714,  540, 1714,  520, 1734,
                  538, 40142,  8930, 2312,  512};  // NEC 807F10EF

uint16_t Cool[] = {8974, 4512,  560, 1712,  530, 582,  548, 582,  548, 582,  538, 594,  560,
570,  562, 570,  542, 588,  560, 570,  548, 1726,
                518, 1738,  538, 1712,  540, 1714,  538, 1712,  516, 1738,  538, 1714,  538,
572,  558, 1714,  540, 570,  560, 1714,  540, 572,
                 560, 570,  538, 594,  558, 574,  562, 1712,  518, 594,  560, 1712,  540, 572,
544, 1730,  528, 1726,  520, 1730,  542, 1714,
                  516, 40182,  8952, 2308,  518};  // NEC 807F50AF

uint16_t Breeze[] = {8962, 4518,  540, 1732,  522, 586,  560, 570,  552, 578,  558, 572,
538, 592,  560, 570,  536, 594,  538, 594,  558, 1716,
                538, 1714,  538, 1716,  532, 1720,  538, 1714,  518, 1738,  516, 1736,  540,
1714,  538, 570,  538, 1736,  534, 578,  546, 586,
                 558, 574,  536, 594,  536, 596,  538, 594,  536, 1738,  516, 594,  538, 1734,
520, 1734,  542, 1712,  542, 1710,  520, 1734,
                  518, 40162,  8928, 2290,  538};  // NEC 807FA05F




void setup_wifi() {
  delay(10);
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
```

```
  }

  randomSeed(micros());
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
void callback(String topic, byte* message, unsigned int length) {
  Serial.print("Message arrived on topic: ");
  Serial.print(topic);
  Serial.print(". Message: ");
  String messageInfo;

  for (int i = 0; i < length; i++) {
    Serial.print((char)message[i]);
    messageInfo += (char)message[i];
  }
  Serial.println();

  // If a message is received on the topic room/lamp, you check if the message is either on
or off. Turns the lamp GPIO according to the message
  if(topic=="/ThinkIOT/Aricond/control"){
      Serial.print("Aircond Command");
      if(messageInfo == "on&off"){
        Serial.print("On");
         irsend.sendRaw(powerOn, sizeof(powerOn) / sizeof(powerOn[0]), 38);
      }
      else if(messageInfo == "swing"){
        Serial.print("Swing");
        irsend.sendRaw(Swing, sizeof(Swing) / sizeof(Swing[0]), 38);
      }
       else if(messageInfo == "speed"){
        Serial.print("Speed");
        irsend.sendRaw(Speed, sizeof(Speed) / sizeof(Speed[0]), 38);
      }
       else if(messageInfo == "cool"){
        Serial.print("Cool");
        irsend.sendRaw(Cool, sizeof(Cool) / sizeof(Cool[0]), 38);
      }
  }
  Serial.println();
}


void reconnect() {
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    String clientId = "ESP32Client-";
    clientId += String(random(0xffff), HEX);
```

```
    if (client.connect(clientId.c_str())) {
      Serial.println("Connected");
      client.publish("/ThinkIOT/Publish", "Welcome");
      client.subscribe("/ThinkIOT/Aricond/control");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      delay(5000);
    }}
}

// Setup function
void setup() {
  Serial.begin(115200);
  irsend.begin();  // Initialize IR transmitter
  setup_wifi();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
  dht.setup(DHT_PIN, DHTesp::DHT22);

  pinMode(LDR_PIN, INPUT);  // Set LDR_PIN as an input
  pinMode(pirPin, INPUT);   // Set pirPin as an input
  pinMode(ledPin, OUTPUT);  // Set ledPin as an output
  digitalWrite(pirPin, LOW);

 /*  // Give the sensor some time to calibrate
  Serial.print("Calibrating sensor");
  for (int i = 0; i < calibrationTime; i++) {
    Serial.print(".");
    delay(1000);
  }
  Serial.println(" done");
  Serial.println("SENSOR ACTIVE");
  delay(50); */
}


int getLightIntensity(int analogValue) {
  // Inverting the mapping as 0 is bright light and 4500 is darkness
  // Mapping these to a scale of 0 to 1000 lux
  return map(analogValue, 0, 4095,0, 4095); //return map(analogValue, 0, 4095, ##tukar
sini untuk dptkn actual measurement, 0);
}



// Main loop
void loop() {
  if (!client.connected()) {
```

```
    reconnect();
  }
  client.loop();

 // Handle motion detection
  boolean currentMotion = digitalRead(pirPin) == HIGH;

  if (currentMotion) {
    if (!motionDetected) {
      // Motion detected for the first time
      client.publish("/ThinkIOT/motion", "1"); // Publish motion detected
      digitalWrite(ledPin, HIGH); // Turn on the LED
    }
    motionDetected = true;
    motionTimer = millis(); // Reset timer whenever motion is detected
  } else if (motionDetected && millis() - motionTimer > motionDelay) {
    // Motion has stopped for more than 10 seconds
    motionDetected = false;
    client.publish("/ThinkIOT/motion", "0"); // Publish motion ended
    digitalWrite(ledPin, LOW); // Turn off the LED
  }
  unsigned long now = millis();
  if (now - lastMsg > 2000) { // Publish data every 2 seconds
    lastMsg = now;

    TempAndHumidity data = dht.getTempAndHumidity();

    // Publish temperature and humidity
    String temp = String(data.temperature, 2);
    client.publish("/ThinkIOT/temp", temp.c_str());
    String hum = String(data.humidity, 1);
    client.publish("/ThinkIOT/hum", hum.c_str());

    // Read LDR value and convert to light intensity
    int ldrValue = analogRead(LDR_PIN);
    int lightIntensity = getLightIntensity(ldrValue);
    String lightIntensityString = String(lightIntensity);
    client.publish("/ThinkIOT/lightIntensity", lightIntensityString.c_str());
    // Print sensor values to the serial monitor
    Serial.print("Temperature: ");
    Serial.println(temp);
    Serial.print("Humidity: ");
    Serial.println(hum);
    Serial.print("Light Intensity: ");
    Serial.println(lightIntensityString);
  }
}
```

APPENDIX E: Developing Fuzzy Logic Rule-Based System in Python for Raspberry Pi Integration with Node-RED

```
import sys
import json
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl

#THE ANTECEDENT
temperature = ctrl.Antecedent(np.arange(15, 32, 1), 'temperature')
temperature['low'] = fuzz.trapmf(temperature.universe, [15, 15, 19, 22])
temperature['moderate'] = fuzz.trapmf(temperature.universe, [21, 23, 25, 27])
temperature['high'] = fuzz.trapmf(temperature.universe, [26, 28, 31, 31])



# 2. Light Intensity (Lux) - In LDR mode (4095 is darkest, 0 is brightest)
light_intensity = ctrl.Antecedent(np.arange(0, 4096, 1), 'light_intensity')
light_intensity['low'] = fuzz.trapmf(light_intensity.universe, [3000, 3500, 4095, 4095])
light_intensity['moderate'] = fuzz.trimf(light_intensity.universe, [1500, 2500, 3500])
light_intensity['high'] = fuzz.trapmf(light_intensity.universe, [0, 0, 200, 2000])


# 3. Vacancy
vacancy = ctrl.Antecedent(np.arange(0, 2, 1), 'vacancy')
vacancy['vacant'] = fuzz.sigmf(vacancy.universe, 0.5, -10)  # Sigmoidal shape
vacancy['occupied'] = fuzz.sigmf(vacancy.universe, 0.5, 10)  # Sigmoidal shape


# 4. Air Conditioning
air_conditioning = ctrl.Consequent(np.arange(0, 6, 1), 'air_conditioning')
air_conditioning['off'] = fuzz.trimf(air_conditioning.universe, [0, 0, 1])
air_conditioning['low'] = fuzz.trimf(air_conditioning.universe, [1, 2, 3])
air_conditioning['high'] = fuzz.trimf(air_conditioning.universe, [3, 4, 5])


# 5. Lighting
lighting = ctrl.Consequent(np.arange(0, 5, 1), 'lighting')
lighting['off'] = fuzz.trapmf(lighting.universe, [0, 0, 0, 1])
lighting['dim'] = fuzz.trimf(lighting.universe, [1, 2, 3])
lighting['bright'] = fuzz.trimf(lighting.universe, [3, 4, 5])


##########THE FUZZY LOGIC RULES####################
#DEFINING RULES FOR THE AIRCONDITIONING
Rule_1 = ctrl.Rule(temperature['high'] & vacancy['occupied'], air_conditioning['high'])
Rule_2 = ctrl.Rule(temperature['moderate'] & vacancy['occupied'], air_conditioning['low'])
```

```
Rule_3 = ctrl.Rule(temperature['low'] & light_intensity['low'] & vacancy['occupied'],
air_conditioning['off'])
Rule_4 = ctrl.Rule(temperature['low'] & light_intensity['moderate'] & vacancy['occupied'],
air_conditioning['off'])
Rule_5 = ctrl.Rule(temperature['low'] & light_intensity['high'] & vacancy['occupied'],
air_conditioning['off'])
Rule_6 = ctrl.Rule(temperature['moderate'] & light_intensity['high'] & vacancy['occupied'],
air_conditioning['low'])
Rule_7 = ctrl.Rule(temperature['high'] & light_intensity['moderate'] & vacancy['occupied'],
air_conditioning['high'])
Rule_8 = ctrl.Rule(temperature['high'] & light_intensity['high'] & vacancy['occupied'],
air_conditioning['high'])
Rule_9 = ctrl.Rule(vacancy['vacant'] & temperature['low'] & light_intensity['low'] ,
air_conditioning['off'])
Rule_10= ctrl.Rule(vacancy['vacant'] & temperature['moderate'] & light_intensity['low'] ,
air_conditioning['off'])
Rule_11= ctrl.Rule(vacancy['vacant'] & temperature['high'] & light_intensity['low'] ,
air_conditioning['off'])
Rule_12 = ctrl.Rule(vacancy['vacant'] & temperature['low'] & light_intensity['moderate'] ,
air_conditioning['off'])
Rule_13= ctrl.Rule(vacancy['vacant'] & temperature['moderate'] &
light_intensity['moderate'] , air_conditioning['off'])
Rule_14= ctrl.Rule(vacancy['vacant'] & temperature['high'] & light_intensity['moderate'] ,
air_conditioning['off'])
Rule_15 = ctrl.Rule(vacancy['vacant'] & temperature['low'] & light_intensity['high'] ,
air_conditioning['off'])
Rule_16= ctrl.Rule(vacancy['vacant'] & temperature['moderate'] & light_intensity['high'] ,
air_conditioning['off'])
Rule_17= ctrl.Rule(vacancy['vacant'] & temperature['high'] & light_intensity['high'] ,
air_conditioning['off'])
Rule_18 = ctrl.Rule(temperature['moderate'] & light_intensity['moderate'] &
vacancy['occupied'], air_conditioning['low'])
Rule_19 = ctrl.Rule(temperature['moderate'] & light_intensity['low'] &
vacancy['occupied'], air_conditioning['low'])
Rule_20 = ctrl.Rule(temperature['high'] & light_intensity['low'] & vacancy['occupied'],
air_conditioning['high'])

Rule_aircond= [Rule_1, Rule_2, Rule_3, Rule_4, Rule_5, Rule_6, Rule_7, Rule_8,
Rule_9,
          Rule_10, Rule_11, Rule_12, Rule_13, Rule_14, Rule_15, Rule_16,
Rule_17,Rule_18,Rule_19,Rule_20]
a_aircond = ctrl.ControlSystem(Rule_aircond)       #Base class to contain a Fuzzy Control
System.
b_aircond= ctrl.ControlSystemSimulation(a_aircond)  #Calculate results from a
ControlSystem.


#DEFINING RULES FOR THE LIGHTING
Rule_1_1 = ctrl.Rule(temperature['low'] & light_intensity['low'] & vacancy['occupied'],
lighting['bright'])
```

```
Rule_1_2 = ctrl.Rule(temperature['moderate'] & light_intensity['low'] &
vacancy['occupied'], lighting['bright'])
Rule_1_3 = ctrl.Rule(temperature['high'] & light_intensity['low'] & vacancy['occupied'],
lighting['bright'])

Rule_1_4 = ctrl.Rule(temperature['low'] & light_intensity['moderate'] &
vacancy['occupied'], lighting['dim'])
Rule_1_5 = ctrl.Rule(temperature['moderate'] & light_intensity['moderate'] &
vacancy['occupied'], lighting['dim'])
Rule_1_6 = ctrl.Rule(temperature['high'] & light_intensity['moderate'] &
vacancy['occupied'], lighting['dim'])

Rule_1_7 = ctrl.Rule(temperature['low'] & light_intensity['high'] & vacancy['occupied'],
lighting['off'])
Rule_1_8 = ctrl.Rule(temperature['moderate'] & light_intensity['high'] &
vacancy['occupied'], lighting['off'])
Rule_1_9 = ctrl.Rule(temperature['high'] & light_intensity['high'] & vacancy['occupied'],
lighting['off'])

Rule_1_10 = ctrl.Rule(temperature['low'] & light_intensity['low'] & vacancy['vacant'],
lighting['off'])
Rule_1_11 = ctrl.Rule(temperature['moderate'] & light_intensity['low'] &
vacancy['vacant'], lighting['off'])
Rule_1_12 = ctrl.Rule(temperature['high'] & light_intensity['low'] & vacancy['vacant'],
lighting['off'])

Rule_1_13 = ctrl.Rule(temperature['low'] & light_intensity['moderate'] &
vacancy['vacant'], lighting['off'])
Rule_1_14 = ctrl.Rule(temperature['moderate'] & light_intensity['moderate'] &
vacancy['vacant'], lighting['off'])
Rule_1_15 = ctrl.Rule(temperature['high'] & light_intensity['moderate'] &
vacancy['vacant'], lighting['off'])

Rule_1_16 = ctrl.Rule(temperature['low'] & light_intensity['high'] & vacancy['vacant'],
lighting['off'])
Rule_1_17 = ctrl.Rule(temperature['moderate'] & light_intensity['high'] &
vacancy['vacant'], lighting['off'])
Rule_1_18 = ctrl.Rule(temperature['high'] & light_intensity['high'] & vacancy['vacant'],
lighting['off'])



# Add more rules as needed

# Combine all rules into a list
Rule_lighting = [Rule_1_1, Rule_1_2, Rule_1_3, Rule_1_4, Rule_1_5, Rule_1_6,
Rule_1_7, Rule_1_8, Rule_1_9, Rule_1_10,
                Rule_1_11, Rule_1_12, Rule_1_13, Rule_1_14, Rule_1_15, Rule_1_16,
Rule_1_17, Rule_1_18]
```

```
# Create the control system and simulation object
a_lighting = ctrl.ControlSystem(Rule_lighting)
b_lighting = ctrl.ControlSystemSimulation(a_lighting)



line = sys.stdin.readline().strip()
if line:
    try:
        data = json.loads(line)  # Attempt to parse the JSON string
    except json.JSONDecodeError:
        print("Error: Invalid JSON format")
    else:
        # Extract and set the fuzzy inputs using the exact key names from the JSON
        temperature_value = data.get('temperature', 0)
        light_intensity_value = data.get('lightIntensity', 0)  # Changed to match JSON key
        vacancy_value = data.get('vacancy', 0)

        # Set inputs for air conditioning system
        b_aircond.input['temperature'] = temperature_value
        b_aircond.input['light_intensity'] = light_intensity_value
        b_aircond.input['vacancy'] = vacancy_value
        b_aircond.compute()  # Compute the fuzzy system

        # Output for air conditioning
        output_aircond = b_aircond.output['air_conditioning']
        print(f"The Air Conditioning will adjust to: {output_aircond:.4f}")

        # Set inputs for lighting system
        b_lighting.input['temperature'] = temperature_value
        b_lighting.input['light_intensity'] = light_intensity_value
        b_lighting.input['vacancy'] = vacancy_value
        b_lighting.compute()  # Compute the fuzzy system

        # Output for lighting
        output_lighting = b_lighting.output['lighting']
        print(f"The Lighting Brightness will adjust to: {output_lighting:.4f}")
```

APPENDIX F: Simulating Fuzzy Logic Rule-Based System in Python by mapping the Fuzzy Logic Outputs to Energy Consumption Values

```
import pandas as pd

# Define the functions for calculating energy consumption
def lighting_energy_consumption(brightness):
    if 0 <= brightness < 0.5:
        return 1.3743  # Off
    elif 0.5 <= brightness < 1:
        return 2.7633  # 10%
    elif 1 <= brightness < 2:
        return 5.0050  # 30%
    elif 2 <= brightness < 3:
        return 7.2777  # 50%
    elif 3 <= brightness < 4:
        return 10.2527  # 75%
    else:
        return 12.9123  # 100%

def air_conditioning_energy_consumption(mode):
    if 0 <= mode < 1:
        return 0.6887  # Off
    elif 1 <= mode < (5/3):
        return 47.6710  # Low
    elif (5/3) <= mode < (7/3):
        return 51.1773  # Low (Cold Mode On)
    elif (7/3) <= mode < 3:
        return 53.7267  # Med
    elif 3 <= mode < (11/3):
        return 57.4893  # Med (Cold Mode On)
    elif (11/3) <= mode < (13/3):
        return 59.0827  # High
    else:
        return 62.6360  # High (Cold Mode On)

# Load the existing data
file_path = 'C:/Users/User/Documents/UTEM IWAAAANNNNNN/File Laptop
UteM/YEAR 4 SEM 2 SESSION 2023 2024/FINAL YEAR PROJECT II
BEKU4894/CODING FUZZY LOGIC IWAN/Results Fuzzy
Logic/energy_management_results.csv'
data = pd.read_csv(file_path)

# Apply the energy consumption calculations
data['lighting_energy'] = data['lighting_output'].apply(lighting_energy_consumption)
data['ac_energy'] =
data['air_conditioning_output'].apply(air_conditioning_energy_consumption)
```

```
# Calculate the total energy consumption
data['total_energy'] = data['lighting_energy'] + data['ac_energy']

# Save the updated data to a new CSV file
output_file_path = 'C:/Users/User/Documents/UTEM IWAAAANNNNNN/File Laptop UteM/YEAR 4 SEM 2 SESSION 2023 2024/FINAL YEAR PROJECT II BEKU4894/CODING FUZZY LOGIC IWAN/Results Fuzzy Logic/updated_energy_management_results.csv'
data.to_csv(output_file_path, index=False)
```