An Intelligent Electricity Management Unit: AI-Driven Power Forecasting and Personalized Consumption Insights with Application Integration

R25-065

Project Proposal Report

Pivithuru N.H.A.S.

BSc (Hons) in Information Technology Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

January 2025

An Intelligent Electricity Management Unit: AI-Driven Power Forecasting and Personalized Consumption Insights with Application Integration

R25-065

Project Proposal Report

Pivithuru N.H.A.S.

BSc (Hons) in Information Technology Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

January 2025

Declaration

I declare that this is our own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made the text.

2025/02/02

………………………………… …………………………………

Signature Date

03/02/25

………………………………… …………………………………

Signature of the Supervisor Date

2025/02/02

………………………………… …………………………………

Signature of the Co-Supervisor Date

Abstract

To minimize waste and maximize power consumption, energy management in residential settings is becoming more and more crucial. A smart Home Energy Management System (HEMS) is presented in this research project, allowing users to efficiently monitor and manage their household's energy consumption. The solution consists of an Internet of Things (IoT) plug-in unit that serves as a power supply middleman between household electrical gadgets and a wall outlet. These units collect real-time energy consumption data and transmit it to a cloud-based system for further analysis.

The system uses generative AI to offer tailored recommendations to optimize energy usage on a device-by-device basis, while machine learning techniques are used to analyze time series data and forecast future patterns of energy consumption.

The main interface via which users connect and control several IoT devices is a web application constructed with a React frontend and a Laravel backend. Users can establish electricity consumption limitations for each device, track energy usage, and get actionable insights using the app. With the ability to manually reactivate via the app, the IoT unit automatically switches off power to the corresponding device if the predetermined energy limit is exceeded. This system provides a scalable and easy-to-use way to encourage energy conservation, assisting homes in cutting expenses and supporting sustainable energy usage habits.

Keywords: Home Energy Management System, Internet of Things, Generative AI, Machine Learning, Web Application

Table Of Contents

List Of Figures

List Of Tables

List Of Abbreviations

Abbreviation	Definition
HEMS	Home Energy Management System
IoT	Internet of Things
AI	Artificial Intelligence
Gen AI	Generative Artificial Intelligence
ML	Machine Learning

Table 1:Abbreviations

1. Introduction

1.1 Background & Literature Survey

With the increasing global demand for energy and rising electricity costs, efficient energy management has become a critical concern for households and businesses alike. The economic challenges following the COVID-19 pandemic have further intensified the need for cost-effective solutions to monitor and control electricity usage. Smart home technologies, driven by the Internet of Things (IoT), artificial intelligence (AI), and machine learning (ML), have emerged as promising solutions to address these challenges by enabling remote monitoring, automation, and optimization of energy consumption.

Existing Home Energy Management Systems (HEMS) primarily focus on overall energy consumption but lack device-level monitoring capabilities. This limitation makes it difficult for users to identify high-energy-consuming appliances and take corrective actions to optimize energy use. Moreover, rental property owners face challenges in managing energy consumption based on tenants' payments, leading to inefficiencies and potential financial losses. Current solutions do not provide mechanisms to allocate energy usage according to the amount paid by tenants, resulting in uncontrolled electricity consumption and disputes between landlords and tenants.

Several studies have explored IoT-based energy management systems and smart home automation. These studies highlight the potential of integrating renewable energy sources, such as solar and wind power, with smart home systems to reduce reliance on the main grid and lower energy costs. However, the seamless integration of these renewable sources with grid power remains a challenge, requiring advanced energy distribution and storage mechanisms. Additionally, existing literature indicates a growing need for user-friendly, customizable energy management solutions that cater to individual household needs while ensuring scalability for larger applications.

The proposed research aims to bridge these gaps by developing an IoT-based smart Home Energy Management System (HEMS) that offers device-level energy monitoring and control. The system will allow users to track individual appliance usage, set consumption limits, and optimize energy allocation based on predefined budgets, making it particularly beneficial for rental properties. Additionally, the system will facilitate the efficient integration of renewable energy sources, ensuring optimal utilization of green energy while maintaining a balance with grid power. The use of AI and ML will further enhance the system by providing predictive analytics and energy-saving suggestions based on historical usage patterns.

This study builds upon previous research efforts by addressing key limitations and introducing novel features such as tenant-specific energy allocation and real-time device-level control. By leveraging cutting-edge IoT technologies, the proposed solution aims to empower users to make informed energy decisions, reduce costs, and contribute to a more sustainable future.

1.2 Research Gap

Several studies have explored the implementation of IoT-based smart home systems and home energy management solutions. These studies primarily focus on overall energy consumption, smart automation, and the integration of renewable energy sources. However, a significant gap exists in terms of device-level energy monitoring and control, which is crucial for precise energy optimization and cost management. Existing systems provide aggregated energy data without offering insights into individual appliance usage, making it challenging for users to identify and manage high-energy-consuming devices effectively. Additionally, while simulations and theoretical models have been proposed, practical implementation challenges such as real-time performance, scalability, and security remain underexplored. Addressing these gaps can lead to more efficient and user-centric smart home energy management systems.

Study	Focus	Limitations
A Home Energy Management System (Cristina STOLOJESCU-CRISAN., 2023) [1]	The research focuses on developing a Home Energy Management System (HEMS) within the context of smart home automation. It aims to monitor and optimize household energy consumption by integrating renewable energy sources, particularly photovoltaic (solar) panels. The proposed system, controlled via the qToggle app, allows users to track energy consumption and generation in real time, promoting energy efficiency and sustainability. The paper highlights the benefits of IoT-based energy management, emphasizing cost reduction, environmental impact, and the integration of smart technology with renewable energy sources.	Despite the promising benefits of the proposed Home Energy Management System (HEMS), there are several limitations to consider. The system's performance is heavily reliant on the availability of solar energy, which can fluctuate due to environmental factors such as weather conditions and seasonal changes. Additionally, the proposed solution is primarily designed for residential use, limiting its applicability to larger-scale buildings or industrial settings. Compatibility with other smart home platforms and existing energy management infrastructures may also present challenges, as the system depends on specific IoT hardware such as ESP8266 chips and Raspberry Pi boards. Another limitation is related to security and privacy concerns; while the system avoids cloud storage, it may still face risks associated with local data storage and potential unauthorized access. Finally, the initial cost of installation and maintenance may

		pose financial challenges for some homeowners, making widespread adoption difficult.
Home Energy Management System and Optimizing Energy in Microgrid Systems (Jirawadee Polprasert., 2021) [2]	This research focuses on the development of a Home Energy Management System (HEMS) aimed at optimizing the efficiency of renewable energy utilization within residential settings. The system seeks to enhance the responsiveness of energy generation to ensure it meets household demand while minimizing reliance on energy storage solutions. By integrating renewable energy sources such as solar, wind, and water with the main power grid, the proposed HEMS enables better control over energy production, distribution, and consumption. Additionally, it emphasizes data collection and analysis to identify optimal energy management strategies that can reduce overall costs and improve sustainability.	The proposed HEMS primarily focuses on overall energy optimization but lacks the capability to manage energy usage at an individual appliance level. This limitation restricts users from identifying and controlling specific high-energy-consuming devices, which could further improve energy efficiency. Additionally, the system's effectiveness is dependent on the availability and consistency of renewable energy sources, which can be affected by weather conditions and seasonal variations. Another limitation is the potential complexity of integrating the system with existing grid infrastructures, which may require significant initial investment and technical expertise. Furthermore, the absence of real-time dynamic load balancing features may impact the system's ability to respond instantly to fluctuating energy demands within the household.
Development of IoT-based Smart Home Application with Energy Management (Prathyusha M R., 2023) [3]	This research focuses on developing an IoT-based smart home model that integrates various IoT-enabled devices to provide essential household amenities while optimizing energy consumption. The proposed system includes smart sensors and appliances strategically placed within multiple rooms to monitor and control environmental factors such as lighting, heating, and security. The study aims to analyze the daily power consumption of these smart appliances using Arduino and simulate the hardware design through the Tinkercad platform. By leveraging state-of-the-	The proposed smart home model focuses primarily on simulating and analyzing energy consumption without real-world deployment, which may limit practical insights into system performance. Additionally, the research considers a predefined set of IoT devices and sensors, restricting scalability and customization for different household needs. The study relies on simulated environments like Tinkercad, which may not account for real-life issues such as network latency, hardware malfunctions, or user behavior

	art technologies like IoT, artificial intelligence, and communication systems, the research seeks to enhance the efficiency, convenience, and automation of smart homes while addressing energy management challenges.	variations. Another limitation is the additional energy required to power IoT devices, which could offset some of the energy-saving benefits. Lastly, security and privacy concerns related to data transmission and storage are not comprehensively addressed, posing potential risks to user information.

Table 2:System Comparison

Identified Gap:

1. Lack of Device-Level Energy Monitoring: Existing systems provide overall energy consumption analysis but do not monitor individual device usage, making it difficult to optimize energy consumption at the appliance level.

2. Limited Practical Implementation: Many proposed systems rely on simulations and theoretical models rather than real-world deployment, which can lead to discrepancies in actual performance.

3. Scalability Challenges: Current solutions may not scale efficiently to larger households or buildings with diverse energy demands and multiple IoT devices.

4. High Energy Consumption of Smart Devices: IoT-enabled smart home systems often require additional power to maintain connectivity and automation, potentially offsetting energy savings.

5. Integration with Existing Grid Systems: Seamless integration of smart home energy management systems with traditional power grids remains a challenge, requiring further development to ensure efficiency and reliability.

6. Security and Privacy Concerns: Data transmission and storage in smart home systems pose potential risks, and current solutions do not comprehensively address cybersecurity threats.

7. User-Centric Customization: Existing systems lack flexibility in allowing users to customize energy management settings based on their specific preferences and usage patterns.

While existing research contributes significantly to smart home energy management and automation, it primarily focuses on overall energy consumption without addressing device-level monitoring. Practical implementation challenges, scalability issues, and security concerns remain key areas that need further investigation. Addressing these gaps can lead to the development of more efficient, secure, and user-friendly smart home energy management systems.

1.3 Research Problem

The post-COVID-19 economic downturn has led to increased financial strain on households, with electricity prices rising significantly. Managing electricity consumption efficiently has become a necessity to reduce costs and promote energy conservation. However, existing home energy management systems primarily focus on overall energy usage rather than providing detailed device-level insights. Additionally, rental accommodations, such as apartments and hostels, lack an effective system to control electricity supply based on the amount paid by tenants, leading to inefficiencies and disputes over energy usage. The integration of renewable energy sources, such as solar power, into household energy management remains another challenge due to the lack of proper energy allocation and optimization mechanisms.

Problem Statement:

Current home energy management solutions do not offer device-specific energy monitoring and control, making it difficult for users to optimize consumption and reduce costs effectively. Moreover, rental property owners face challenges in allocating electricity based on tenants' payments, resulting in overuse and financial losses. A smart system is required to provide personalized energy management, allowing users to track and control energy consumption at the device level while ensuring fair allocation for tenants and effective integration with renewable energy sources.

Key Features of the Research Problem:

1. Rising Energy Costs: Increasing electricity prices post-COVID-19 make it essential for households to track and reduce energy consumption effectively.

2. Lack of Device-Level Monitoring: Existing systems provide only aggregated data, making it difficult to identify and control high-energy-consuming appliances individually.

3. Rental Property Energy Management: No efficient solution exists to allocate electricity usage based on tenants' paid amounts, leading to disputes and wastage.

4. Integration with Renewable Energy: Households face challenges in efficiently balancing grid and renewable energy sources, limiting the full potential of sustainable energy.

5. User-Friendly Energy Allocation: A need for an intuitive system that allows users to set limits and monitor energy usage easily, both for homeowners and tenants.

Significance of the Research Problem:

Addressing these issues is crucial for enhancing energy efficiency, reducing costs, and promoting sustainable energy consumption. A smart Home Energy Management System (HEMS) that enables device-wise control, tenant-specific energy allocation, and renewable energy integration can provide significant benefits. It will empower homeowners and tenants to make informed decisions, optimize energy usage, and contribute to a more sustainable future while mitigating financial strain. The proposed system will bridge the gap in existing solutions by offering a more precise, flexible, and fair energy management approach.

2. Objectives

2.1 Main Objective

The main objective of this research is to develop an intelligent Home Energy Management System (HEMS) that integrates IoT, machine learning, and generative AI to optimize household energy consumption. The system aims to provide real-time device-level monitoring, energy consumption predictions, and personalized energy-saving suggestions while ensuring seamless interaction through a user-friendly web application. The proposed system will empower users to track, analyze, and control their energy usage efficiently, ultimately reducing costs and promoting sustainable energy practices.

2.2 Specific Objectives

1. Design and Develop an IoT-based Energy Monitoring Unit:

- Build and program an IoT-based hardware unit using Arduino to monitor real-time energy consumption of household appliances.
- Ensure seamless communication between the IoT unit and the web application for data transmission.

2. Implement Machine Learning for Time Series Analysis:

- Develop ML models to analyze energy consumption patterns and provide accurate future energy usage predictions.
- Utilize historical data to identify trends and potential areas for energy optimization.

3. Integrate Generative AI for Personalized Energy Suggestions:

- Leverage generative AI to provide customized energy-saving suggestions based on user behavior and energy consumption patterns.
- Offer actionable insights to help users optimize their energy usage and reduce wastage.

4. Develop a Web Application (Laravel + React):

- Create an interactive web-based interface to connect with the IoT unit, display real-time and predictive analytics, and visualize energy usage trends.
- Enable users to set electricity consumption limits and manually reset them when needed.
- Integrate GenAI-generated recommendations to assist users in making informed energy-saving decisions.

5. Ensure System Scalability and Usability:

- Design the system to accommodate multiple IoT units for different appliances or rooms.
- Provide an intuitive and responsive user experience across various devices.

3. Methodology

3.1 Project Overview

The proposed Home Energy Management System (HEMS) aims to provide an efficient solution for monitoring and optimizing household energy consumption. The system consists of an IoT-based energy monitoring unit that connects with a web application developed using Laravel (backend) and React (frontend). The web application serves as the central interface where users can register IoT units, monitor real-time energy usage, view machine learning-based energy consumption predictions, and receive personalized energy-saving suggestions generated by AI. The system also provides functionalities such as setting energy usage limits, restoring power supply after exceeding limits, and managing user accounts. To ensure scalability and reliability, AWS cloud services will be used to store and process data.

3.2 Feasibility Study

3.2.1 Technical Feasibility

The technical feasibility of the web application component focuses on its ability to efficiently manage communication between the IoT unit and the application, provide real-time monitoring, and offer a seamless user experience. Key technical aspects include:

1. Connectivity:

- The IoT unit will communicate with the web application via Wi-Fi, ensuring a stable and real-time data transmission.
- The web app will facilitate two-way communication, allowing users to monitor and control energy consumption remotely.

2. Product Registration:

- Users can register their IoT units through the app, which will automatically create a profile for each registered unit.
- The profile will include unit-specific data such as consumption trends, predictions, suggestions, and operational settings.

3. Technology Stack:

- The web application will be developed using React.js for the frontend, providing an interactive and responsive user experience.
- Laravel will be used for the backend to ensure robust data handling, API communication, and security.
- AWS Cloud services will be utilized for data storage, processing, and scalability, ensuring high availability and performance.

4. Unit Restoration Feature:

- In cases where the power supply is cut off due to exceeding set limits, users can restore the unit directly through the app, ensuring convenience and control.

5. User Account Management:

- The app will include user account functionalities, enabling individuals to create and manage their profiles, track their energy consumption, and set preferences.

6. Unit Profile Visualization:

- Each registered unit will have a dedicated profile where users can access all related data, including real-time analytics, predictions, suggestions, and alerts.

3.2.2 Economy Feasibility

The economic feasibility of the web application component evaluates the financial viability and cost-effectiveness of the proposed solution. Key economic considerations include:

1. Development Costs:

- The use of open-source technologies such as React and Laravel minimizes initial development costs.
- AWS cloud services provide scalable pricing models, allowing the project to start with lower costs and scale as needed.

2. Operational Costs:

- Cloud-based hosting reduces the need for on-premise infrastructure, lowering maintenance and operational expenses.
- Wi-Fi-based connectivity eliminates the need for additional networking hardware, reducing setup costs.

3. User Affordability:

- The system aims to provide cost-effective energy management solutions to end-users by offering insights that help reduce energy consumption and costs over time.

4. Scalability and ROI:

- The modular design of the web application allows for easy expansion, making it suitable for a wide range of users, from individual homeowners to rental property owners.
- The potential for energy savings through efficient monitoring and control ensures a high return on investment for end-users.

3.3 System Overview Diagram

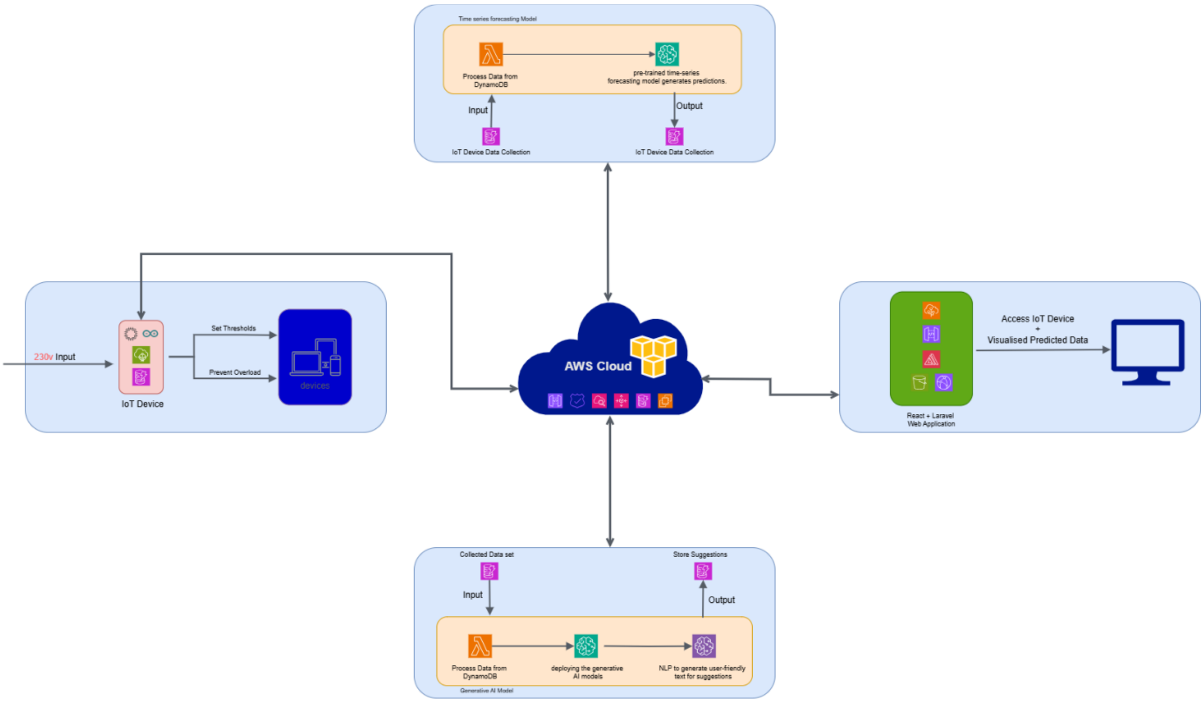

Figure 1:System Overview Diagram

3.4 Design Phase - Individual Component

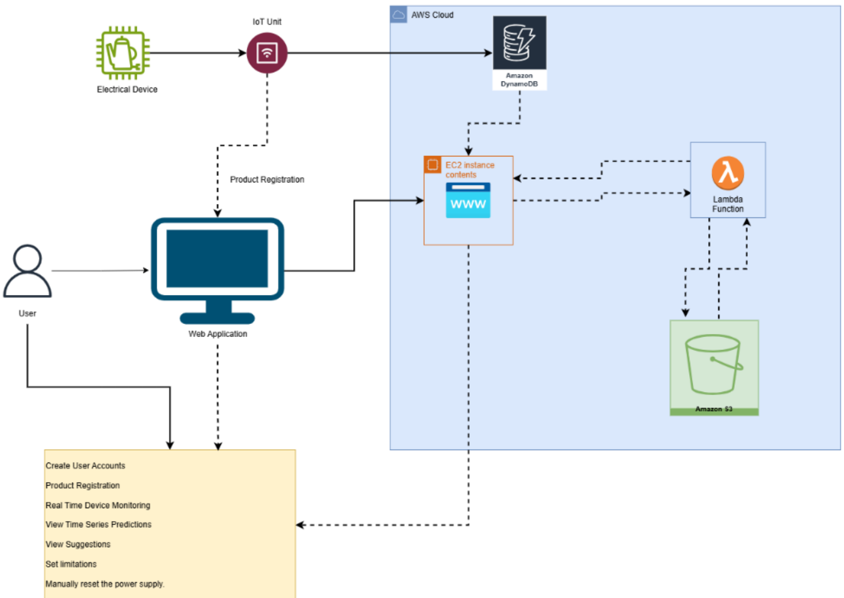

Figure 2:System Diagram for Individual Component

3.5 Project Requirements

The web application component of the Home Energy Management System (HEMS) is responsible for providing a seamless interface for users to monitor and control their household energy consumption. It connects with the IoT-based unit via Wi-Fi to collect real-time data, display machine learning-based energy predictions, and provide generative AI-based personalized suggestions. The web app, developed using React for the frontend and Laravel for the backend, offers features such as user account creation, unit registration, profile management, and energy limit control. AWS cloud services will be used to ensure scalability, data storage, and system reliability. The following functional and non-functional requirements define the key aspects necessary for the successful development and deployment of the web application.

3.5.1 Functional Requirements

The functional requirements outline the core functionalities that the web application must provide to ensure efficient interaction with the IoT unit and user management.

1. User Management:

- Allow users to create, update, and manage their accounts.
- Implement authentication and authorization (e.g., login/logout, role-based access).

2. IoT Unit Registration and Management:

- Enable users to register IoT units via the application.
- Automatically create a profile for each registered unit containing real-time and historical energy data.
- Provide the ability to view and update unit details.

3. Real-time Data Monitoring:

- Display energy consumption data received from the IoT unit in real-time.
- Visualize data trends through graphs and charts for better user insights.

4. Energy Usage Limits:

- Allow users to set electricity consumption limits via the app.

- Notify users when the limit is exceeded and provide an option to manually reset the unit.

5. Prediction and Insights:

- Integrate machine learning models to display time-series predictions of energy usage.
- Provide AI-generated personalized suggestions to optimize energy consumption.

6. Alerts and Notifications:

- Send notifications to users regarding exceeded limits, abnormal energy usage, and system updates.
- Notify users when power restoration is required.

7. AWS Cloud Integration:

- Store energy consumption data securely in the AWS cloud.
- Ensure smooth data retrieval and processing from cloud storage.

8. Profile Management:

- Allow users to view energy consumption history and personalized insights in their unit profiles.
- Provide options to update profile preferences and settings.

3.5.2 Non-Functional Requirements

The non-functional requirements define the quality attributes and constraints that the web application must meet to ensure usability, security, and performance.

1. Performance:

- The application should provide real-time data updates with minimal latency (within 2-5 seconds of data transmission).
- Must handle simultaneous connections for multiple IoT units efficiently.

2. Scalability:

- The system should support an increasing number of users and IoT units without degradation in performance.
- AWS cloud services should be leveraged for dynamic scaling based on demand.

3. Security:

- Implement secure authentication (e.g., JWT or OAuth) to protect user accounts and data.
- Ensure encrypted communication between the IoT unit, web app, and cloud storage.

4. Availability:

- Ensure 99.9% uptime through AWS hosting solutions to guarantee reliable access to the system.
- Implement redundancy to prevent data loss in case of failures.

5. User Experience:

- The web application should have a responsive design, ensuring smooth usability on desktop and mobile devices.
- Provide an intuitive and easy-to-navigate interface with a focus on energy insights.

6. Maintainability:

- The codebase should be modular and follow industry best practices to allow for easy updates and feature enhancements.
- Ensure proper documentation for both frontend and backend components.

7. Compliance:

- The system should adhere to industry standards for data privacy and security, such as GDPR and ISO/IEC 27001.

8. Data Backup and Recovery:

- Automated backup strategies should be implemented to prevent data loss and enable recovery in case of failures.

3.6 Work Breakdown Structure and Gantt Chart

3.6.1 Work Breakdown Structure

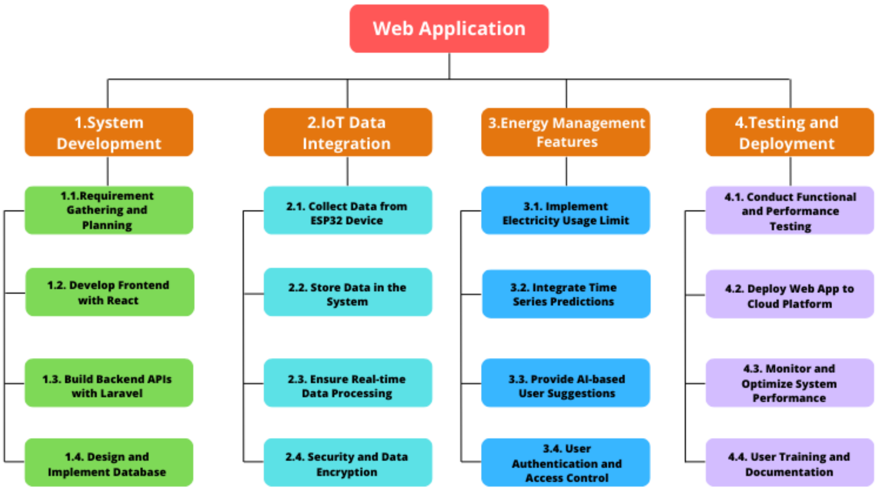

Figure 3:Work Breakdown Structure

The Work Breakdown Structure for the web application component of the Home Energy Management System (HEMS) outlines the key tasks and deliverables required to develop, deploy, and maintain the system. The WBS divides the project into manageable sections, ensuring a systematic approach to project execution. The primary goal is to create a scalable and user-friendly web application using React (frontend) and Laravel (backend) that connects with the IoT-based unit via Wi-Fi, displays predictions and suggestions, and provides various energy management features.

Work Breakdown Structure for the Web Application Component

1. Project Planning & Requirement Analysis

- Define project scope and objectives
- Gather requirements related to IoT unit connectivity and data processing
- Identify technical and economic feasibility
- Define cloud infrastructure and security requirements
- Develop project timeline and milestones

2. System Design

- Design system architecture (Frontend, Backend, Database, Cloud)
- UI/UX design for web application
- API design for communication between IoT unit and web app
- Define data models and storage structures
- Create wireframes and design mockups

3. Frontend Development (React.js)

- Set up React project and integrate UI libraries
- Develop responsive UI components (Dashboard, User profiles, etc.)
- Implement user authentication (Login, Registration)
- Implement unit registration and profile creation
- Display real-time energy monitoring graphs
- Display ML-based predictions and GenAI suggestions
- Implement energy limit settings and manual reset feature
- Integrate notifications and alerts
- Perform frontend testing and debugging

4. Backend Development (Laravel)

- Set up Laravel project and database configuration
- Develop RESTful APIs for IoT unit communication
- Implement authentication and authorization mechanisms
- Develop CRUD operations for user and unit management
- Implement real-time data processing and storage in AWS
- Integrate machine learning model for predictions

- Implement GenAI module for personalized suggestions
- Develop backup and recovery features
- Perform backend testing and API validation

5. Database Management (AWS Cloud)

- Set up AWS RDS for relational data storage
- Define database schema for users and IoT units
- Implement data security measures (encryption, access control)
- Create backup and restore strategies
- Optimize queries for performance efficiency

6. Integration and Testing

- Integrate frontend with backend APIs
- Connect IoT unit with the web application via Wi-Fi
- Conduct unit testing of individual components
- Perform system integration testing
- Conduct user acceptance testing (UAT)

7. Deployment and Maintenance

- Deploy the web application to AWS cloud
- Configure CI/CD pipelines for automated deployments
- Monitor system performance and uptime
- Provide ongoing maintenance and bug fixes
- Implement periodic feature updates based on user feedback

Summary:

The Work Breakdown Structure divides the project into logical phases, covering project planning, system design, development, testing, and deployment. It ensures a structured approach to building the web application, focusing on usability, security, and seamless interaction with the IoT unit and cloud infrastructure.

3.6.2 Gantt Chart

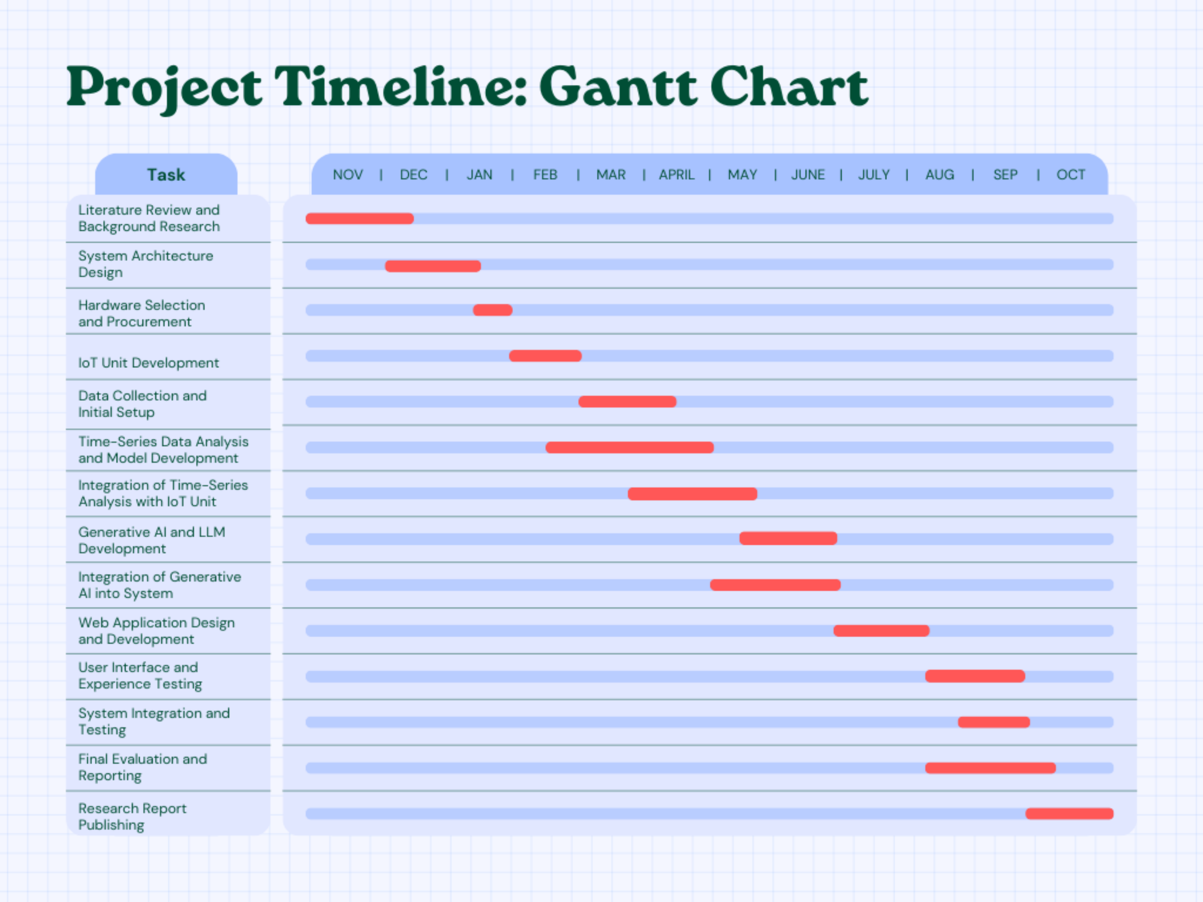

Figure 4:Gantt Chart

3.7 Deployment, Marketability and Commercialization

Deployment:

The deployment of the Home Energy Management System (HEMS) web application will be carried out in a phased approach to ensure smooth integration with the IoT unit and scalability for end-users. The web application will be hosted on AWS cloud services, ensuring high availability, scalability, and security. Key deployment steps include:

1. Cloud Deployment:

- Host the Laravel backend on AWS EC2 instances.
- Store data securely using AWS RDS (Relational Database Service).
- Utilize AWS S3 for storing logs and backups.
- Implement load balancing to manage traffic efficiently.

2. Frontend Hosting:

- Deploy the React.js frontend on AWS Amplify or S3 with CloudFront for fast global access.

3. Continuous Integration/Deployment (CI/CD):

- Implement CI/CD pipelines using GitHub Actions for automatic deployment and updates.
- Conduct regular updates to enhance performance and introduce new features.

4. Security Considerations:

- Use HTTPS for secure communication.
- Implement role-based access control (RBAC) to restrict unauthorized access.
- Perform regular security audits to address potential vulnerabilities.

5. User Support and Documentation:

- Provide user manuals and FAQs for easy onboarding.
- Offer technical support and troubleshooting via online help centers.

Marketability:

The proposed HEMS web application has significant market potential due to the increasing global demand for energy-efficient solutions. The system's ability to monitor and control energy at the device level, provide AI-based insights, and offer customized solutions for rental properties makes it attractive to a wide range of target users, including:

1. Target Audience:

- Homeowners looking to reduce energy costs and optimize consumption.
- Landlords and property managers who need tenant-based energy management solutions.
- Businesses and small offices seeking better control over energy expenses.

2. Unique Selling Points:

- Device-level Monitoring: Unlike existing solutions, our system offers detailed insights into individual appliance usage.
- Tenant-Specific Energy Allocation: Enables property owners to allocate energy based on tenant payments.
- AI-Driven Insights: Machine learning and generative AI provide actionable energy-saving recommendations.
- Remote Management: Users can monitor and control energy usage from anywhere using the web app.
- Renewable Energy Integration: Optimizes the use of solar and other renewable sources for a sustainable approach.

3. Marketing Strategies:

- Utilize digital marketing channels such as social media, SEO, and content marketing to create awareness.
- Collaborate with smart home device manufacturers for partnerships.
- Offer free trials or demo versions to attract potential customers.
- Participate in energy management expos and trade shows.

Commercialization:

To successfully bring the product to market, a strategic commercialization plan is necessary to ensure adoption and revenue generation. The commercialization strategy includes:

1. Business Model:

- Subscription-based Model: Charge users monthly or yearly fees for premium features such as advanced analytics and cloud storage.
- One-time Purchase Model: Sell the IoT unit as a one-time purchase while offering basic web app services for free.
- Freemium Model: Offer free basic monitoring with paid access to advanced features like predictive analytics and personalized AI suggestions.

2. Revenue Streams:

- Sale of IoT devices bundled with the web application.
- Subscription plans for cloud-based energy data storage and analytics.
- Partnering with energy companies to provide insights and recommendations for optimized consumption.

3. Partnerships and Collaborations:

- Collaborate with smart appliance manufacturers to integrate our system with their products.
- Work with utility companies to provide energy consumption insights to their customers.
- Seek government incentives and funding opportunities for energy-saving initiatives.

4. Scalability Plan:

- Expand the solution to support commercial and industrial energy management.
- Introduce a mobile app for better accessibility and real-time alerts.
- Continuously upgrade features based on user feedback and emerging technology trends.

Conclusion:

The HEMS web application presents a viable opportunity in the growing energy management market. By offering a comprehensive solution that integrates IoT, AI, and cloud technology, the system aims to address the challenges of energy efficiency and affordability. A well-planned deployment, targeted marketing, and an effective commercialization strategy will ensure the success and sustainability of the project in the market.

4. Description of Personal and Facilities

Member	Task	Focus
Pivithuru N.H.A.S.	Web Application Development (Laravel + React) The web application component of the Home Energy Management System (HEMS) will be responsible for providing an interactive and user-friendly platform to monitor energy consumption, display predictions and suggestions, and manage IoT unit functionalities. The web app will be developed using React.js for the frontend and Laravel for the backend, with cloud integration via AWS.	Analyze system requirements, identify necessary technologies, and collaborate with IoT and ML/AI teams to ensure smooth integration. Develop the frontend of the web application using React.js
		Set up the project structure, design responsive UI components for dashboards and user profiles, implement forms for unit registration and authentication, and ensure real-time data updates.
		Configure the database, develop RESTful APIs for data exchange, implement authentication and authorization, and integrate AWS cloud storage for data processing and retrieval.
		Establish communication, enable real-time data visualization, implement the unit restoration feature after exceeding limits, and provide remote control options.
		Fetch time-series energy predictions and AI-based suggestions via APIs, optimize data processing, and provide actionable insights for users within their unit profiles.
		Allow users to create, update, and manage accounts, implement security measures, and provide customized energy consumption tracking for different user roles.
		Set up backend hosting on AWS EC2, deploy the frontend using AWS Amplify/S3, and configure CI/CD pipelines for automated deployments and updates.
		Improve API response times, secure data transmission using encryption, conduct vulnerability

		assessments, and ensure compliance with industry security standards.
		Perform integration testing to verify communication between components, conduct user acceptance testing, and address identified bugs or performance issues.
		Create user manuals and technical documentation, conduct training sessions for end-users, and provide support during the transition to the maintenance phase.

Table 3:Description of personal and facilities

5. Budget and budget justification (if any)

Resource	Cost
Hardware:	
Microcontroller	1800 LKR
Current Sensor (ACS712)	800 LKR
Voltage Sensor (ZMPT101B)	550 LKR
Relay Module (5V Single Channel Relay)	300 LKR
LEDs and Buzzer (Standard LED + Buzzer)	250 LKR
DC Power Adapter (5V/3.3V DC Adapter)	1700 LKR
Cloud Services:	
IoT Core	\$5–10/month
DynamoDB	\$5–10/month
Lambda	\$2–5/month
S3	\$3–5/month
CloudWatch	\$1–3/month
API Gateway	\$2–5/month
Others:	
Internet	7000 LKR
Travelling	8000 LKR
Other	5000 LKR

Table 4:Table of Budget

6. References

[1] J. G. Cristina STOLOJESCU-CRISAN, "A Home Energy Management System," in *IEEE*, 2022.

[2] K. W. P. N. A. O. Jirawadee Polprasert, "Home Energy Management System and Optimizing," in *IEEE*, 2021.

[3] B. B. Prathyusha M R, "Development of IoT-based Smart Home," in *IEEE*, 2023.