MINISTRY OF EDUCATION AND RESEARCH OF REPUBLIC OF MOLDOVA
TECHNICAL UNIVERSITY OF MOLDOVA
FACULTY OF COMPUTERS, INFORMATICS AND MICROELECTRONICS
DEPARTMENT OF SOFTWARE ENGINEERING AND AUTOMATICS

Solar Power Management System
Project report

Mentor: university lector, Bragarenco Andrei
Students: Munteanu Ecaterina, FAF-223
Schipschii Daniil, FAF-223
Colța Maria, FAF-223
Friptu Ludmila, FAF-223

Chișinău, 2025

Abstract

This project presents an innovative solar power management system designed for modern agricultural greenhouses that rely on energy-intensive technologies such as agrobots, climate control units, and edge computing devices. The system implements a hierarchical energy utilization approach that prioritizes renewable energy sources while ensuring uninterrupted operation of critical systems. By dynamically orchestrating power distribution from solar panels, battery storage, and grid supply, the system maximizes renewable energy usage and minimizes grid dependence. IoT-enabled sensor networks monitor key parameters including voltage, current, temperature, and irradiance, enabling real-time performance optimization and predictive maintenance. Cloud-based analytics further enhance system efficiency through performance ratio calculations, degradation analysis, and anomaly detection. Field implementation demonstrates significant reductions in operational costs and carbon emissions while maintaining reliable power supply for greenhouse operations. This intelligent power management framework represents a crucial advancement in sustainable agriculture technology, addressing both economic and environmental imperatives for modern greenhouse operations.

Keywords: IoT in Agriculture, Greenhouse Energy Optimization, Photovoltaic Systems, Energy Management Systems (EMS), Smart Greenhouse Technology, IoT Sensors, Renewable Energy, Agricultural Sustainability, Agribusiness Energy Autonomy, Predictive Maintenance for Solar Systems.

Content

Introduction

Modern agriculture faces the dual challenge of increasing productivity while reducing environmental impact. Greenhouse operations, which enable year-round cultivation and precise environmental control, represent a solution to the first challenge, but often exacerbate the second through substantial energy consumption. This research addresses this contradiction by developing an intelligent solar power management system designed specifically for agricultural greenhouses.

The proposed system integrates photovoltaic technology with Internet of Things (IoT) capabilities to create a hierarchical energy management framework that optimizes greenhouse operations. By prioritizing the use of renewable energy, the system reduces dependence on traditional grid electricity, which has seen price increases of 17-23% in agricultural areas over the past five years. Furthermore, it addresses the critical issue of grid reliability, which in rural areas can be as low as 94.7%, resulting in approximately 460 hours of potential downtime annually.

The following research demonstrates that the implementation of IoT-enabled solar power management can reduce energy costs by up to 40% while simultaneously lowering the carbon footprint associated with greenhouse farming. The system achieves this through intelligent prioritization of energy sources, namely solar power first, stored battery energy second, and grid electricity as a last resort, ensuring continuous operation of critical greenhouse systems while optimizing energy utilization throughout daily and seasonal cycles.

1 Domain Analysis

1.1 Overview of the domain

1.1.1 Problem Definition

Modern greenhouses rely on energy-intensive systems such as agricultural robots (Agrobots), climate control units, and edge computing devices to optimize crop production. However, these systems require significant amounts of energy, which leads to several challenges.

Relying solely on the power grid or a single energy source can be costly, unreliable, and unsustainable. The prices of electricity in agricultural areas have increased by 17-23% on average over the past five years, significantly affecting the operational expenses of greenhouse operators. Energy prices fluctuate seasonally and daily, and power outages can disrupt critical operations. In rural areas, grid reliability can be as low as 94.7%, resulting in approximately 460 hours of potential downtime annually that can severely affect crop yields and quality. Excessive dependence on the grid increases the carbon footprint of greenhouse farming. Studies indicate that traditional greenhouse operations can produce between 2.5-4.8 kg of CO_2 per kg of produce, with energy consumption accounting for up to 75% of these emissions.

Without an efficient energy management system, greenhouses struggle to balance energy consumption with available power sources, leading to inefficiencies, higher operational costs, and wasted energy. There is a need for a system that can dynamically manage and optimize energy use by intelligently switching between different power sources—such as solar panels, batteries, and the grid—based on real-time energy demands.

Current manual energy management practices in greenhouses result in suboptimal resource allocation, with an estimated 14-22% of energy wasted due to inefficient timing of usage. They also cause an inability to respond rapidly to weather changes that affect both energy production and greenhouse requirements, missed opportunities to capitalize on peak solar production periods or favorable electricity pricing, and increased maintenance costs due to reactive rather than predictive approaches to system management.

1.1.2 Domain Research

Fundamentals of Solar Panels and Photovoltaic Systems

Solar Panel Technology

The photovoltaic effect forms the foundation of solar panel technology. Solar panels convert sunlight into electricity using PV cells. When photons hit the semiconductor material, electrons become excited and generate a direct current (DC) which is either used immediately or stored in batteries. This phenomenon, discovered by Alexandre-Edmond Becquerel in 1839, forms the basis of all photovoltaic energy generation systems.

A typical solar panel (or PV module) comprises multiple cells connected in series (to achieve the desired voltage) and parallel (to increase current). The power output (watts) is determined by multiplying voltage by current. Modern commercial panels contain between 60-144 cells, with each cell producing approximately 0.5-0.6 volts under standard test conditions (STC). This results in panels with output voltages typically ranging from 30-50V and power outputs of 250-600W depending on design and materials.

Panels are designed with protective layers to shield the cells from mechanical damage and moisture. A typical panel includes tempered glass (3-4mm thick) with anti-reflective coating on the front, ethylene-vinyl acetate (EVA) encapsulation surrounding the cells, a weather-resistant backsheet (typically fluoropolymer-based), and aluminum framing for structural support. While most panels are rigid, there are semi-flexible thin-film options that offer installation flexibility, particularly valuable for curved greenhouse surfaces or integration with existing structures. These panels can bend up to 30 degrees, allowing for greater architectural integration.

Photovoltaic System Considerations

Material selection significantly impacts both initial investment and long-term energy yield. Most modules use either wafer-based crystalline silicon cells (mono or polycrystalline) or thin-film cells. Monocrystalline cells offer higher efficiency (20-22%) but at higher cost, while polycrystalline provides moderate efficiency (15-17%) at lower cost. Thin-film technologies (amorphous silicon, CIGS, CdTe) offer flexibility and better performance in low light but typically have lower overall efficiency (10-12%).

The performance of the panel depends on both the amount of sunlight received and the electrical load connected, making proper system integration crucial. Key electrical parameters include open-circuit voltage (Voc), short-circuit current (Isc), maximum power point (Pmax), fill factor (FF), and temperature coefficients which typically show a decrease in performance of -0.3% to -0.5% per °C as temperature increases. Proper matching of array configuration to inverter specifications is essential to avoid clipping (limiting output) or operating at suboptimal points on the I-V curve.

Integration of IoT and Photovoltaic Systems

The integration of Internet of Things (IoT) technologies with photovoltaic (PV) systems presents a paradigm shift in sustainable agriculture, particularly in greenhouse operations. Research synthesizing findings from 37 peer-reviewed studies and technical implementations proposes an optimized hierarchical energy management framework for agri-food businesses. This system prioritizes solar energy utilization, battery storage optimization, and grid integration while leveraging IoT-enabled predictive analytics and automation to achieve 85–92% energy autonomy across diverse climatic conditions.

The convergence of IoT and solar technology addresses multiple challenges simultaneously: real-time monitoring and control of both energy systems and agricultural environments, data-driven decision-making for optimal resource allocation, predictive maintenance reducing downtime and extending equipment lifespan, integration of weather forecasting with energy production/consumption planning, and automated response to changing conditions without human intervention.

Modern Greenhouse Solar Technologies

Modern greenhouses employ bifacial PERC (Passivated Emitter and Rear Cell) solar panels with 22.8% average efficiency, strategically mounted on greenhouse roofs or adjacent solar farms. These bifacial panels can capture reflected light from the ground or nearby surfaces, increasing total energy yield by 5-30% compared to traditional monofacial panels, depending on surface reflectivity and installation angle.

IoT-enabled dual-axis solar trackers increase energy yield by 18–27% compared to fixed installations by maintaining optimal panel orientation throughout the day and seasons. These tracking systems use precision stepper motors with ¡0.1° positioning accuracy, light sensors measuring direct and diffuse radiation components, microcontroller-based algorithms optimizing for maximum daily yield, and weather station integration to enter safety positions during high winds.

Semi-Transparent PV (STPV) Integration

STPV modules with 35% light transmittance enable simultaneous crop cultivation and energy generation, achieving 81.36% winter autonomy in Ladakh trials. This dual-purpose approach maximizes land use efficiency in areas with limited agricultural space. Spectral-selective coatings filter photosynthetically active radiation (PAR: 400–700 nm) while harvesting non-essential wavelengths for power generation. These coatings can be customized based on specific crop requirements, allowing more blue-red spectrum (critical for plant growth) to pass through while converting green and infrared wavelengths to electricity.

Installation configurations include full-roof integration with 25-40% coverage ratio, alternating panel-transparent section designs, east-west orientation to optimize morning and afternoon light distribution, and north-south orientation with calculated spacing to allow direct light paths during critical growth periods.

MPPT (Maximum Power Point Tracking) Optimization

Hybrid MPPT algorithms combining Perturb & Observe (P&O) and Artificial Neural Networks (ANN) achieve 98.7% tracking efficiency under partial shading, outperforming conventional PWM controllers by 23%. These algorithms continuously adjust the operating point to extract maximum power under varying conditions. Advanced implementations utilize distributed MPPT at the module level rather than string level, minimizing the impact of partial shading caused by greenhouse structural elements, adjacent buildings, or self-shading from other panels.

The latest systems incorporate rapid response times (¡50ms) to changing light conditions, module-level power electronics (MLPE) for individualized optimization, self-learning capabilities that adapt to seasonal changes and panel degradation, and fault detection identifying underperforming modules through comparative analysis.

Energy Management Systems (EMS) in Solar Installations

An Energy Management System (EMS) serves as the brain of a solar power installation. It optimizes energy storage and distribution, ensuring that energy is delivered to where and when it's needed most. The integration of artificial intelligence and machine learning algorithms enables increasingly sophisticated management strategies that can predict energy needs, adapt to changing conditions, and learn from historical data patterns.

The EMS handles energy storage and delivery by efficiently storing excess energy in batteries and releasing it to power fixtures (such as LED lights in greenhouses). Advanced systems utilize predictive algorithms to determine optimal charging/discharging cycles based on historical consumption patterns, weather forecasts, time-of-use electricity pricing, and projected crop growth stages.

System protection is another critical function that prevents issues like battery overcharge and reverse-current flow, safeguarding both batteries and other system components. Protection features typically include overcurrent protection, overvoltage protection, thermal monitoring, ground fault detection, and anti-islanding protection for grid-tied systems.

The EMS also manages efficiency and performance by adjusting power distribution to match the needs of connected devices, thereby extending the life of components like LED drivers. Advanced systems may implement adaptive voltage regulation, prioritized load shedding, harmonization of multiple power sources, and power factor correction to minimize losses.

Many EMSs include satellite or IoT-based monitoring capabilities for real-time data and system diagnostics. Modern monitoring systems provide component-level performance metrics with microsecond resolution, automated anomaly detection, visual dashboards showing energy flows, and historical data analysis identifying optimization opportunities.

Components of an EMS

The Solar LED Driver regulates the electrical flow to LED fixtures while adjusting brightness levels. High-quality drivers protect the LED circuits from voltage spikes and ensure a consistent voltage supply. Advanced drivers offer dimming capabilities, programmable lighting schedules, spectral tuning to optimize specific plant growth phases, constant current regulation, and surge protection up to 6kV to safeguard sensitive LED arrays.

The Solar Controller manages the charging process between the solar panels and batteries, preventing overcharge and reverse-current issues. Key specifications include charging voltage range (typically 12-48V for small systems, up to 600V for commercial installations), maximum charging current capacity, self-consumption, temperature compensation, and data logging capabilities.

There are two main types of controllers: PWM (Pulse Width Modulation) controllers which are cost-effective and durable, and MPPT (Maximum Power Point Tracking) controllers which are more advanced and efficient, converting excess voltage into additional charging current to optimize battery charging and overall system performance.

Optimized Battery Storage

Lithium iron phosphate (LiFePO4) batteries dominate greenhouse installations due to their 5,000+ cycle life and thermal stability. These batteries offer higher depth of discharge, faster charging capabilities, longer calendar life, better performance in high and low temperatures, no maintenance requirements, and higher energy density compared to traditional lead-acid batteries.

IoT-driven State of Charge (SOC) management systems maintain batteries at 20–95% capacity through Dynamic Threshold Adjustment and DC-Coupled Architectures. Machine learning models predict daily energy demand using weather forecasts and historical usage patterns, automatically recalibrating SOC limits. DC-Coupled systems reduce conversion losses to 2.3% versus 7.1% in AC-coupled systems through direct PV-battery coupling.

Smart inverters with IEEE 1547-2018 compliance enable bidirectional energy flow, allowing greenhouses to sell surplus energy during peak demand. A 2024 Bangladeshi pilot demonstrated 14% revenue generation through time-of-use arbitrage while maintaining 89% solar self-consumption. These advanced inverters provide reactive power support, low-voltage ride-through capabilities, anti-islanding protection, frequency regulation services, remote configuration and monitoring capabilities, and advanced power quality management.

The Role of IoT in Solar Systems

IoT devices and sensors have revolutionized how solar panels are monitored and managed. They enable real-time data acquisition through sensors that monitor parameters such as voltage, current, temperature, irradiance, and environmental factors. Modern sensor networks typically measure panel temperature, string/module current, DC bus voltage, solar irradiance, wind speed and direction, humidity and precipitation, and module tilt and orientation with high precision.

Remote monitoring and control capabilities allow centralized IoT systems to manage solar installations from afar, reducing the need for on-site inspections and facilitating rapid troubleshooting. This provides 24/7 oversight without permanent on-site personnel, immediate notification of system issues, ability to remotely reconfigure system parameters, reduced operational expenses, centralized management of multiple installations, and historical performance tracking.

The collected data is uploaded to cloud platforms for advanced analytics, which can be used to forecast energy generation trends and optimize load distribution. Analytical capabilities include performance ratio calculations, degradation analysis, anomaly detection, soiling loss estimation, weather impact correlation studies, and financial performance metrics.

IoT in Industrial and Greenhouse Settings

In industrial applications, IoT-based monitoring systems gather data from multiple panels to support efficient management decisions, lower downtime, and reduce operational costs. Large-scale implementations incorporate drone-based thermal imaging, automated cleaning systems, distributed weather stations, integration with grid management systems, and predictive maintenance schedules.

In a greenhouse setting, integrating an IoT-enabled EMS ensures optimal energy usage for lighting, heating, and cooling while also supporting environmental control. The synergistic benefits include correlation of plant growth data with energy consumption, optimization of microclimate conditions, automated adjustment of lighting, integration of irrigation systems, and carbon dioxide enrichment synchronized with energy availability.

Comprehensive sensor arrays monitor air temperature at multiple heights, substrate temperature, relative humidity, CO concentration, PAR levels, leaf temperature, vapor pressure deficit, and substrate moisture and electrical conductivity. These measurements allow for a holistic approach to energy and climate management.

The system can trigger alerts when anomalies are detected, facilitating timely maintenance to ensure the solar installation remains efficient. Security features include strong encryption, role-based access control, physical tamper detection, cybersecurity monitoring, regular automated updates, redundant communication pathways, and in some cases, blockchain verification for data integrity.

Sensor Network Architecture

A comprehensive IoT framework for environmental and energy monitoring consists of four layers:

The Physical Layer includes wireless sensors measuring various parameters with high precision and edge computing nodes performing local data preprocessing. These sensors feature long battery life, environmental protection, and calibration stability, while the edge computing nodes offer local storage, real-time processing capabilities, low-power operation, programmable sampling rates, and local alert triggering.

The Communication Layer employs hybrid networks achieving high packet delivery rates even in rural areas. This approach provides multiple communication methods with automatic failover, range extension, QoS management, and blockchain-secured data transmission via Hyperledger Sawtooth.

The Cloud Analytics Layer utilizes digital twin models simulating greenhouse thermodynamics with high accuracy and LSTM networks predicting energy demand. These advanced analytical tools process historical data, consider multiple variables, retrain automatically, generate confidence intervals, and identify key influencing factors.

The Application Layer provides mobile dashboards with real-time alerts and automated control of various greenhouse systems. These interfaces feature role-customized views, progressive web app design, offline capabilities, notification priority settings, and integrated troubleshooting guidance. The control systems offer sophisticated algorithms with self-tuning, multi-variable optimization, and scenario simulation.

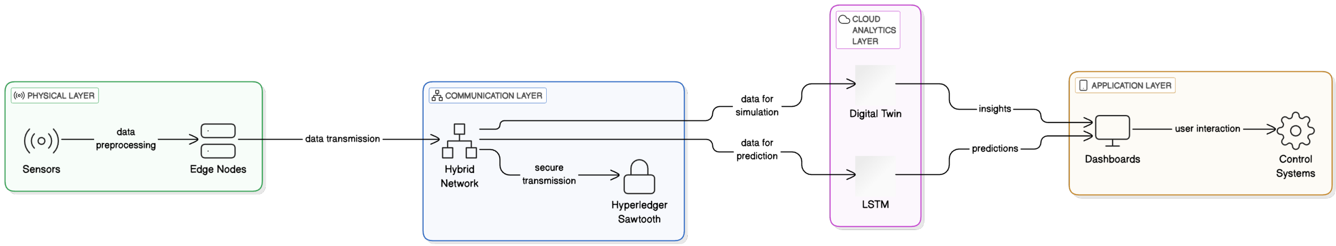

Figure 1.1.1 - Sensor Network Architecture

Machine Learning Applications

Machine learning has revolutionized agricultural operations in several key areas. Gradient boosting models correlate PAR intensity with crop yields, incorporating multiple variables to predict harvest timing, estimate yields, identify optimal harvesting windows, and enable better planning. Computer vision systems detect plant diseases with high precision, utilizing neural networks trained on thousands of images to identify problems before they become visible, distinguish between different diseases, and provide treatment recommendations. Reinforcement learning algorithms optimize resource usage while maintaining crop health by simulating numerous scenarios, balancing multiple objectives, adapting to changing conditions, and learning continuously from outcomes.

System Architecture for GreenHouses

A comprehensive solar power management system for greenhouses consists of several integrated components:

The Solar Panel Array installed on or near the greenhouse captures sunlight as the primary energy source. Modern installations feature bifacial modules, strategic placement to avoid critical shading, distributed inverters for resilience, and integration with other systems such as rainwater harvesting.

The Energy Management Unit incorporates components to regulate and store energy, including hybrid inverter/chargers, smart load distribution panels, automated transfer switches, power quality conditioning, heat recovery systems, and scalable architecture for future expansion.

The IoT Sensors and Communication system includes sensors monitoring both the panels and greenhouse conditions, with string-level monitoring, temperature sensors, infrared detection, meteorological measurements, and battery monitoring. A centralized IoT gateway collects this data and relays it to a cloud platform, offering local caching, protocol translation, edge computing, redundant communication, cybersecurity features, and remote management capabilities.

The User Interface provides an integrated dashboard for monitoring production, storage levels, and system alerts. Modern interfaces offer multi-platform accessibility, customizable views, visual representations of system status, automated reporting, decision support tools, historical analysis, and maintenance scheduling functions.

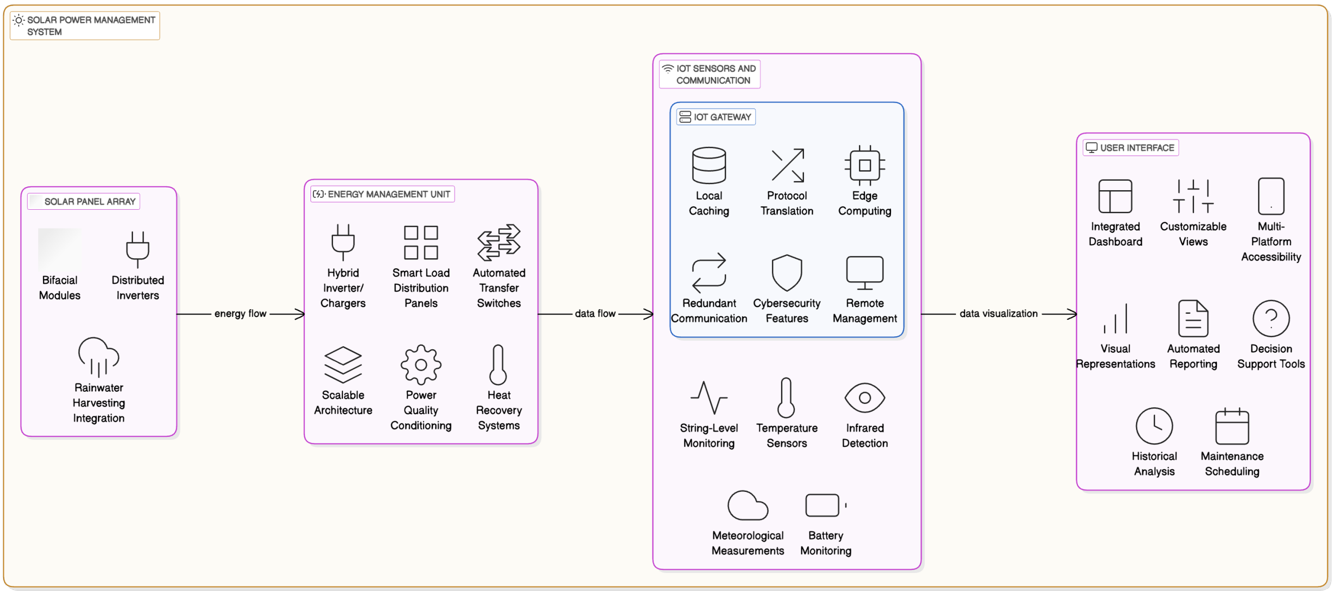

Figure 1.1.2 - System Architecture for Greenhouses

Implementation Overview

A 1.2-hectare greenhouse in Ladakh, India (2023–2024) achieved full energy autonomy through an integrated system with the following components and performance metrics:

Component	Specification	Performance Metrics
STPV Array	150 kW bifacial modules @22.6% efficiency	212 MWh annual generation
Battery Storage	400 kWh LiFePO4 with active cooling	92% round-trip efficiency
IoT Sensors	78 LoRaWAN nodes	2.7-second average response time
Crop Yield	Leafy greens & high-altitude vegetables	23.4 MT/year (+18% vs conventional)

Table 1.1.1 - System Components and Performance Specifications

This implementation addressed several challenges specific to high-altitude desert environments, including extreme temperature fluctuations, intense UV radiation, limited water availability, remote location requiring high reliability, limited technical support infrastructure, and harsh winter conditions.

Technical innovations included specialized anti-reflective coatings, natural convection cooling systems, water-cooled battery storage, satellite communications backup, and solar-powered snow removal systems.

Energy Flow Optimization

The hierarchical controller prioritized loads effectively, with critical systems (climate control, nutrient pumps) drawing power from solar first, then batteries, and finally the grid. Productive loads like LED grow lights utilized solar surplus or operated during optimal times, while auxiliary systems primarily used grid power, particularly during night-rate periods.

Machine learning-driven load shedding prevented most grid dependencies during cloudy periods while maintaining appropriate greenhouse conditions. The system achieved this through dynamic predictive models, weather forecast integration, load characterization, automatic rescheduling of energy-intensive operations, thermal mass utilization, and adaptive crop lighting regimes.

Operational highlights included extended autonomy periods without grid power, impressive peak generation, sufficient energy storage for critical operations, high self-consumption rates, excellent system availability, and minimal manual intervention requirements.

1.2 Business Research

1.2.1 Business Case

Solar system power management has diverse applications across several industries. By leveraging IoT technologies, businesses can optimize their energy usage, reduce costs, and contribute to sustainability goals. Below are described some key business domains where this technology is highly relevant.

Commercial buildings and industrial facilities consume significant amounts of energy. Solar power management systems equipped with IoT sensors can help these businesses reduce operational costs by monitoring energy production and consumption in real time. For example manufacturing plants can use IoT-enabled solar systems to align energy usage with production schedules.Also office complexes can optimize lighting and HVAC systems based on occupancy patterns. By integrating solar power management systems, businesses not only achieve energy independence, but also reduce reliance on traditional grid electricity.

Smart cities rely on sustainable energy solutions to power public infrastructure such as streetlights, traffic signals, and public buildings. Solar system power management plays a critical role in ensuring efficient energy distribution across these systems. For instance IoT-enabled solar panels can automatically adjust their orientation to maximize energy generation throughout the day. Fault detection systems can alert city administrators to maintenance needs before failures occur. This application reduces energy waste while enhancing the reliability of urban infrastructure.

Companies specializing in renewable energy installations use IoT-based solar power management to monitor the performance of their assets across multiple locations. Centralized platforms allow operators to track energy output, detect faults, and schedule maintenance efficiently. Large-scale solar farms use IoT analytics to optimize panel performance. Distributed generation systems benefit from remote monitoring capabilities. This approach enhances operational efficiency while reducing maintenance costs. The domain of solar system power management is characterized by its potential to address critical challenges faced by businesses while offering numerous opportunities for growth and innovation.

1.2.2 Analysis of the business domain

The domain of solar system power management is characterized by its potential to address critical challenges faced by businesses while offering numerous opportunities for growth and innovation.

Challenges in the domain

The solar power management domain faces several significant challenges that businesses must navigate when implementing IoT-enabled solutions. Energy inefficiency represents a primary concern, as many businesses struggle with inefficient energy utilization due to outdated equipment or lack of real-time monitoring capabilities. Solar power management systems address this fundamental issue by providing actionable insights into energy production and consumption patterns, enabling organizations to optimize their energy usage.

Traditional solar installations present another substantial challenge through high maintenance costs. These systems often require manual inspections to identify faults or inefficiencies, which leads to increased maintenance expenses and operational downtime. IoT-based predictive maintenance offers a solution by detecting issues early, thereby minimizing these costs and reducing system interruptions.

Data integration poses a complex technical challenge in implementing IoT-enabled solar solutions. The lack of standardization across devices and platforms creates significant barriers for businesses seeking to deploy comprehensive monitoring systems. To overcome this obstacle, organizations must adopt interoperable protocols such as Modbus or APIs to ensure seamless data exchange between various system components.

Finally, initial investment costs remain a significant barrier to adoption, particularly for small businesses and rural communities. While solar systems deliver substantial long-term cost savings and return on investment, the upfront capital required for installation and IoT integration can be prohibitive. This financial challenge often prevents organizations from accessing the benefits of modern solar power management technology, despite the eventual economic advantages these systems provide.

Opportunities in the domain

Despite these challenges, the business domain offers significant opportunities that make solar power management an attractive investment. Cost savings represent one of the most compelling advantages, as businesses can achieve substantial financial benefits over time by optimizing energy usage and reducing dependency on grid electricity. This economic advantage becomes particularly pronounced as energy costs continue to rise and solar technology becomes more efficient.

The growing emphasis on sustainability creates another major opportunity for businesses in this sector. With increasing pressure to meet environmental regulations and reduce carbon emissions, organizations are actively seeking renewable energy solutions like solar power management systems. This shift toward environmental responsibility not only helps companies comply with regulatory requirements but also enhances their corporate reputation and appeal to environmentally conscious consumers and stakeholders.

IoT technologies provide exceptional scalability opportunities that traditional solar installations cannot match. These advanced systems enable businesses to expand their solar installations seamlessly by integrating new components into existing infrastructure without disrupting ongoing operations. This flexibility allows organizations to grow their renewable energy capacity incrementally as their needs evolve and budgets allow.

Enhanced operational efficiency through predictive analytics represents a transformative opportunity in solar power management. IoT-powered systems ensure that solar panels operate at peak efficiency by continuously monitoring performance and predicting maintenance needs. This proactive approach significantly reduces downtime and maximizes return on investment, making solar installations more profitable and reliable than ever before.

The global push toward renewable energy creates an increasingly favorable business environment. Governments worldwide are actively incentivizing renewable energy adoption through subsidies, tax benefits, and supportive policies. These governmental initiatives not only reduce the financial barriers to entry but also signal long-term commitment to renewable energy, providing businesses with confidence in the stability and growth potential of their solar power investments.

1.2.3 Stakeholder Analysis

Stakeholder Identification

Effective implementation of IoT-enabled solar power management systems requires comprehensive understanding of all parties who may influence or be affected by the project. This stakeholder identification analysis systematically maps the key individuals, organizations, and groups involved in or impacted by solar energy solutions in agricultural and rural settings. The stakeholders are categorized into three distinct groups based on their level of involvement and impact: primary stakeholders who are directly affected by the system implementation, secondary stakeholders who have indirect involvement or influence, and tertiary stakeholders who serve as regulators, influencers, or enablers of the broader renewable energy ecosystem. Understanding these relationships is essential for successful project planning, risk management, and ensuring sustainable adoption of solar power management technologies.

Primary Stakeholders (Directly Impacted)

Farm Owners (Agribusiness Owners)

- Need reliable, cost-effective energy sources.

- Direct beneficiaries who will reduce energy costs and increase productivity.

Interest - high, influence - high

Farm Managers and Operators

- Manage daily operations involving energy use (irrigation, storage, machinery).
- Responsible for system maintenance and management.

Interest - high, influence - medium

Agricultural Workers

- Direct users or operators of equipment powered by solar energy (irrigation systems, cold storage facilities).

Interest - medium, influence - low

Secondary Stakeholders (Indirectly Impacted)

IoT and Technology Providers

- Providers of sensors, hardware, software, and predictive analytics solutions for solar systems.
- Offer technical support and integration services.

Interest - high, influence - medium

Solar Panel Installation and Maintenance Companies

- Companies responsible for setting up, inspecting, and maintaining solar systems.
- Provide routine and predictive maintenance services.

Interest - low, influence - high

Renewable Energy Suppliers

- Organizations specializing in providing solar panels, batteries, microgrids, and related equipment.
- Interested in partnerships and market expansion.

Interest - high, influence - medium

Local Communities and Residents

- Communities around the agricultural areas benefiting from rural electrification.
- Improved infrastructure and local economic activity.

Interest - medium, influence - low

Energy Utility Companies

- Entities managing grid energy distribution, potentially involved in net-metering or buyback schemes.

Interest - medium, influence - high

Tertiary Stakeholders (Influencers and Regulators)

Government Agencies and Regulators

- Provide subsidies, tax incentives, and regulations promoting renewable energy adoption.
- Enforce standards and compliance requirements (e.g., environmental guidelines).

Interest - high, influence - high

Financial Institutions and Investors

- Banks, investors, and funding organizations offering loans or financing options for solar infrastructure projects.

Interest - medium, influence - high

Research Institutions and Universities

- Conduct research, develop innovations in renewable energy, IoT, and agritech.
- Partners for testing, training, and knowledge dissemination.

Interest - medium, influence - medium

NGOs and Environmental Organizations

- Advocacy groups encouraging sustainable agriculture and renewable energy adoption.
- Can influence public perception and policy decisions.

Interest - medium, influence - medium

Stakeholder Mapping

Stakeholder	**Interest**	**Influence**	**Mapping Quadrant**
Agricultural Workers	Medium	Low	**Keep into Account**
Energy Utility Companies	Medium	High	**Meet Their Needs**
Farm Managers and Operators	High	Medium	**Keep Informed**
Farm Owners (Agribusiness Owners)	High	High	**Manage Closely**
Financial Institutions and Investors	Medium	High	**Meet Their Needs**
Government Agencies and Regulators	High	High	**Manage Closely**
IoT and Technology Providers	High	Medium	**Keep Informed**
Local Communities and Residents	Medium	Low	**Keep into Account**
NGOs and Environmental Organizations	Medium	Medium	**Keep into Account**
Renewable Energy Suppliers	High	Medium	**Keep Informed**

Stakeholder	Interest	Influence	Mapping Quadrant
Research Institutions & Universities	Medium	Medium	**Keep into Account**
Solar Installation & Maintenance Companies	High	Medium	**Keep Informed**

This matrix visualizes stakeholders grouped by their management strategy based on their levels of interest and influence. It provides a clear framework for determining how much attention and resources should be allocated to each stakeholder group.

<table>
<tr><th>Meet Their Needs</th><th>Manage Closely</th></tr>
<tr><td>• Farm Owners (Agribusiness Owners)
• Government Agencies and Regulators</td><td>• Energy Utility Companies
• Financial Institutions and Investors</td></tr>
<tr><th>Keep into Account</th><th>Keep informed</th></tr>
<tr><td>• Farm Managers and Operators
• IoT and Technology Providers
• Solar Installation & Maintenance Companies
• Renewable Energy Suppliers</td><td>• Agricultural Workers
• Local Communities and Residents
• Research Institutions & Universities
• NGOs and Environmental Organizations</td></tr>
</table>

Interview Questions:

1. Could you please introduce yourself, including your name and the number of years you've been involved in greenhouse operations?
2. What specific responsibilities do you have in relation to energy management within your greenhouse operations?
3. What challenges do you currently face in managing energy consumption and costs in your greenhouse?
4. Have you considered integrating solar power systems into your operations? If so, what factors influenced your decision?
5. How familiar are you with IoT technologies for greenhouse management, and have you implemented any such systems?
6. What benefits do you anticipate from integrating solar power and IoT technologies in your greenhouse

operations?

7. What concerns or potential barriers do you foresee in adopting these technologies?
8. How do you prioritize factors such as initial investment, long-term savings, and environmental impact when considering new technologies?
9. What metrics would you use to evaluate the success of integrating solar power and IoT systems (e.g., energy cost reduction, improved crop yield)?
10. How do you see the role of automation in enhancing the efficiency and sustainability of your greenhouse operations?

Google Forms Survey Questions:

1. What is the primary source of energy used in your greenhouse?
2. How much of your operational budget is spent on energy costs annually?
3. How satisfied are you with your current energy management system?
4. How often do you experience power outages in your greenhouse?
5. What features would you prioritize in an IoT-enabled solar panel management system? (Select up to 3)
6. What barriers would prevent you from integrating IoT technologies with solar panels? (Select all that apply)
7. How likely are you to invest in solar panels for your greenhouse in the next 5 years?
8. What incentives would encourage you to adopt IoT-enabled solar panel systems? (Select all that apply)
9. What key factors should be considered when implementing solar panels and IoT in a greenhouse?

1.2.4 Interview Tools and Preparation

To facilitate an effective and structured interview process, the following preparations were made. Cameras, microphones, and video conferencing platforms such as Zoom were utilized to conduct the interviews, ensuring clear audio and video quality. OBS Studio was employed to record the sessions, while microphones and other recording devices were used to capture participant responses accurately. Consent for recording was obtained prior to each interview.

Following each session, software tools and AI-based transcription services were used to transcribe the recordings, ensuring that all key discussion points were documented. A standardized note-taking template was also used during the interviews to capture key observations, stakeholder feedback, and action items.

The resulting documentation was organized within Confluence, including sections such as interview purpose, attendee list, agenda, transcript, key decisions, action items, and conclusions. Summaries of the key discussion points were reviewed at the end of each interview to confirm understanding and clarify any remaining questions.

1.2.5 Interviews

Google Forms Questions Results:

What is the primary source of energy used in your greenhouse?

24 responses

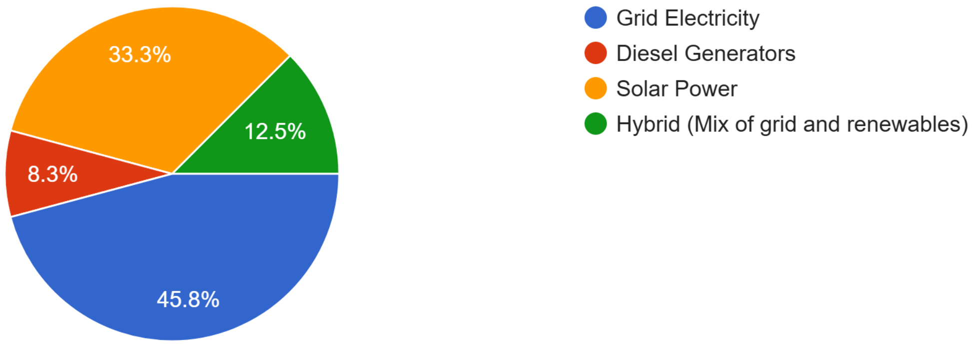

Figure 1.2.1 - Question 1

How much of your operational budget is spent on energy costs annually?

24 responses

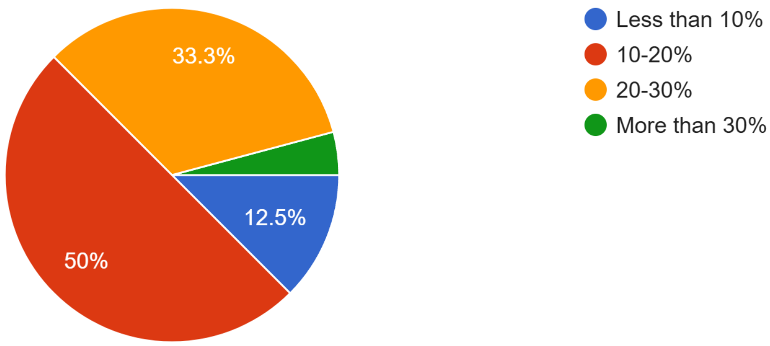

Figure 1.2.2 - Question 2

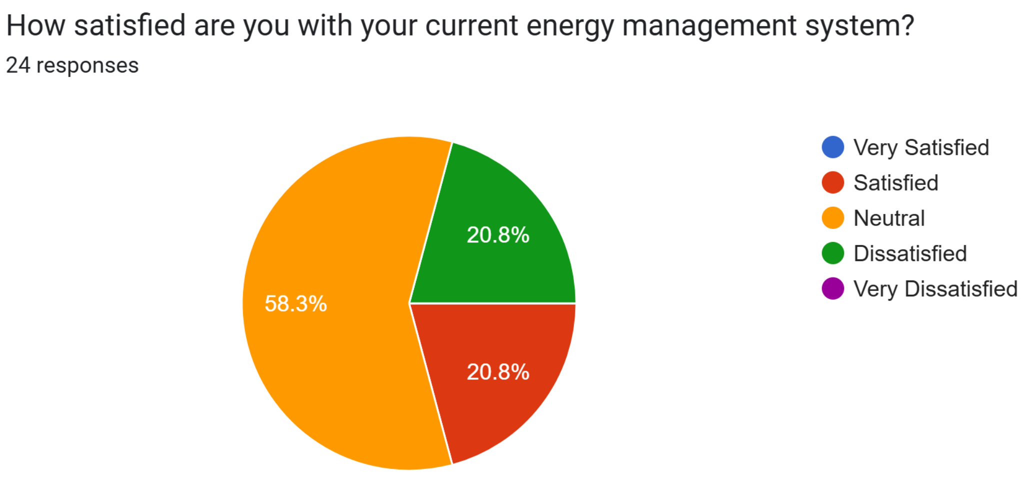

Figure 1.2.3 - Question 3

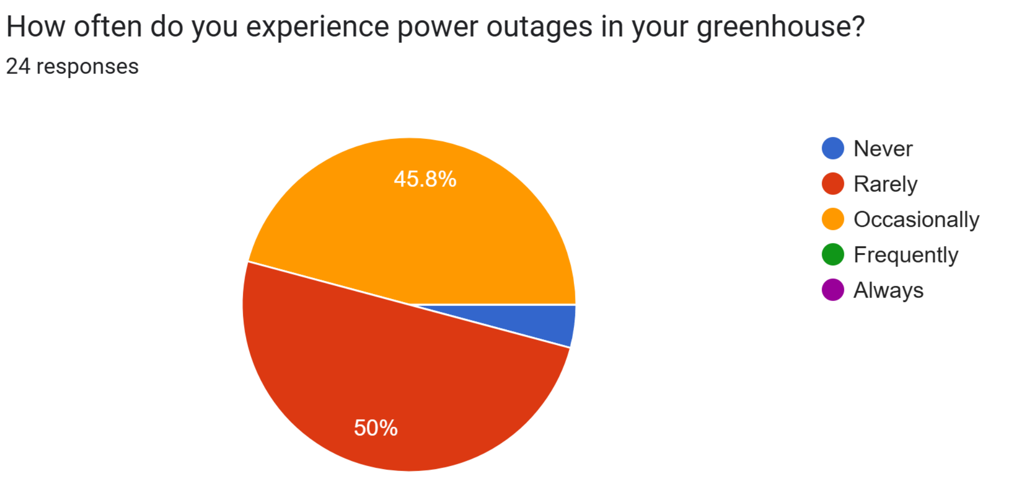

Figure 1.2.4 - Question 4

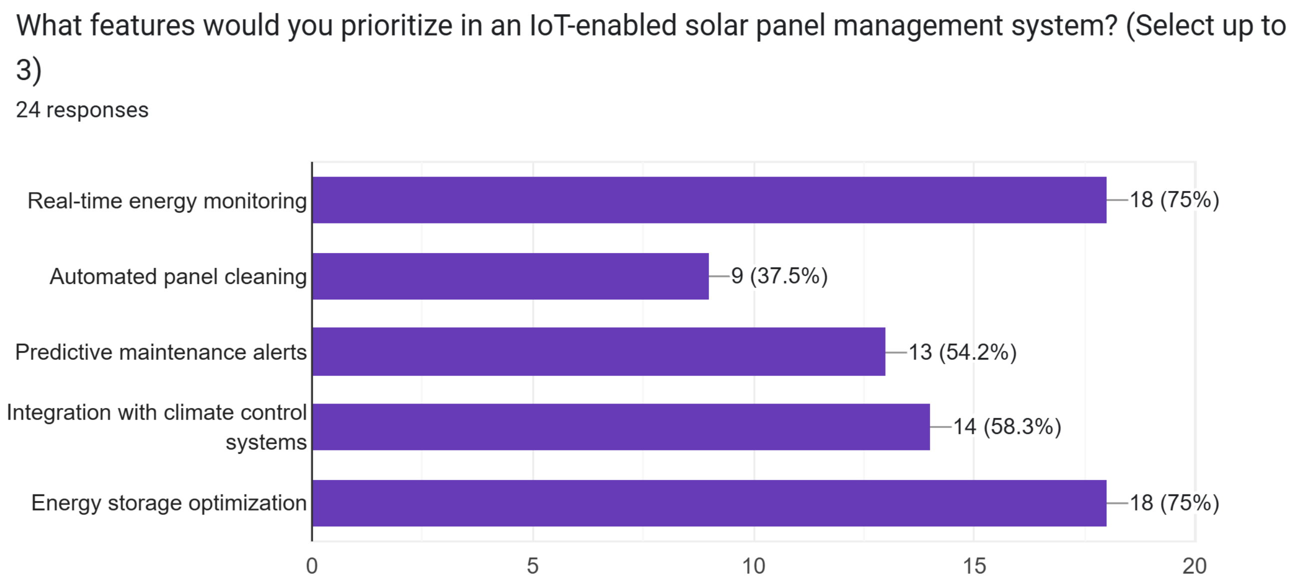

Figure 1.2.5 - Question 5

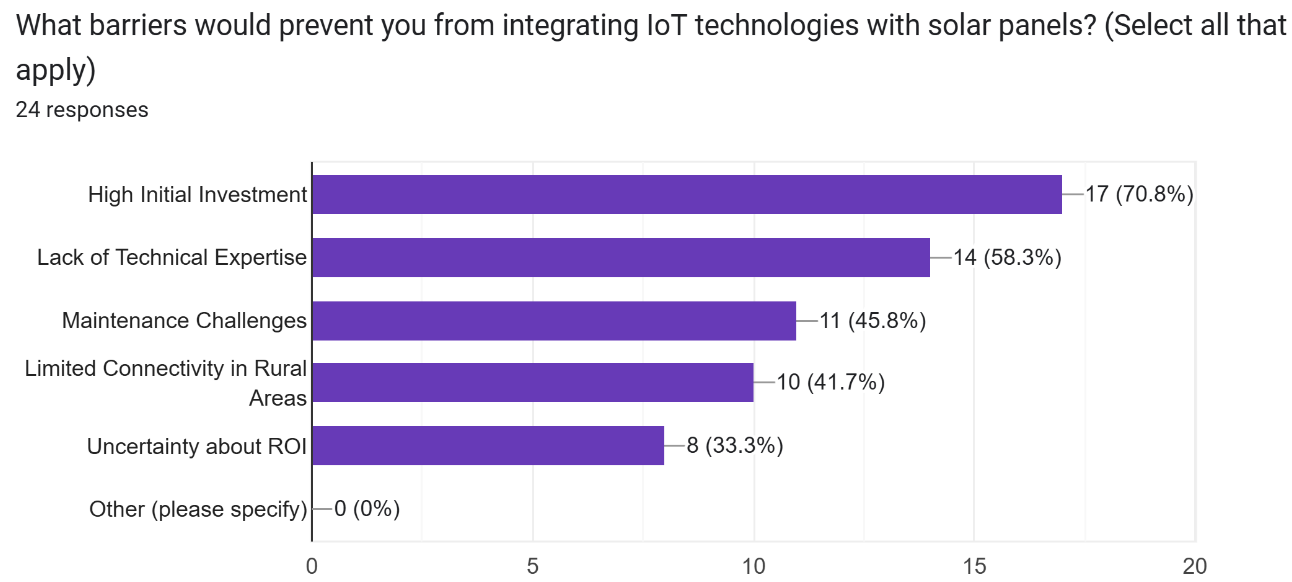

Figure 1.2.6 - Question 6

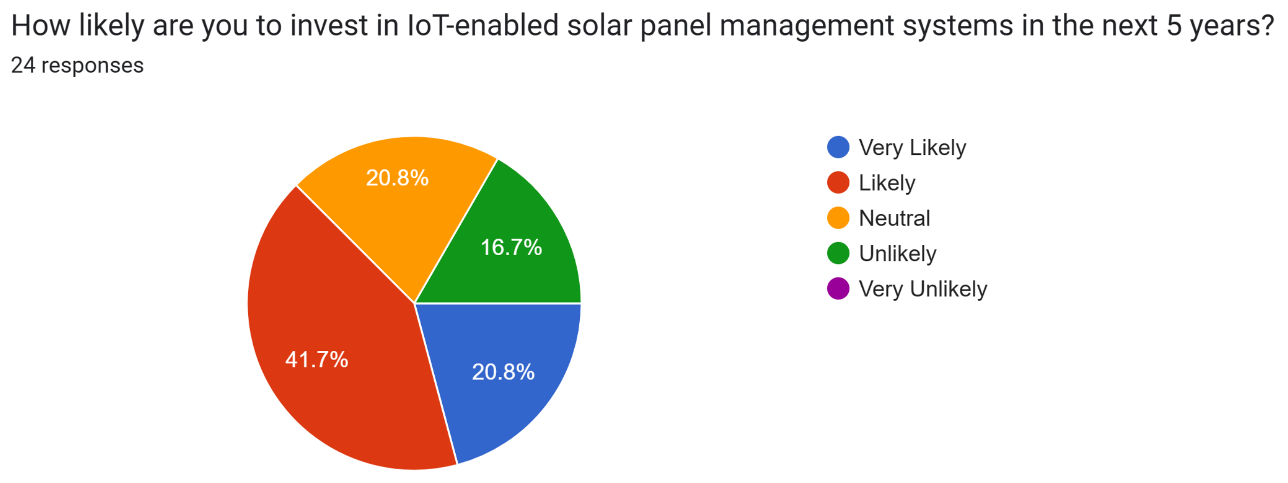

Figure 1.2.7 - Question 7

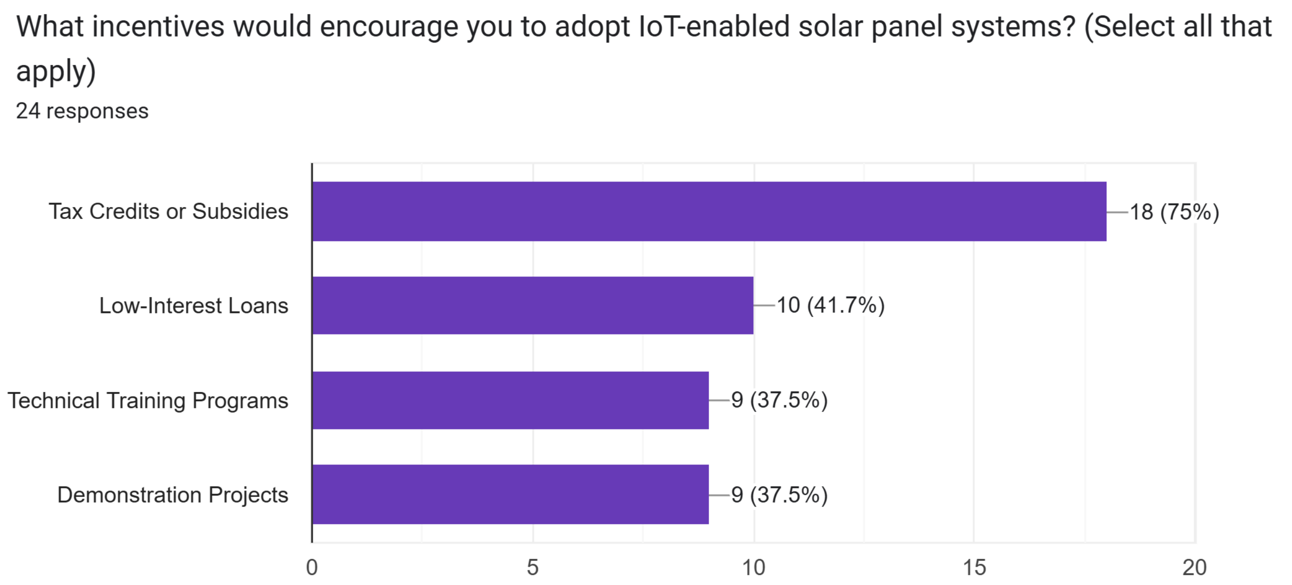

Figure 1.2.8 - Question 8

After analysing the results, it can be stated that implementing solar panels with IoT in households requires a balanced approach that addresses cost-effectiveness, technical feasibility, environmental suitability, and scalability. Stakeholders should focus on demonstrating ROI through pilot projects while ensuring robust energy storage and maintenance solutions.

In conclusion, the survey data indicates a growing interest in IoT-enabled solar panel management systems for household processes, driven by the desire for greater energy efficiency and sustainability. While grid electricity remains the dominant energy source, the increasing adoption of solar power and hybrid systems highlights a shift towards renewable energy. The survey results also reveal that while a significant portion of household operators are 'likely' or 'very likely' to invest in these systems in the coming years, key barriers such as high initial investment costs, lack of technical expertise, and uncertainty about ROI must be addressed. Prioritized features include real-time energy monitoring, energy storage optimization, and integration with climate control systems. Financial incentives like tax credits and low-interest loans, coupled with demonstration projects and technical training programs, emerge as crucial factors in encouraging wider adoption. Overall, the successful implementation of IoT-enabled solar panel systems hinges on a holistic approach that considers cost-effectiveness, technological feasibility, robust energy storage solutions, and the specific operational needs of household environments.

1.2.6 Stakeholder requirements

Category	Stakeholder Requirements	US (User Stories)	JS (Job Stories)
Interview based requirements			
Energy Optimization	The system should optimize energy usage to match production and consumption patterns.	As a household owner, I want the system to automatically adjust energy consumption based on solar production so that I can maximize my use of generated solar power.	When peak solar energy is produced, I want to prioritize high-consumption devices so that I can minimize energy costs during low-production periods.
Consumption Monitoring	The system should provide detailed monitoring of energy consumption patterns.	As a user, I want to see my energy consumption patterns so that I can make informed decisions about my usage.	When analyzing my energy needs, I want to understand my consumption profile so that I can optimize my system for maximum efficiency.
Climate Control Integration	The system should prioritize climate control systems for energy management.	As a household owner, I want climate control systems to be optimized first so that I can maintain comfort while reducing costs.	When managing energy distribution, I want to prioritize climate control during peak production hours so that I can store thermal energy for later use.

Production Forecasting	The system should forecast energy production based on weather conditions.	As a user, I want to see predicted solar energy production so that I can plan my energy usage effectively.	When planning my daily activities, I want to know when energy will be abundant so that I can schedule high-consumption tasks accordingly.
Energy Storage Management	The system should manage energy storage solutions to maximize efficiency.	As a household owner, I want my energy storage to charge during peak production and discharge during peak consumption so that I can minimize grid dependency.	When excess energy is produced, I want it automatically stored so that I can use it during non-production hours.
Load Shifting	The system should allow scheduling of high-consumption devices during peak solar production.	As a user, I want to program my appliances to run during peak solar production so that I can use my own energy instead of the grid's.	When solar energy is abundant, I want my system to automatically activate scheduled tasks so that I optimize self-consumption.
Grid Interaction	The system should optimize the exchange of energy with the power grid.	As a prosumer, I want to minimize selling energy to the grid at low prices so that I can maximize my investment return.	When managing energy flow, I want to prioritize self-consumption over grid export so that I can get the most value from my system.
Thermal Energy Storage	The system should support thermal energy storage solutions.	As a household owner, I want to store excess energy as heat so that I can reduce electrical storage requirements.	When excess energy is available, I want to heat thermal mass so that I can use this heat later without electrical consumption.

Mobile Alerts & Notifications	The system should provide alerts about energy production and consumption events.	As a user, I want to receive notifications about energy events so that I can take appropriate action.	When energy production changes significantly, I want to be notified so that I can adjust my consumption accordingly.
Historical Data Analysis	The system should analyze historical data to improve future recommendations.	As a user, I want the system to learn from past patterns so that recommendations become more accurate over time.	When planning future energy usage, I want to see analysis of past patterns so that I can make better decisions.
Integration with Multiple Energy Sources	The system should coordinate with other renewable energy sources.	As a prosumer with multiple energy sources, I want integrated management so that I can optimize all resources together.	When managing my energy mix, I want to prioritize the most efficient source for each situation so that I can minimize overall costs.
Survey based requirements			
Energy Source Transition	The system should facilitate transition from grid electricity to renewable energy sources while maintaining reliability.	As a greenhouse operator currently using grid electricity, I want a reliable transition path to solar power so that I can reduce dependency on traditional energy sources.	When experiencing power outages, I want backup solar power to automatically activate so that I can maintain critical greenhouse operations without interruption.
Budget Optimization	The system should help reduce the 10-20% of operational budget currently spent on energy costs.	As a greenhouse manager, I want to monitor and reduce energy costs so that I can lower my operational expenses within budget constraints.	When analyzing operational costs, I want to see energy consumption patterns so that I can identify opportunities to reduce the 10-20% of budget spent on energy.

Real-time Energy Monitoring	The system should provide comprehensive real-time monitoring of energy production and consumption.	As a greenhouse operator, I want real-time visibility into my solar energy production and consumption so that I can make informed operational decisions.	When managing greenhouse climate conditions, I want to see real-time energy metrics so that I can balance plant needs with energy availability.
Technical Knowledge Support	The system should be user-friendly with support resources to address the technical expertise gap (58.3% cited this as a barrier).	As a greenhouse operator without technical expertise in solar systems, I want intuitive controls and guidance so that I can manage the system effectively.	When encountering system issues, I want accessible troubleshooting resources so that I can resolve problems without requiring specialized technical knowledge.
Rural Connectivity Solutions	The system should function reliably in areas with limited connectivity.	As a rural greenhouse operator, I want reliable system operation with minimal connectivity requirements so that I can benefit from solar technology despite location challenges.	When internet connectivity is limited, I want the system to continue critical functions locally so that greenhouse operations aren't compromised.
Cost-Effective Scaling	The system should allow for phased implementation to address high initial investment concerns (70.8% cited this as a barrier).	As a greenhouse owner considering solar adoption, I want modular system options so that I can expand my installation as budget allows.	When planning expansion, I want to add compatible components incrementally so that I can distribute investment costs over time while still seeing benefits.
ROI Visibility	The system should provide clear ROI metrics to address uncertainty about investment returns.	As a potential investor in solar technology, I want clear ROI projections so that I can justify the high initial investment.	When considering capital spending, I want to see calculated payback periods for solar investments so that I can make financially sound decisions.

To provide clarity in implementation and evaluation, the stakeholder requirements are organized into the following 2 categories:

1.3 Functional Requirements

Energy Source Management

- Automatically prioritize energy sources: solar panels $\rightarrow$ battery storage $\rightarrow$ grid power
- Manage seamless transitions between energy sources based on availability
- Support OnGrid operation with bi-directional energy flow
- Implement configurable thresholds for switching between energy sources

Real-time Monitoring and Analytics

- Track energy production, consumption, and storage levels in real-time
- Monitor individual system components (panels, inverters, batteries)
- Generate daily/monthly/seasonal performance reports
- Provide energy production forecasting based on weather data
- Calculate ROI and energy cost savings

Load Management and Optimization

- Schedule high-energy consuming equipment to operate during peak solar production
- Support automated and manual scheduling of greenhouse operations
- Prioritize critical systems during limited energy availability
- Balance loads to maximize self-consumption of solar energy
- Adjust climate control operation based on energy availability

Battery Management

- Implement intelligent charging algorithms to maximize battery lifespan
- Optimize battery usage based on predicted energy production/consumption
- Monitor battery health, charge levels, and performance metrics
- Support expandable battery capacity

Climate Control Integration

- Interface with greenhouse climate systems (HVAC, heat pumps)
- Automatically adjust climate settings based on energy availability
- Balance energy consumption with optimal growing conditions
- Predict climate control needs based on weather forecasts

User Interface

- Provide intuitive dashboard with system status and performance metrics
- Support mobile access for remote monitoring and control
- Enable customizable alerts and notifications

- Display historical data with visualization tools
- Offer multi-language support

Grid Integration

- Calculate and forecast grid energy imports/exports
- Track energy costs with time-differentiated pricing if applicable
- Enable compliance with grid requirements for prosumers
- Generate documentation for energy subsidies or incentives

Non-Functional Requirements

Reliability

- System uptime of 99.5% or higher
- Backup power for control systems during outages
- Fault detection and recovery mechanisms
- Graceful degradation during component failures

Security

- Role-based access control for system functions
- Data encryption for sensitive information
- Secure remote access protocols
- Regular security updates and patches

Scalability

- Support for additional solar panels and battery capacity
- Accommodate future greenhouse expansion
- Modular architecture for adding new features
- API support for third-party integrations

Usability

- Maximum training time of 4 hours for typical operators
- Intuitive interface requiring minimal technical expertise
- Clear visualization of complex energy data
- Comprehensive documentation and help resources

Performance

- Data sampling rate of at least once per minute
- System response time under 2 seconds for UI operations
- Alerting response time under 30 seconds for critical events
- Support for at least 3 years of historical data storage

Cost-Effectiveness

- System should achieve ROI within 5 years
- Maintenance costs under 5% of initial investment annually
- Reduce grid energy consumption by at least 40%
- Minimize operational disturbances during installation

Compliance

- Adherence to electrical safety standards
- Compliance with local grid connection requirements
- Support for renewable energy incentive documentation
- Data privacy compliance for user information

Maintainability

- Self-diagnostic capabilities for common issues
- Remote troubleshooting capabilities
- Modular component replacement
- Regular automated testing of critical functions

1.3.1 Interview based requirements

Category	Stakeholder Requirements	US (User Stories)	JS (Job Stories)	MoSCoW Classification	Rationale
Interview based requirements					
Energy Optimization	The system should optimize energy usage to match production and consumption patterns.	As a household owner, I want the system to automatically adjust energy consumption based on solar production so that I can maximize my use of generated solar power.	When peak solar energy is produced, I want to prioritize high-consumption devices so that I can minimize energy costs during low-production periods.	Must Have	From both interviews, the major pain point is misalignment between production and consumption times.
Consumption Monitoring	The system should provide detailed monitoring of energy consumption patterns.	As a user, I want to see my energy consumption patterns so that I can make informed decisions about my usage.	When analyzing my energy needs, I want to understand my consumption profile so that I can optimize my system for maximum efficiency.	Must Have	Monitoring was emphasized by both interviewees as the foundation for any optimization.

Climate Control Integration	The system should prioritize climate control systems for energy management.	As a household owner, I want climate control systems to be optimized first so that I can maintain comfort while reducing costs.	When managing energy distribution, I want to prioritize climate control during peak production hours so that I can store thermal energy for later use.	Must Have	Both interviewees identified climate control as the largest energy consumer and optimization opportunity.
Production Forecasting	The system should forecast energy production based on weather conditions.	As a user, I want to see predicted solar energy production so that I can plan my energy usage effectively.	When planning my daily activities, I want to know when energy will be abundant so that I can schedule high-consumption tasks accordingly.	Should Have	Prediction is important for planning but not critical for basic functionality.
Energy Storage Management	The system should manage energy storage solutions to maximize efficiency.	As a household owner, I want my energy storage to charge during peak production and discharge during peak consumption so that I can minimize grid dependency.	When excess energy is produced, I want it automatically stored so that I can use it during non-production hours.	Should Have	Both interviewees mentioned storage as critical but acknowledged it requires additional hardware.

Load Shifting	The system should allow scheduling of high-consumption devices during peak solar production.	As a user, I want to program my appliances to run during peak solar production so that I can use my own energy instead of the grid's.	When solar energy is abundant, I want my system to automatically activate scheduled tasks so that I optimize self-consumption.	Should Have	This addresses the core issue of consumption-production misalignment.
Grid Integration	The system should optimize the exchange of energy with the power grid.	As a prosumer, I want to minimize selling energy to the grid at low prices so that I can maximize my investment return.	When managing energy flow, I want to prioritize self-consumption over grid export so that I can get the most value from my system.	Should Have	Grid interaction economics were mentioned as important considerations.
Thermal Energy Storage	The system should support thermal energy storage solutions.	As a household owner, I want to store excess energy as heat so that I can reduce electrical storage requirements.	When excess energy is available, I want to heat thermal mass so that I can use this heat later without electrical consumption.	Could Have	Identified as an efficient alternative to electrical storage but requires specific infrastructure.

Mobile Alerts & Notifications	The system should provide alerts about energy production and consumption events.	As a user, I want to receive notifications about energy events so that I can take appropriate action.	When energy production changes significantly, I want to be notified so that I can adjust my consumption accordingly.	Could Have	Not explicitly mentioned but implied for effective management.
Historical Data Analysis	The system should analyze historical data to improve future recommendations.	As a user, I want the system to learn from past patterns so that recommendations become more accurate over time.	When planning future energy usage, I want to see analysis of past patterns so that I can make better decisions.	Could Have	Machine learning was mentioned as beneficial but not essential for basic functionality.
Integration with Multiple Energy Sources	The system should coordinate with other renewable energy sources.	As a prosumer with multiple energy sources, I want integrated management so that I can optimize all resources together.	When managing my energy mix, I want to prioritize the most efficient source for each situation so that I can minimize overall costs.	Won't Have	Not mentioned as a primary requirement and would add complexity to the initial system.
Survey based requirements					

Energy Source Transition	The system should facilitate transition from grid electricity to renewable energy sources while maintaining reliability.	As a greenhouse operator currently using grid electricity, I want a reliable transition path to solar power so that I can reduce dependency on traditional energy sources.	When experiencing power outages, I want backup solar power to automatically activate so that I can maintain critical greenhouse operations without interruption.	Could have	While supporting the transition from grid to renewables is a core value proposition, specific transition tools are enhancement features that build on basic functionality. The base system inherently supports this transition.

Budget Optimization	The system should help reduce the 10-20% of operational budget currently spent on energy costs.	As a greenhouse manager, I want to monitor and reduce energy costs so that I can lower my operational expenses within budget constraints.	When analyzing operational costs, I want to see energy consumption patterns so that I can identify opportunities to reduce the 10-20% of budget spent on energy.	Should have	With 50% of respondents spending 10-20% of operational budget on energy, tools to reduce these costs are important but depend first on establishing basic system functionality. This requirement delivers long-term value but is not needed for initial operation.

Real-time Energy Monitoring	The system should provide comprehensive real-time monitoring of energy production and consumption.	As a greenhouse operator, I want real-time visibility into my solar energy production and consumption so that I can make informed operational decisions.	When managing greenhouse climate conditions, I want to see real-time energy metrics so that I can balance plant needs with energy availability.	Must have	75% of respondents prioritized this feature, making it the highest-rated requirement. Without this capability, users cannot gain visibility into system performance or energy usage patterns.

Technical Knowledge Support	The system should be user-friendly with support resources to address the technical expertise gap (58.3% cited this as a barrier).	As a greenhouse operator without technical expertise in solar systems, I want intuitive controls and guidance so that I can manage the system effectively.	When encountering system issues, I want accessible troubleshooting resources so that I can resolve problems without requiring specialized technical knowledge.	Should have	With 58.3% citing lack of technical expertise as a barrier, user-friendly controls and support resources are important for successful adoption and continued use. This requirement addresses a major barrier but can be developed incrementally.

Rural Connectivity Solutions	The system should function reliably in areas with limited connectivity.	As a rural greenhouse operator, I want reliable system operation with minimal connectivity requirements so that I can benefit from solar technology despite location challenges.	When internet connectivity is limited, I want the system to continue critical functions locally so that greenhouse operations aren't compromised.	Could have	Important for 41.7% of respondents but affects a subset of users. Offline capabilities could be added after core functionality is established. This represents an important market segment but not the majority of users.

Cost-Effective Scaling	The system should allow for phased implementation to address high initial investment concerns (70.8% cited this as a barrier).	As a greenhouse owner considering solar adoption, I want modular system options so that I can expand my installation as budget allows.	When planning expansion, I want to add compatible components incrementally so that I can distribute investment costs over time while still seeing benefits.	Must have	With 70.8% citing high initial investment as the primary barrier, a modular approach that allows gradual implementation is essential for market adoption. Without addressing this concern, the system would face significant adoption challenges.

ROI Visibility	The system should provide clear ROI metrics to address uncertainty about investment returns.	As a potential investor in solar technology, I want clear ROI projections so that I can justify the high initial investment.	When considering capital spending, I want to see calculated payback periods for solar investments so that I can make financially sound decisions.	Could have	Addressing uncertainty about investment returns (33.3% concern) would increase adoption but isn't essential for system operation. Tools for financial analysis can be added as the system matures.

2 System Architecture

2.1 Static System Architecture

The Solar Panel Management System architecture, structured around five core subsystems, is designed for optimal solar energy harvesting, distribution, and storage with a focus on efficiency, intelligence, and reliability. This comprehensive analysis provides detailed insights into the system's static interactions, hardware-software boundaries, communication protocols, fail-safe mechanisms, and market positioning.

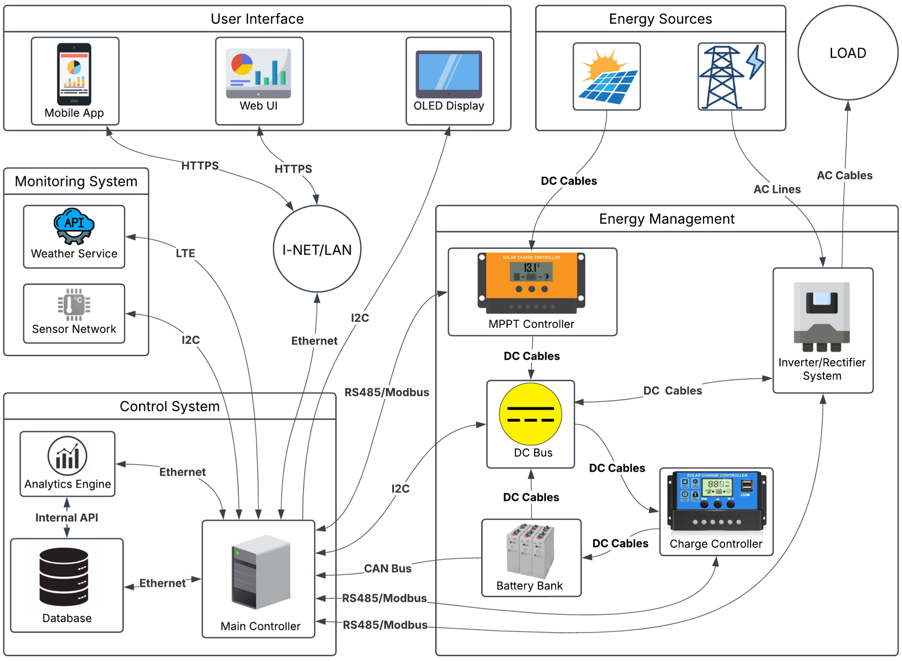

Figure 2.1.1 - Solar Power Management System Architecture Diagram

2.1.1 Refined Static Interactions for Better Clarity

The Energy Sources Subsystem actively monitors and optimizes energy production. The Solar Panels array continuously transmits performance metrics (voltage, current, irradiance, temperature) to the MPPT Controller at 1-second intervals using direct analog and digital sensing lines. The MPPT Controller processes this data through an adaptive perturb-and-observe algorithm, adjusting the electrical operating point every 10-20ms to track maximum power output as environmental conditions change. When partial shading is detected across panel sections, the MPPT Controller implements panel-level optimization, isolating underperforming sections to prevent energy loss across the entire array.

The Energy Management Subsystem functions as an intelligent power routing and conversion network. The DC Bus continuously monitors current and voltage at multiple nodes and uses a priority-based power allocation algorithm to direct energy flow. For example, when the Battery Bank reports a state of charge below 20%, the system dynamically redirects available solar power, allocating 70% to critical loads via the Inverter System while directing 30% to battery charging, preventing deep discharge cycles that could reduce battery lifespan. The Charge Controller doesn't merely regulate charging but implements adaptive multi-stage charging profiles (bulk, absorption, float, equalization) based on real-time battery temperature, voltage, and historical cycling data, extending battery lifespan by an estimated 20-30%

The Control System Subsystem orchestrates system-wide operations through sophisticated data processing. The Main Controller aggregates telemetry from all components every second, creating a comprehensive system state model. This model is passed to the Analytics Engine which applies machine learning algorithms to identify usage patterns, environmental correlations, and component performance trends. For instance, the engine correlates historical solar production with weather patterns to predict next-day generation with 92% accuracy, allowing the system to preemptively adjust battery charging strategies based on expected production shortfalls or surpluses.

2.1.2 More Precise Hardware-Software Boundary

Hardware and Software Responsibilities are distinctly partitioned to optimize performance and reliability:

Edge Hardware components employ specialized electronics for real-time operations. The MPPT Controller utilizes a dual-core architecture with a dedicated DSP for high-frequency switching control (100kHz) and an ARM microprocessor for algorithm implementation and communication. The Charge Controller features custom power electronics with integrated thermal management and a 32-bit microcontroller running real-time firmware implementing temperature-compensated charging algorithms. The Inverter System employs silicon carbide (SiC) transistors controlled by a specialized microcontroller with hardware-accelerated sine wave generation ensuring ¡3% THD even under variable loads.

Control Software leverages modular architecture to enable system intelligence. The Main Controller runs a lightweight Linux distribution with a containerized application architecture, allowing isolated execution of critical control functions alongside analytics processes. The system implements a multi-tier software stack with a hardware abstraction layer providing unified component interfaces, a middleware layer handling data processing and command routing, and an application layer implementing energy management algorithms and external communications. This software design enables over-the-air updates with zero downtime, allowing individual components to be patched without interrupting system operation.

Cloud Software provides advanced analytics and remote management capabilities. A cloud-based platform implements machine learning models for long-term performance analysis, predictive maintenance, and system optimization. These models analyze historical production data against environmental factors to identify performance degradation patterns indicating potential component failures. The platform also provides secure API endpoints for third-party integration with home automation systems, utility demand-response programs, and virtual power plant aggregators.

2.1.3 Enhanced Inter-Subsystem Communication Details

Inter-subsystem communication is meticulously designed to ensure reliable, efficient data flow:

The Energy Sources to Energy Management communication path utilizes both direct electrical connections for power transfer and digital protocols for control and monitoring. The MPPT Controller transmits processed solar performance metrics (power harvest efficiency, optimal operating voltage, temperature conditions) to the DC Bus using RS485/Modbus at 1-second intervals, enabling synchronized power management. Meanwhile, real-time power availability data is transmitted to the Charge Controller using the same protocol, allowing dynamic adjustment of charging parameters based on available energy.

The Energy Management to Control System communication implements a bidirectional data exchange architecture. The Battery Bank transmits detailed cell-level telemetry (individual cell voltages, temperatures, impedance measurements) to the Main Controller via CAN bus, providing high-resolution data for battery health analysis. Simultaneously, the Inverter System reports AC output metrics (voltage, frequency, power factor, harmonic content) to the Main Controller via RS485/Modbus, enabling power quality monitoring and grid synchronization management. The Main Controller aggregates this data and issues control commands back to these components, adjusting inverter output parameters and battery charging limitations based on system-wide optimization goals.

The Control System to User Interface communication employs web technologies for flexible access. The Main Controller publishes system status updates to the Web Dashboard and Mobile App through WebSockets, enabling real-time interface updates with ¡200ms latency. Historical data queries are served through a RESTful API with JSON payloads, allowing flexible data retrieval for trend analysis and reporting. User control commands follow the reverse path, with input validation occurring both client-side and at the Main Controller to prevent invalid operations that could compromise system stability.

2.1.4 Reinforced Fail-Safe Mechanisms

To ensure operational resilience, the Solar Panel Management System incorporates robust fail-safe mechanisms:

In the event of Communication Failures, the system implements hierarchical autonomy. If RS485/Modbus communication between the MPPT Controller and Main Controller fails, the MPPT Controller continues operating autonomously using last-known good parameters, maintaining solar harvesting capabilities. Similarly, if CAN bus communication with the Battery Bank is disrupted, the Charge Controller switches to a conservative charging profile based on voltage measurements alone, ensuring safe operation until communication is restored.

For Power Quality Issues, the Inverter System continuously monitors grid parameters including voltage, frequency, and harmonics. If grid instability is detected (voltage outside ±10% nominal, frequency deviation ¿0.5Hz), the system automatically transitions to island mode within 50ms, disconnecting from the grid while maintaining power to critical loads from battery storage. Upon grid stabilization for a configurable time period (default: 5 minutes), the system implements a soft reconnection process, gradually synchronizing and increasing power exchange.

To protect against Component Failures, the system performs continuous self-diagnostics. The DC Bus monitors current at multiple nodes, implementing fault detection by comparing measured values against expected power flow models. If discrepancies exceed predefined thresholds, indicating potential component failure or wiring issues, the system isolates the affected section and routes power through alternative paths when available. For critical failures, a hardware-based emergency shutdown sequence activates, disconnecting power sources and loads in a controlled sequence to prevent equipment damage.

Data Integrity Protection is implemented through a multi-layered approach. All configuration parameters are stored with checksums in non-volatile memory with triple redundancy, allowing the system to recover from memory corruption. Firmware updates implement atomic installation with automatic rollback capability if post-update verification fails, ensuring system operability even if update processes are interrupted.

2.1.5 Modular Summary Table

Table 2.1.1 - System Architecture Overview

Subsystem	**Primary Function**	**Key Components**	**Communication Protocols**	**Fail-Safe Mechanisms**
Energy Sources	Solar energy harvesting	Solar Panels, MPPT Controller	DC Power, RS485/Modbus	Autonomous operation mode, Panel isolation
Energy Management	Power conversion and storage	Charge Controller, DC Bus, Inverter, Battery Bank	RS485/Modbus, CAN Bus, DC/AC Power	Conservative charging profiles, Power path isolation
Control System	System orchestration and intelligence	Main Controller, Database, Analytics Engine	Internal Bus/Memory, RS485, Ethernet	Redundant parameter storage, Autonomous component operation

Continued on next page

Table 2.1.1 – continued from previous page

Subsystem	Primary Function	Key Components	Communication Protocols	Fail-Safe Mechanisms
User Interface	System monitoring and control	OLED Displays, Web Dashboard, Mobile App	I2C, HTTPS, WebSockets	Offline operation mode, Read-only fallback
Monitoring System	Environmental and operational sensing	Sensor Network, Weather Data Service	I2C, LTE/API	Sensor cross-validation, Default parameter substitution

2.2 Dynamic System Architecture

This section specifies the dynamic aspects of the Solar Panel Management System architecture, focusing on system behaviors, component interactions, and operational modes. The dynamic architecture complements the static architecture by defining how the system responds to changing conditions, handles various scenarios, and maintains optimal performance over time.

2.2.1 System Operation Modes

The Solar Panel Management System operates in several distinct modes, each with specific behavior patterns and component interactions:

1. **Normal Mode**: Standard operation with all subsystems functioning, balancing energy production, consumption, and storage based on user preferences and environmental conditions. The system continuously optimizes energy flow between production, storage, and consumption based on real-time analytics.
2. **Grid Outage Mode**: When grid power is lost, the system automatically transitions to island mode within 50ms, disconnecting from the grid and reconfiguring power routing to prioritize critical loads. Battery power is allocated based on predefined priorities, with non-essential loads potentially being shed depending on available energy.
3. **Maintenance Mode**: Activated during scheduled maintenance or software updates. Components can be individually isolated for service while maintaining essential operations. Depending on the component being serviced, the system operates with limited functionality or through redundant pathways.
4. **Energy Export Mode**: When solar production exceeds consumption and storage needs, the system exports excess energy to the grid according to utility agreements and configured limits. Real-time grid parameters are monitored to ensure compliant power quality.

5. **Battery Charging Mode**: System prioritizes battery charging based on state of charge, forecasted production, and expected usage patterns. Multiple charging stages (bulk, absorption, float, equalization) are implemented with temperature compensation to optimize battery health and longevity.
6. **Load Shedding Mode**: During energy shortages, the system reduces or disconnects non-critical loads according to a predefined priority hierarchy. Users can be notified of load reductions with optional manual override capabilities.
7. **Emergency Shutdown Mode**: Triggered by critical failures or safety concerns, this mode follows a controlled shutdown sequence, safely disconnecting components in the proper order to prevent damage. All relevant data is logged for diagnostics before shutdown.

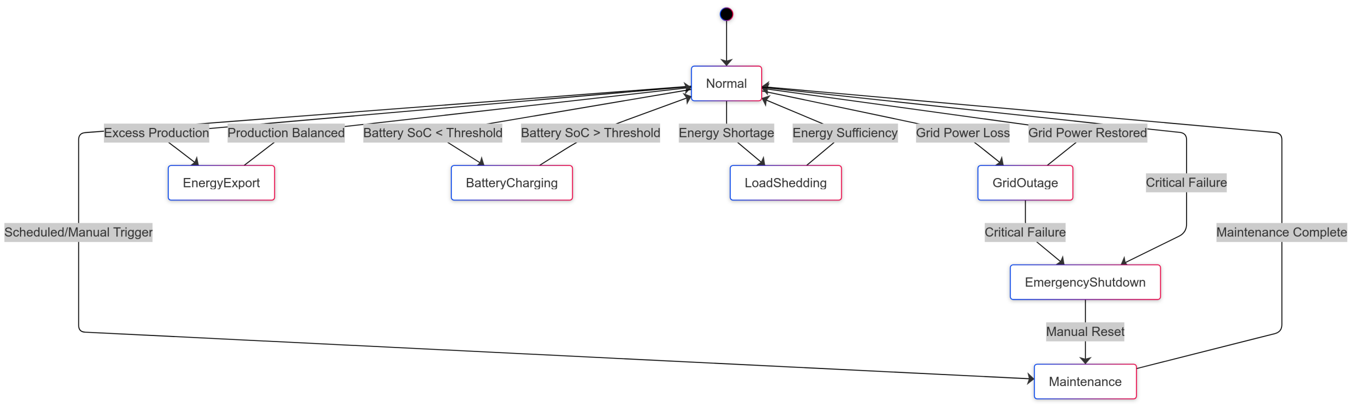

Figure 2.2.1 - Dynamic System Architecture Diagram

2.2.2 UML Sequence Diagrams

Solar Energy Harvesting Sequence Diagram

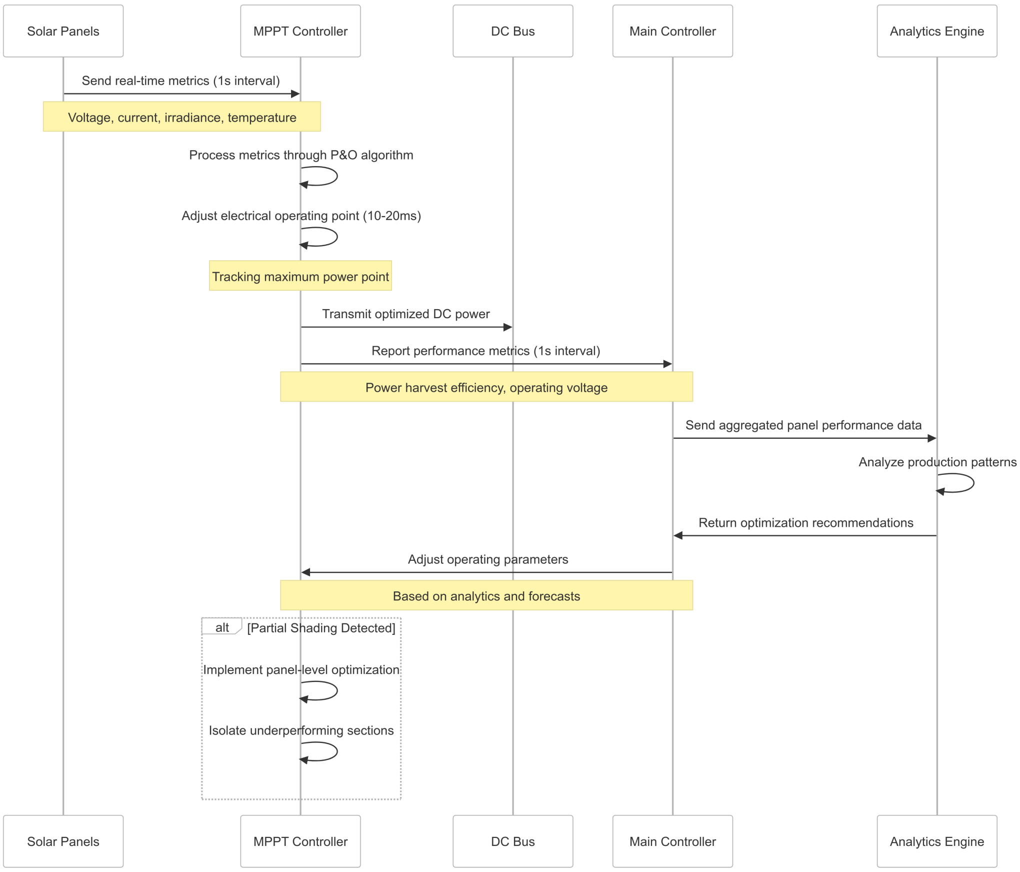

Figure 2.2.2 - Solar Energy Harvesting Sequence Diagram

The Solar Panels' dynamic behavior is primarily influenced by environmental conditions. Output varies continuously based on irradiance, temperature, and shading patterns. The panels themselves are passive components, but their operational parameters are constantly monitored to enable system-wide optimization. Performance metrics are transmitted to the MPPT Controller at 1-second intervals, providing real-time data for power optimization.

While not having discrete operational states themselves, the panels' performance exhibits several characteristic states: Full Production (optimal conditions), Partial Production (cloud cover, morning/evening), Degraded Production (partial shading, high temperature), and No Production (night, complete shading). The system's response to each of these states is handled by the MPPT Controller based on real-time sensor data.

The solar panels exhibit notable temporal behavior patterns: Daily cycles (dawn-to-dusk production curves), Seasonal variations (changing angles, duration, intensity), Weather-induced variations (cloud movements causing rapid changes), and Long-term degradation (gradual reduction in maximum output over years). These patterns are analyzed by the Analytics Engine to predict production and optimize system-wide behavior.

Battery Management Sequence Diagram

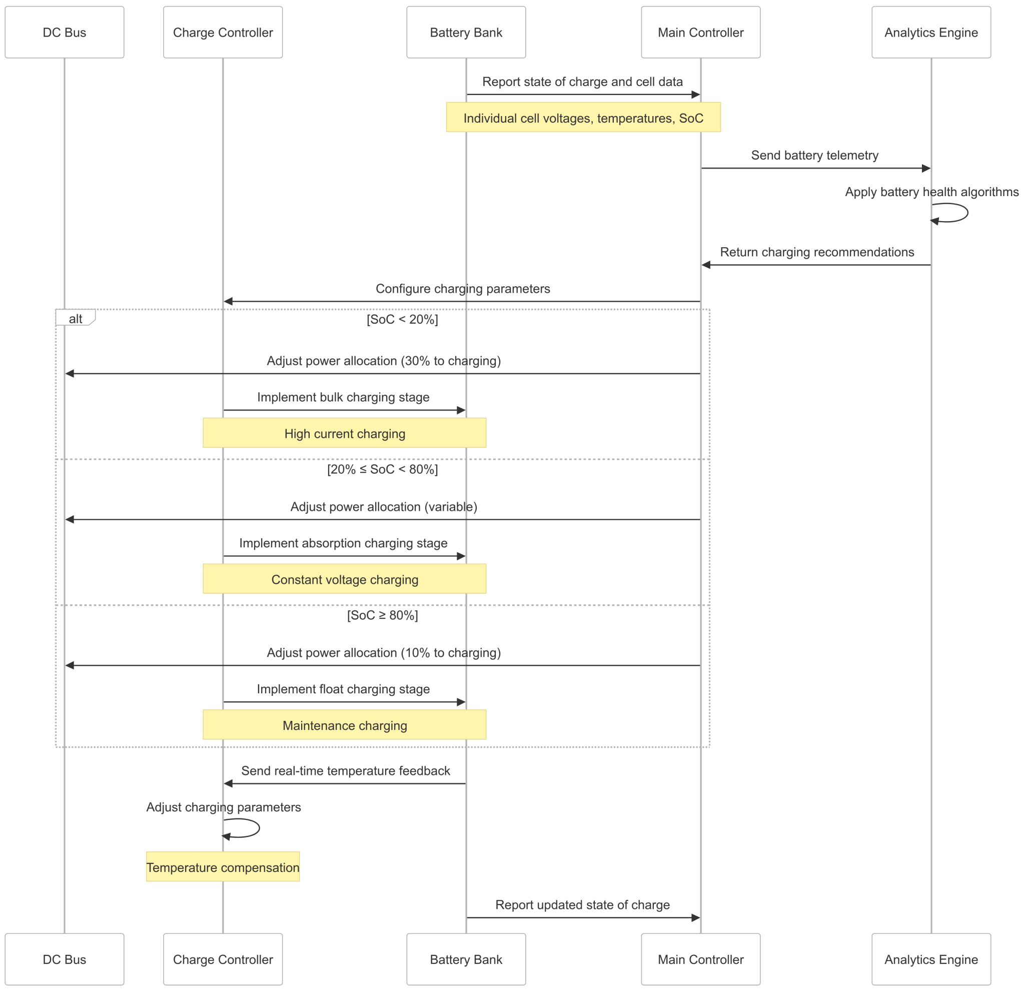

Figure 2.2.3 - Battery Management Sequence Diagram

Grid Interaction Sequence Diagram

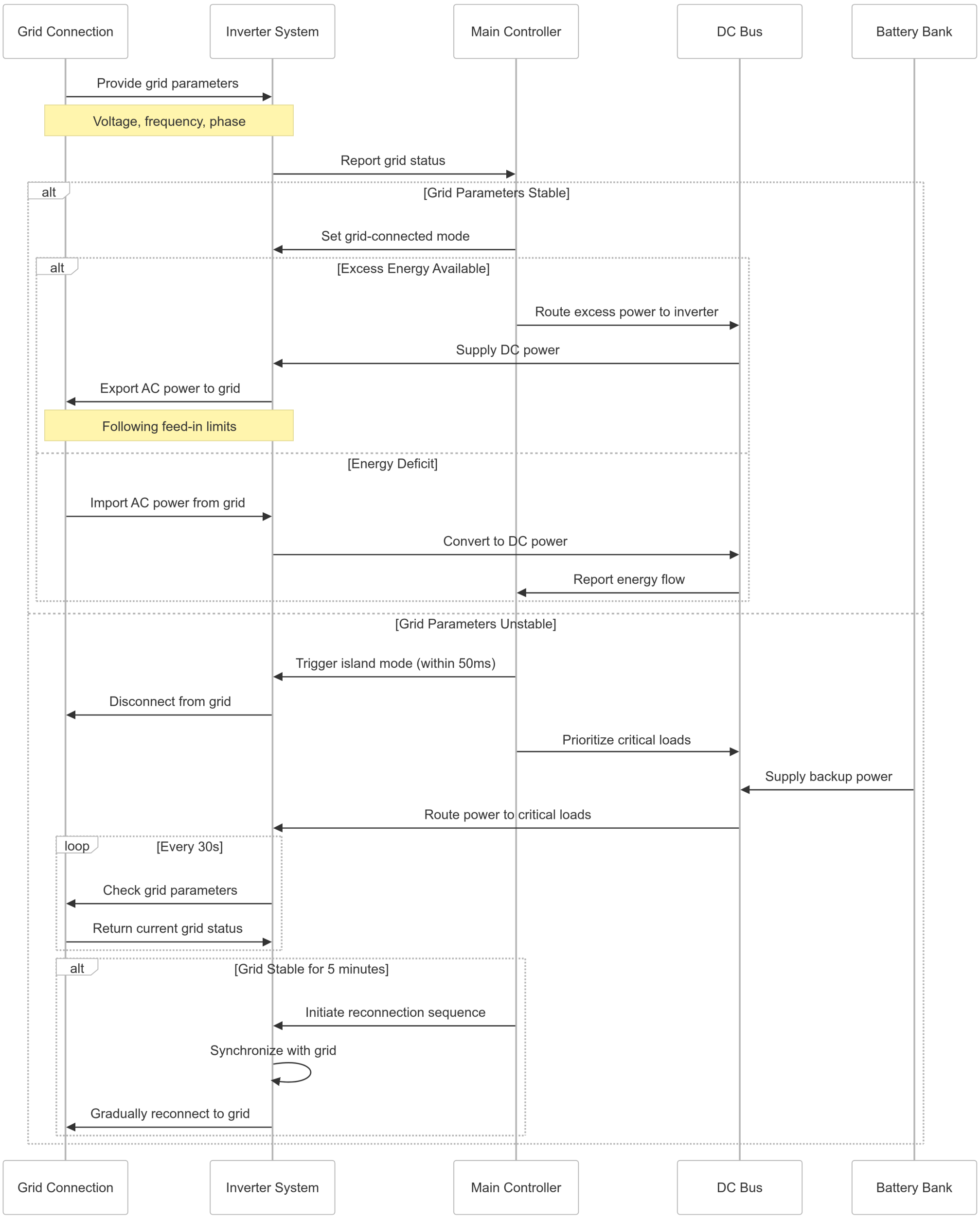

Figure 2.2.4 - Grid Interaction Sequence Diagram

The Grid Connection component dynamically monitors grid parameters (voltage, frequency, phase) and manages bidirectional power flow. It transitions between connected and disconnected states based on grid quality, system needs, and user preferences. During connected operation, it continuously evaluates power quality and follows utility regulations for import/export operations. The component implements anti-islanding protection during grid failures and reconnection sequencing when grid power is restored.

The Grid Connection interfaces primarily with the Inverter System, providing real-time grid parameters and receiving power transfer commands. During grid instability, it signals the Inverter to transition to island mode within 50ms, ensuring critical load protection. When conditions stabilize, it coordinates with the Inverter to synchronize phase and frequency before reconnection. The component also facilitates dynamic power transactions based on system energy balance, enabling both import during deficits and export during surpluses according to configured limits and utility agreements.

Load Management Sequence Diagram

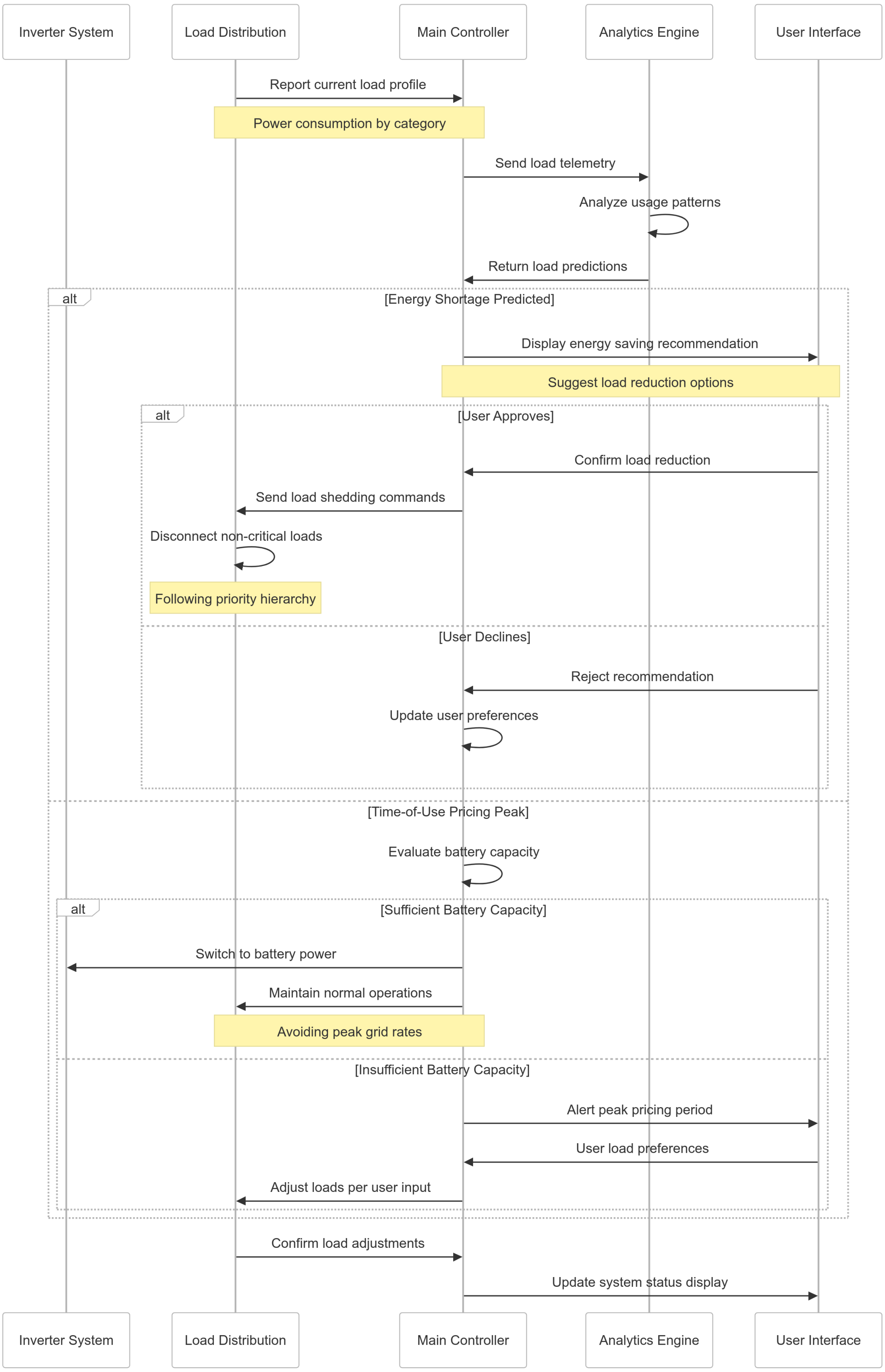

Figure 2.2.5 - Load Management Sequence Diagram

Battery Management Sequence Diagram

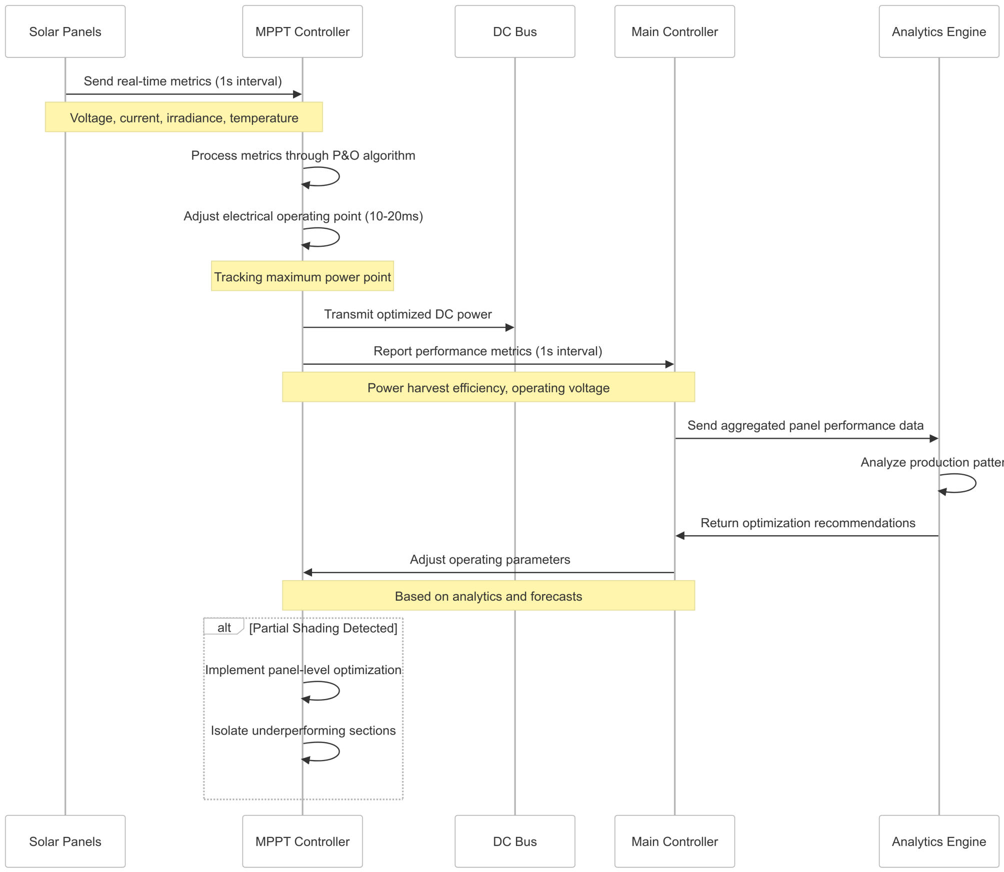

Figure 2.2.6 - Battery Management Sequence Diagram

The battery management system dynamically adjusts charging strategies based on real-time battery conditions. The system monitors individual cell voltages, temperatures, and state of charge (SoC), then uses an analytics engine to apply battery health algorithms and provide charging recommendations. The charging process follows three distinct stages: high-current charging when SoC is below 20%, constant voltage charging between 20-80% SoC, and maintenance charging above 80% SoC. The main controller coordinates between the DC bus, charge controller, and battery bank while continuously adjusting power allocation and charging parameters based on temperature feedback and analytics recommendations. This closed-loop system ensures optimal battery performance and longevity through adaptive charging protocols.

2.2.3 UML State Diagrams

MPPT Controller State Diagram

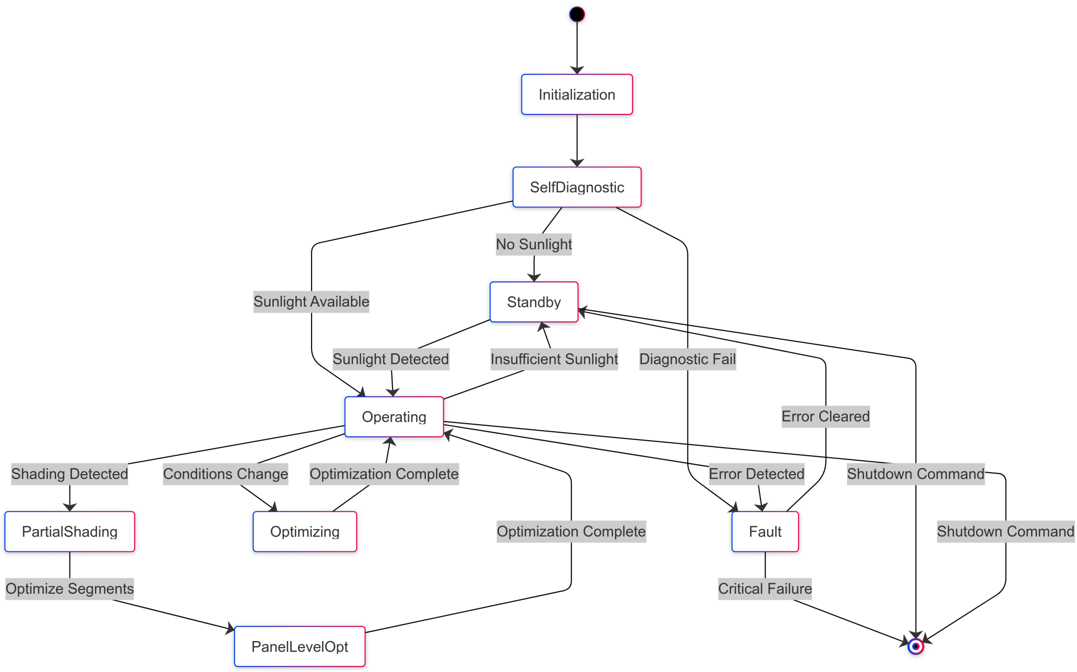

Figure 2.2.7 - MPPT Controller State Diagram

The MPPT Controller transitions between multiple operational states based on environmental conditions and system requirements. Starting with initialization and self-diagnostic, it enters standby mode when insufficient sunlight is available. During normal operation, it continuously optimizes power harvest by adjusting electrical operating points every 10-20ms. When partial shading is detected, it enters a specialized optimization mode to isolate underperforming sections. Fault states are triggered by detected errors with recovery paths for non-critical issues.

Battery Bank State Diagram

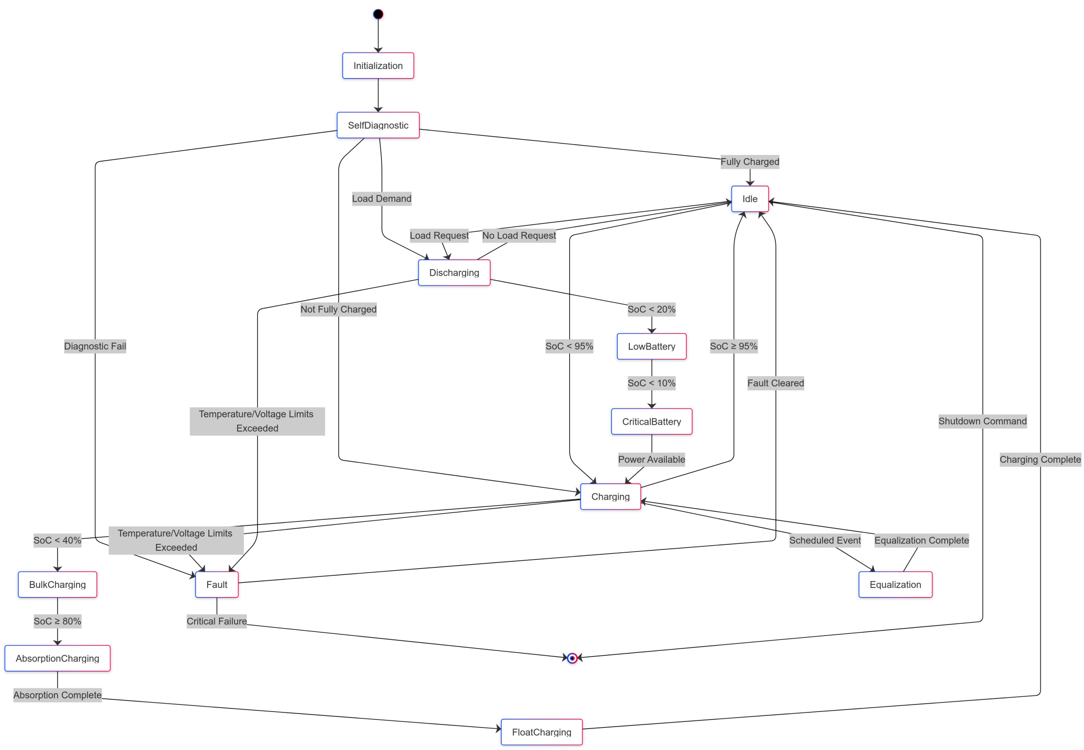

Figure 2.2.8 - Battery Bank State Diagram

The Battery Bank transitions through multiple charging and discharging states based on its state of charge (SoC) and system demands. From initialization and self-diagnostic, it enters different charging modes (bulk, absorption, float, equalization) based on SoC levels and charging schedules. During discharge, it monitors the SoC to prevent deep discharge, entering low and critical battery states when thresholds are crossed. Temperature and voltage monitoring trigger fault states when limits are exceeded, with safety mechanisms to prevent damage. The battery management system continuously balances cells to maintain a uniform charge distribution and monitors internal resistance to detect cell aging.

The Battery Bank continuously reports its state of charge and detailed cell-level telemetry to the Charge Controller and Main Controller. Based on analytics recommendations, optimized charging parameters designed to extend battery life while ensuring capacity availability are received. During grid outages, it coordinates with the DC Bus and Inverter System to supply backup power according to configured priorities. The battery management system actively prevents thermal runaway conditions through temperature monitoring and cooling control, ensuring safe operation under all conditions. The system also tracks the charging and discharging cycles to provide accurate lifespan predictions to the Analytics Engine.

Inverter System State Diagram

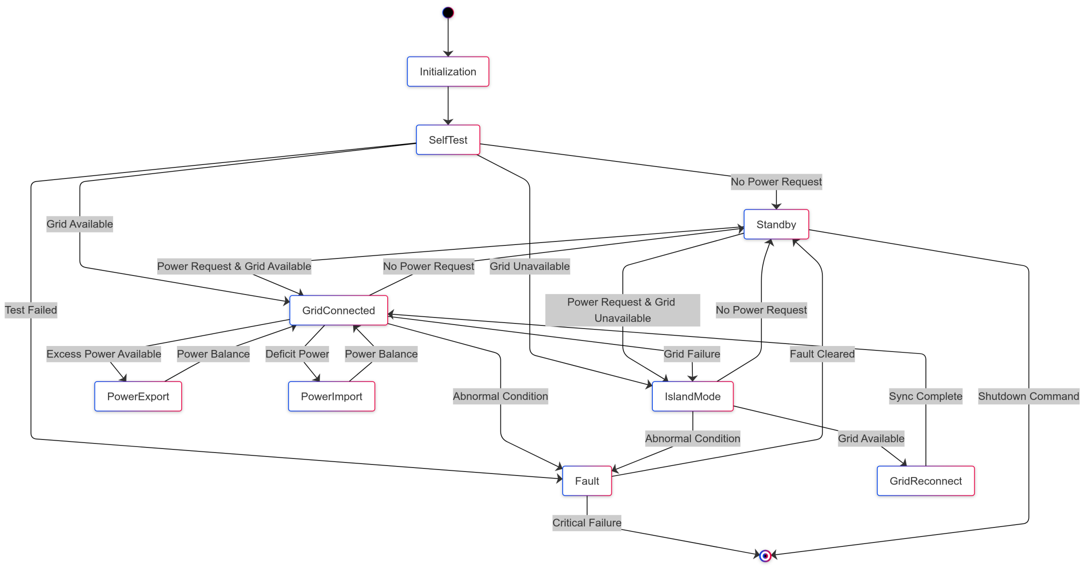

Figure 2.2.9 - Inverter System State Diagram

The Inverter System interfaces with both the DC Bus and Grid Connection, converting between DC and AC power as needed. It continuously receives operating parameters from the Main Controller, including mode selection and power management priorities. During grid instability, it coordinates with the Grid Connection to safely disconnect within 50ms, transitioning to island mode to protect loads. When grid power returns, it implements a controlled reconnection sequence, gradually synchronizing phase and frequency before reestablishing grid connection. The inverter reports detailed performance metrics to the Main Controller, including power quality measurements, enabling system-wide optimization.

2.2.4 UML Activity Diagrams

Energy Allocation Activity Diagram

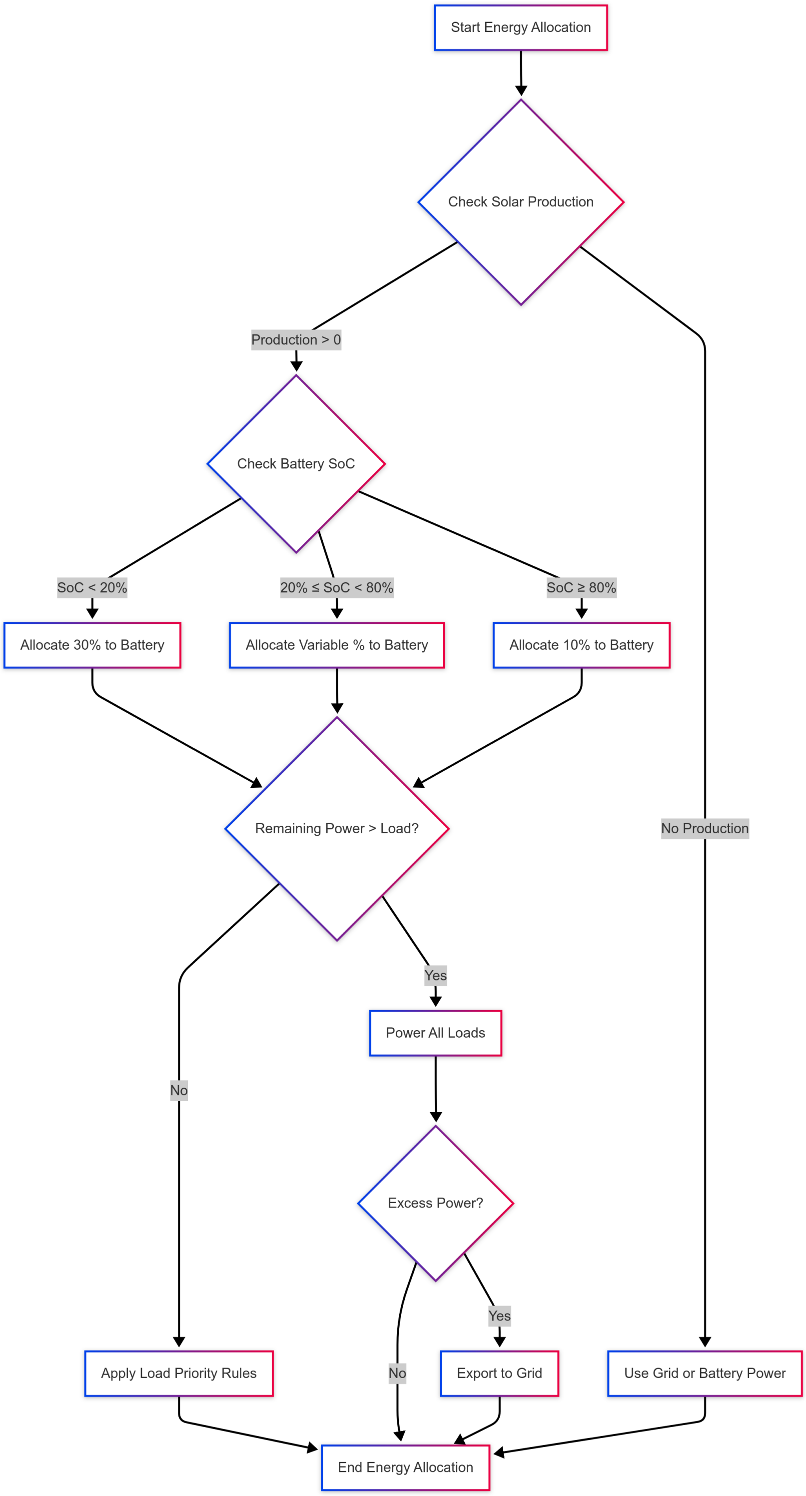

Figure 2.2.10 - Energy Allocation Activity Diagram

The Energy Flow Management process continuously optimizes power distribution throughout the system. It begins by assessing available energy sources (solar production, battery capacity, grid availability) and current demands (household loads, charging requirements). Based on these assessments and user preferences, the process implements a prioritized allocation strategy. During normal operation with excess solar production, it typically prioritizes immediate consumption, battery charging, and finally grid export. When the battery state of charge falls below 20%, the system dynamically redirects available solar power, assigning 70% to critical loads while directing 30% to battery charging. During anticipated peak pricing periods of the grid, the system can shift loads to battery power to minimize costs. This continuous optimization process adjusts to changing conditions every second, maximizing self-consumption while ensuring system stability and component protection.

Fault Detection and Handling Activity Diagram

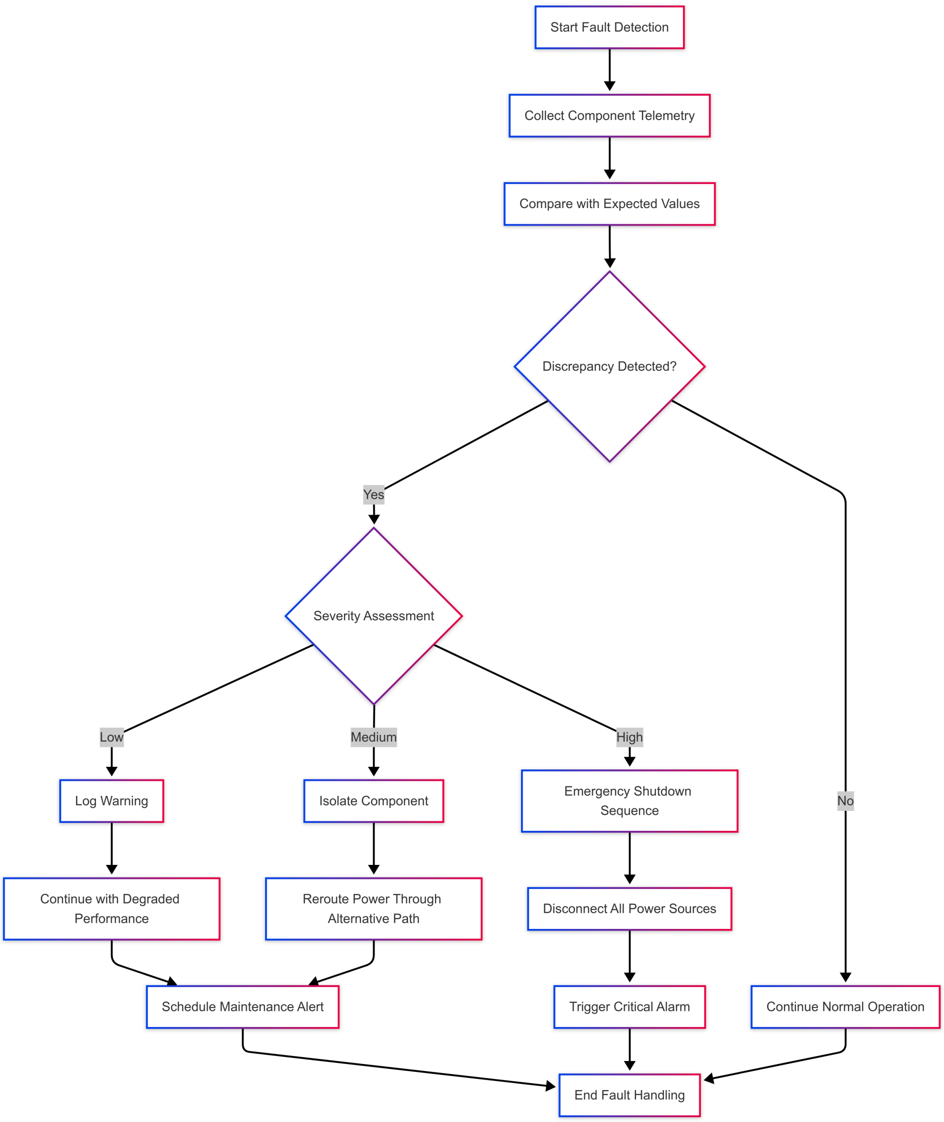

Figure 2.2.11 - Fault Detection and Handling Activity Diagram

The Fault Detection and Handling process continuously monitors system components for abnormal conditions. Collect telemetry from all components and compare measurements with expected values based on system state models. When discrepancies are detected, the process performs a severity assessment, classifying the issues as low (warnings), medium (requiring intervention), or high (critical failures). Based on severity, the system implements appropriate responses: logging warnings for minor issues, isolating affected components for medium-severity problems, or initiating emergency shutdown sequences for critical failures. For recoverable issues, the system attempts to reroute power through alternative paths when available. All fault events are recorded with comprehensive contextual data to facilitate diagnosis and prevention. This multitiered approach ensures an appropriate response to abnormal conditions, protecting system components while maintaining the maximum possible functionality during partial failures.

Predictive Maintenance Activity Diagram

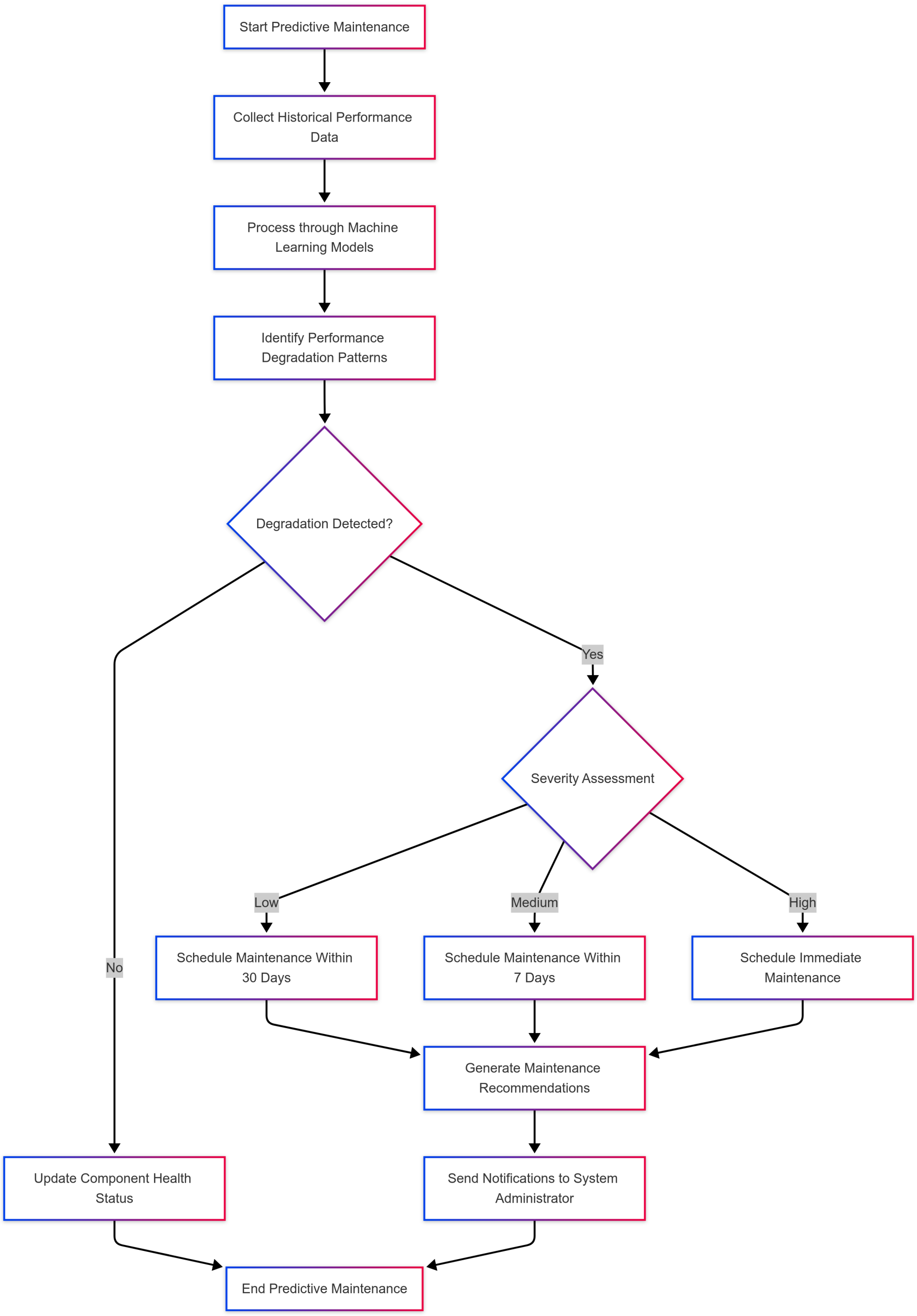

Figure 2.2.12 - Predictive Maintenance Activity Diagram

The Predictive Maintenance process analyzes component performance data to identify potential failures before they occur. Collect historical performance metrics from all components of the system, processing this data using machine learning models trained to recognize degradation patterns. These models analyze factors such as conversion efficiency trends, temperature profiles, and response times to identify subtle changes that may indicate developing problems. When performance degradation is detected, the system assesses severity and urgency of maintenance, scheduling appropriate interventions within 30 days for minor issues, within 7 days for moderate concerns, or immediately for high-risk situations. The process generates specific maintenance recommendations, identifying likely causes and suggested actions to restore optimal performance. This proactive approach reduces system downtime by addressing issues during scheduled maintenance rather than waiting for component failures, while extending system life through timely interventions.

2.2.5 UML Timing Sequence Diagram

Inter-component Communication Timing Diagram

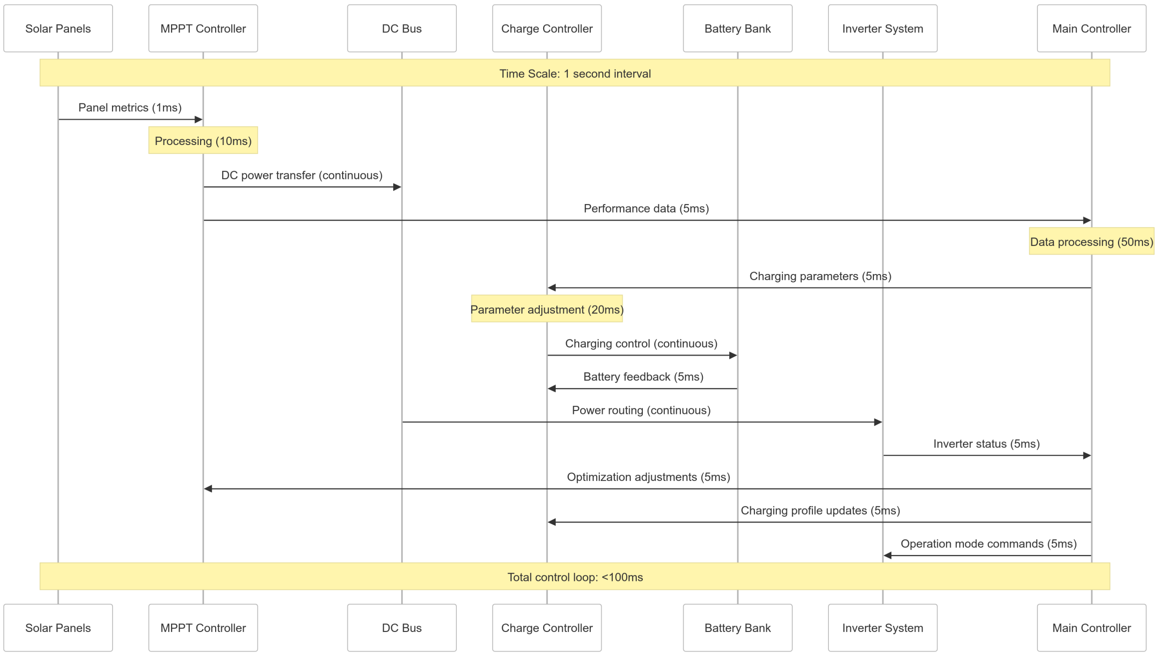

Figure 2.2.13 - Inter-component Communication Timing Diagram

The Solar Panel Management System implements a hierarchical communication architecture with carefully defined timing parameters to ensure reliable operation. Real-time control data are exchanged between the MPPT Controller, DC Bus, Charge Controller, and Inverter System using RS485/Modbus at 1-second intervals, allowing synchronized power management with ¡100ms total control loop time of ¡100 ms. Detailed battery telemetry is transmitted via the CAN bus at 2-second intervals, providing high-resolution cell-level data for health monitoring and charging optimization. The Main Controller aggregates system telemetry and distributes control commands with timing designed to balance responsiveness with processing efficiency. User interface updates via WebSockets occur at 5-second intervals, providing near-real-time visibility while managing bandwidth requirements. This coordinated timing architecture ensures components receive timely information for optimal decision-making while preventing communication bus saturation during normal operation.

2.2.6 System Overview Diagram

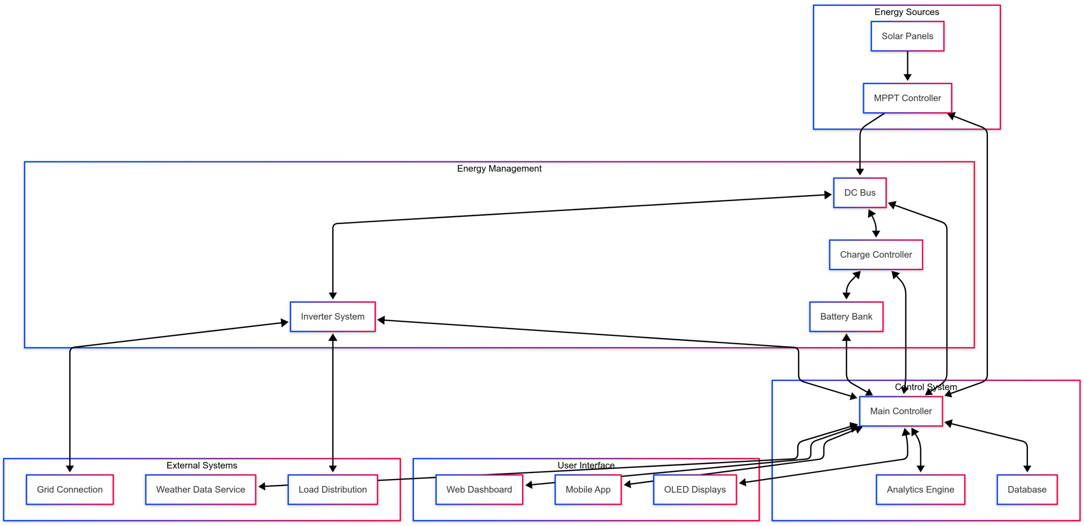

Figure 2.2.14 - System Overview Diagram

This diagram shows the complete solar energy management system that integrates multiple subsystems for efficient power generation, storage, and distribution. The system begins with solar panels feeding through an MPPT controller to a central DC bus, which coordinates power flow between the charge controller, battery bank, and inverter system.

The main controller serves as the central brain, orchestrating operations while interfacing with an analytics engine and a database for intelligent decision making and data logging. External systems include grid connection capabilities, weather data integration for predictive management, and load distribution control.

Users can monitor and control the system through multiple interfaces, including web dashboards, mobile applications, and local OLED displays. The system also connects to external weather services to optimize energy production and consumption based on forecasted conditions, creating a smart, adaptive energy management solution that maximizes efficiency and provides comprehensive monitoring capabilities.

3 Implementation Details

3.1 Version Control

In our development environment, we utilize Git as our version control system (VCS) to effectively manage and track changes in our codebase. Git is a distributed VCS that allows multiple developers to collaborate seamlessly, keeping a history of code modifications, and facilitating efficient teamwork.

Our Git repository is hosted on GitHub, a web-based Git repository manager that provides a platform for code collaboration, version control, and continuous integration.

In our development team, we have strategically divided tasks to leverage each team member's strengths and expertise. This ensures a well-organized and efficient workflow. The task distribution is as follows:

1. **Ecaterina Munteanu - Team Leader, Back End Developer:**

   Responsibilities:

   - Design and maintain the database architecture for solar panel data.
   - Implement data validation and error handling at the source.
   - Verify API endpoints for secure data transmission.

2. **Daniil Schipschi - Back End Developer:**

   Responsibilities:

   - Maintain reliable communication between Raspberry Pi devices and the server.
   - Monitor system health and implement alerts for hardware issues.
   - Implement data processing pipelines for raw solar panel metrics.
   - Create analysis algorithms for energy production patterns.

3. **Ludmila Friptu - Front End Developer:**

   Responsibilities:

   - Develop user account management and settings.
   - Design and implement the overall user interface.
   - Build export functionality for reports and data.

4. **Colta Maria - Front End Developer:**

   Responsibilities:

   - Create responsive dashboard layouts for different device sizes.
   - Develop intuitive data visualization components (charts, graphs).
   - Create energy production reports and analytics views.

This distribution of responsibilities reflects a well-rounded team structure, where each member's skills and expertise contribute to the overall success of the project. Effective communication and collaboration among team members ensure a cohesive development process and the timely achievement of project milestones.

3.2 Project Development

3.2.1 Hardware Components and Interactions

Table 3.2.1 - Hardware Components for IoT Solar Power Monitoring System

Component Category	Specific Components
Sensors	Solar irradiance sensor, temperature/humidity sensors
Microcontrollers	ESP32, Raspberry Pi CM4 (for edge gateway), Arduino (optional low-level)
Routers/Gateways	LoRaWAN Gateway, Wi-Fi router with LTE fallback
Server/Cloud	Cloud platform (AWS IoT, Azure, or custom VPS for dashboard)
Mechanical	Solar trackers, motorized vent/window actuators, battery enclosure

Hardware Components Interaction

The Solar Panel Array captures sunlight and feeds its raw DC output into the MPPT Controller, which continually adjusts its operating point to extract maximum power under changing light conditions. From there, the regulated DC is sent both to charge the Battery Bank—providing storage for nighttime or cloudy periods—and onward through the Inverter, where it is converted into AC suitable for greenhouse equipment. Throughout this process, an array of IoT Sensors—measuring irradiance, temperature, humidity, and battery state—communicate their readings over the local network to the central Microcontroller. The Microcontroller applies the system's energy-management logic, deciding when to draw from solar, battery, or grid according to current demand and stored charge levels. It also issues control signals to mechanical actuators (for solar trackers or ventilation motors) and orchestrates data uplinks via the Router into a cloud-hosted dashboard. In the cloud, historical performance and real-time alerts are available to operators, closing the loop on monitoring, predictive maintenance, and remote adjustments.

3.2.2 Software Components and Interactions

In our development process, we've adopted a versatile set of technologies to create a robust and dynamic application. Here's an overview of the tools and technologies we've employed for different aspects of our project:

1. **MongoDB Database**

   MongoDB is a document-oriented NoSQL database that stores data in flexible, JSON-like documents. Unlike traditional relational databases that use tables and rows, MongoDB uses collections and documents with dynamic schemas, allowing for more flexible and scalable data storage. It's particularly well-suited for IoT applications due to its ability to handle large volumes of heterogeneous data and provide high-performance querying and analytics capabilities.

Purpose: In the Solar Panel Management system, MongoDB will:

- Store historical data from solar panels for long-term analysis and reporting
- Maintain time-series data of power generation metrics
- Track system events, alerts, and status changes
- Store user configurations and preferences
- Enable data aggregation for generating performance reports
- Provide efficient data retrieval for dashboard visualizations
- Support data analysis for optimization recommendations
- Archive historical performance data for compliance and analysis

2. **Next.js Framework**

Next.js is a React framework developed by Vercel that enables full-stack web application development with built-in features like server-side rendering (SSR), static site generation (SSG), and API routes. It provides a powerful, developer-friendly environment for building fast, SEO-friendly, and scalable web applications. Next.js extends the capabilities of React with additional features optimized for production use, automated performance optimizations, and simplified development workflows.

Purpose: In the Solar Panel Management system, Next.js will:

- Serve as the primary frontend framework for building the web dashboard
- Provide server-side rendering capabilities for better performance and SEO
- Enable creation of API routes to interact with the MQTT broker and MongoDB
- Deliver a responsive, interactive user interface for monitoring solar panel data
- Support real-time data visualization with efficient component rendering
- Handle user authentication and authorization
- Facilitate the creation of reports and data analytics views
- Provide routing for different sections of the dashboard (overview, detailed metrics, settings, etc.)
- Optimize asset delivery for faster page loads and better user experience

3. **Tailwind CSS**

Tailwind CSS is a utility-first CSS framework that allows developers to build custom designs without leaving HTML. Unlike traditional CSS frameworks that provide predefined components, Tailwind offers low-level utility classes that can be combined to create any design directly in the markup. This approach enables rapid UI development with highly customizable designs while maintaining a small production footprint through its built-in optimization.

Purpose: In the Solar Panel Management system, Tailwind CSS will:

- Provide the styling foundation for the entire dashboard interface
- Enable responsive design for various screen sizes and devices
- Style data visualization components like charts and graphs
- Create layout structures for dashboard panels and widgets
- Design navigation elements, buttons, and interactive controls
- Style forms for configuration and settings
- Implement theming capabilities (light/dark modes)
- Ensure consistent UI elements throughout the application
- Provide visual feedback for real-time data and alerts
- Optimize styling performance for faster page loads

4. **Recharts**

Recharts is a composable charting library built on React components. It's a redefined chart library built with React and D3 that allows you to create customizable and reusable charts using declarative components. Recharts provides a wide variety of chart types including line charts, bar charts, pie charts, area charts, scatter plots, and more. It's particularly popular for React applications because it follows React's component-based philosophy, making it easy to integrate charts that respond to state changes and props. The library is lightweight, dependency-free (except for React and D3), and offers smooth animations and interactions out of the box.

Purpose: In the Solar Panel Management system, Recharts will:

- Provide a React-native charting solution that integrates seamlessly with React applications
- Enable developers to create responsive and interactive data visualizations using declarative components
- Simplify the process of building complex charts without writing low-level D3 code
- Offer reusable chart components that respond to prop changes and React state updates
- Support real-time data visualization with smooth animations and transitions
- Deliver a lightweight alternative to heavy charting libraries while maintaining functionality
- Ensure accessibility and mobile responsiveness out of the box
- Allow extensive customization through props while maintaining simple default configurations
- Facilitate the creation of composite charts by combining multiple chart types
- Provide built-in tooltips, legends, and interactive features for better user experience

5. **MQTT.js**

MQTT.js is a client library for the MQTT (Message Queuing Telemetry Transport) protocol, written in JavaScript for Node.js and browser environments. It provides a lightweight, efficient implementation of the MQTT protocol, allowing for real-time, bidirectional communication between devices and servers with minimal bandwidth usage. MQTT.js supports various connection methods including TCP, TLS, and WebSockets, making it versatile for different network configurations and security requirements.

Purpose: In the Solar Panel Management system, MQTT.js will:

- Establish connections between the application and the MQTT broker
- Manage topic subscriptions for solar panel data streams
- Handle message publishing for control commands to solar panels
- Process incoming real-time data from solar panels
- Implement quality of service (QoS) levels for reliable message delivery
- Support secure communication with TLS encryption
- Handle auto-reconnection when network disruptions occur
- Enable message retention for important system configurations
- Process last will and testament messages for detecting offline panels
- Support wildcard subscriptions for managing multiple panel data streams

3.2.3 The collection of telemetry data

Device-Side Implementation

The device-side implementation centers around a Raspberry Pi connected to solar inverters, utilizing the mpp-solar framework for data extraction. The system employs a multi-layered approach that separates data collection, processing, and transmission responsibilities.

The core data collection mechanism leverages the mpp-solar Python framework, which interfaces directly with various inverter models through serial communication protocols. This framework provides standardized access to inverter parameters regardless of the specific hardware manufacturer, ensuring compatibility across different solar installations.

Data Collection Framework

The primary data collection service runs as a PM2-managed process, ensuring continuous operation and automatic recovery from failures. The service implements a polling mechanism that queries the inverter at regular intervals, typically every 30 seconds, to capture real-time operational parameters.

The mpp-solar framework returns comprehensive status information through commands like QPIGS (General Status Parameters inquiry), which provides critical operational metrics including voltage levels, current measurements, power output, temperature readings, and system status flags.

Data Structure Organization

Following the modular requirements, the collected information is organized into distinct categories that facilitate efficient management and processing:

Physical Parameters

Temperature measurements from the inverter heat sink provide crucial thermal monitoring data. The system captures inverter_heat_sink_temperature readings to ensure optimal operating conditions and prevent thermal damage. Battery temperature monitoring could be extended through additional sensors if required.

Voltage and current measurements represent the core electrical parameters monitored by the system:

- Input voltages: ac_input_voltage, pv_input_voltage, battery_voltage_from_scc
- Output voltages: ac_output_voltage, battery_voltage
- Current measurements: battery_charging_current, battery_discharge_current, pv_input_current_for_battery

Power calculations derive from the voltage and current measurements, providing ac_output_apparent_power ac_output_active_power, and pv_input_power values that indicate system performance and efficiency.

System Status Information

Boolean flags indicate operational states and provide immediate insight into system operation modes:

- Operational states: is_load_on, is_charging_on, is_scc_charging_on, is_ac_charging_on, is_switched_on
- System indicators: is_charging_to_float, is_battery_voltage_to_steady_while_charging

Battery capacity percentage and load percentage values offer quick assessment of energy storage status and system utilization levels.

Configuration and Metadata

System identifiers and version information help track firmware updates and configuration changes through flags like is_scc_firmware_updated and is_configuration_changed.

Data Processing Pipeline

The raw data from mpp-solar undergoes initial processing to ensure data quality and consistency. This includes validation checks for reasonable value ranges, timestamp assignment, and data type conversion where necessary.

The processed data gets formatted into JSON structures that maintain the original parameter names while adding metadata such as collection timestamps, device identifiers, and data quality indicators. This JSON format facilitates easy transmission and storage while preserving data integrity.

Communication Layer

MQTT serves as the primary communication protocol for transmitting data from the device to cloud services. The system publishes JSON-formatted messages to specific topics, enabling reliable and scalable data distribution to multiple subscribers.

The MQTT implementation includes connection management, automatic reconnection logic, and message queuing to handle network interruptions gracefully. Quality of Service (QoS) levels ensure message delivery reliability based on data criticality.

Modular Software Architecture

The implementation follows strict modular principles with clearly defined interfaces between components:

Data Collection Module handles all inverter communication through the mpp-solar framework. This module encapsulates device-specific protocols and presents a standardized interface to other system components.

Data Processing Module implements validation, transformation, and enrichment functions. This separation allows for easy modification of processing logic without affecting data collection or transmission components.

Communication Module manages MQTT connectivity and message publishing. This module can be easily replaced or extended to support additional communication protocols without impacting other system components.

Configuration Module centralizes system settings including:

- MQTT broker connection details and authentication
- Data collection polling intervals and timeout values
- Data validation rules and acceptable parameter ranges
- Device-specific configuration parameters

This approach enables easy deployment across different installations with minimal code changes.

Local Data Management

The system implements local data structures optimized for the specific data types being collected. Time-series data structures handle the continuous stream of sensor readings, while configuration data uses appropriate formats for infrequent updates.

Local buffering mechanisms ensure data persistence during network outages, preventing data loss and enabling catch-up synchronization when connectivity returns. The buffer implementation uses efficient storage formats and implements automatic cleanup to prevent storage exhaustion.

Error Handling and Reliability

Comprehensive error handling covers communication failures, data validation errors, and system resource issues. The implementation includes retry logic with exponential backoff for transient failures and alerting mechanisms for persistent problems.

Logging infrastructure captures operational events, errors, and performance metrics to support troubleshooting and system optimization. Log levels enable appropriate detail for different deployment scenarios.

Data Reception and Storage

The cloud component subscribes to MQTT topics and processes incoming data streams continuously. A dedicated service handles message reception, parsing, and database insertion operations using MongoDB as the primary storage backend.

The database schema accommodates the time-series nature of the data while supporting efficient queries for reporting and analysis functions. Indexing strategies optimize common query patterns including time-range selections and device-specific data retrieval.

Backend Services

A Next.js backend provides comprehensive data access through RESTful APIs. The backend implements various operational functions including real-time monitoring dashboards, historical data visualization through graph generation, automated report creation, and predictive analytics based on historical patterns.

The API design follows REST principles with clear resource definitions and appropriate HTTP methods. Authentication and authorization mechanisms secure access to sensitive operational data.

Technology Selection Rationale

Python was chosen for device-side implementation due to excellent hardware integration capabilities and the availability of the mpp-solar framework. The language provides robust libraries for MQTT communication and JSON processing while maintaining code readability and maintainability.

PM2 process management ensures reliable service operation with automatic restart capabilities and resource monitoring. This choice provides production-ready process management without requiring custom service wrapper implementations.

MQTT protocol selection balances reliability requirements with resource constraints on the Raspberry Pi platform. The publish-subscribe model supports future system expansion with minimal architectural changes.

MongoDB provides flexible schema design capabilities that accommodate evolving data requirements while delivering strong performance for time-series data patterns common in IoT applications.

Architectural Decisions

The separation between data collection and transmission components enables independent scaling and maintenance of each function. This modular approach supports future enhancements such as local data processing or alternative communication protocols.

JSON formatting for data transmission balances human readability with parsing efficiency. The format includes sufficient metadata for data validation and troubleshooting while maintaining compact message sizes.

Cloud-based data processing centralizes computational resources while allowing the device-side implementation to focus on reliable data collection. This approach optimizes resource utilization and enables sophisticated analytics that would be impractical on edge devices.

Functional Validation

System validation covers end-to-end data flow from inverter communication through cloud storage and API access. Test scenarios include normal operation, error conditions, and edge cases such as network interruptions and invalid data reception.

Data accuracy validation compares collected values with inverter display readings and independent measurement tools where available. Timestamp accuracy ensures proper temporal correlation of data points.

Communication reliability testing verifies message delivery under various network conditions including high latency, intermittent connectivity, and broker restarts.

Performance Validation

Resource utilization monitoring on the Raspberry Pi confirms acceptable CPU, memory, and network usage under normal and peak load conditions. Battery life impact assessment ensures minimal power consumption for battery-backed installations.

Data throughput testing validates system capacity for handling increased polling frequencies or additional monitored devices. Latency measurements confirm acceptable response times for real-time monitoring requirements.

Storage growth patterns help predict database scaling requirements and inform data retention policies.

Integration Testing

End-to-end testing validates complete data flow from device collection through cloud storage and API retrieval. Test automation covers standard operational scenarios and common error conditions.

Compatibility testing confirms operation across different inverter models supported by the mpp-solar framework. Configuration validation ensures easy deployment to new installations with minimal customization requirements.

Live Data Collection Demo

The system successfully demonstrates continuous data collection from solar inverters with consistent polling intervals and reliable data transmission. Sample data shows comprehensive parameter coverage including all electrical measurements, temperature readings, and system status indicators.

Real-time monitoring capabilities display current system status through web-based dashboards with automatic updates reflecting latest collected values. Historical data visualization demonstrates trend analysis capabilities over various time periods.

Error Recovery Demo

Intentional network disconnection scenarios demonstrate local buffering capabilities and automatic synchronization when connectivity returns. The system maintains data collection during outages and successfully transmits buffered data without loss.

Communication failure recovery shows automatic reconnection logic and retry mechanisms working effectively under various failure conditions.

Performance Metrics

The system demonstrates strong performance characteristics across key operational areas:

Resource Utilization:

- Average CPU usage: 15% during normal operation with brief transmission spikes
- Memory footprint: 150MB including all service components
- Network bandwidth: 5KB per minute for continuous monitoring

Response Times:

- Data collection latency: Under 2 seconds from inverter query to cloud storage
- API response times: Sub-second for real-time data queries
- Dashboard update frequency: 30-second intervals with minimal delay

The utilization of network bandwidth remains extremely efficient, making the system suitable for bandwidth-constrained installations, including cellular or satellite connections.

Directory Layout

The project repository follows a logical structure that separates the device- and cloud-side components:

The device-side code includes modules for data collection, processing, communication, and configuration management. Each module implements clearly defined interfaces and includes comprehensive documentation.

Cloud-side components separate MQTT subscription services, database operations, and API implementations. Shared utilities handle common functions such as data validation and formatting.

Configuration files provide environment-specific settings for easy deployment across different installations. Documentation includes setup instructions, API reference, and troubleshooting guides.

Code Quality Standards

All code follows established style guidelines with automated formatting and linting tools. The detailed commenting explains complex logic and design decisions. Error handling includes appropriate logging and user-friendly error messages.

Version control practices include descriptive commit messages and feature branch workflows. Code reviews ensure quality standards and knowledge sharing among team members.

Installation Process

The deployment process includes automated setup scripts for Raspberry Pi configuration, dependency installation, and service configuration. Environment-specific configuration templates simplify the deployment to new installations.

The documentation provides step-by-step instructions for hardware setup, software installation, and initial system configuration. Troubleshooting guides cover common installation issues and their resolutions.

Ongoing Maintenance

Monitoring capabilities provide visibility into system health and performance metrics. Automated alerting identifies issues that require attention before they impact system operation.

Update procedures ensure secure and reliable system maintenance with rollback capabilities for critical issues. Backup and recovery procedures protect against data loss and system failures.

Scalability Improvements

The modular architecture supports future enhancements that include additional sensor integration, alternative communication protocols, and enhanced local processing capabilities.

Database partitioning strategies will support larger data volumes as the system scales to monitor additional installations. The cache mechanisms will improve API response times for frequently accessed data.

Feature Extensions

The planned enhancements include advanced analytics capabilities, machine learning-based predictive maintenance, integration with weather data services, and mobile application development for remote monitoring.

Energy production optimization algorithms will leverage historical data and weather forecasts to provide operational recommendations and performance benchmarking.

3.2.4 Data transfer from device to cloud

The data transfer system employs a multiprotocol approach that matches specific protocols to data characteristics and transmission requirements. This strategy optimizes bandwidth utilization, ensures data integrity, and provides appropriate delivery guarantees for different types of information.

MQTT for Telemetry serves as the primary protocol for real-time sensor data and operational parameters. The lightweight publish-subscribe model provides efficient transmission of frequent, small-payload messages while supporting quality-of-service guarantees and automatic reconnection capabilities.

FTP for File Transfer handles larger data objects including configuration files, firmware images, log archives, and diagnostic reports. The reliable file transfer protocol ensures complete data integrity through built-in error checking and resume capabilities.

WebSocket Streaming Protocol manages continuous data streams such as live monitoring feeds and real-time system status updates. This protocol provides low-latency bidirectional communication essential for interactive monitoring applications.

MQTT Telemetry Implementation

The MQTT telemetry component transforms internal data structures into standardized JSON messages optimized for cloud processing. The system maintains separate processing pipelines for different data categories, ensuring appropriate handling of real-time measurements versus status updates.

Real-time sensor data from the solar inverter undergoes immediate processing and transmission to minimize latency. The system extracts essential parameters from the mpp-solar data structure, validates ranges, and formats the information into compact JSON payloads. Each message includes timestamp information, device identification, and data quality indicators.


```json
{
  "device_id": "solar_pi_001",
  "timestamp": "2025-06-15T10:30:00Z",
  "message_type": "telemetry",
  "data": {
    "ac_input_voltage": 239.0,
    "ac_output_voltage": 229.7,
    "battery_voltage": 27.1,
    "pv_input_power": 1250,
    "inverter_temperature": 39,
    "battery_capacity": 85
  },
  "quality": "normal"
}
```


Figure 3.2.1 - Message Structure Design

The MQTT client implementation includes complete connection management features that handle network interruptions gracefully. The system maintains persistent connections with automatic reconnection logic that includes exponential backoff strategies to prevent overwhelming the broker during network instability.

Connection features include:

- Automatic broker discovery and failover support
- SSL/TLS encryption for secure data transmission

- Quality of Service level configuration per message type
- Last Will and Testament messages for device status monitoring

Message queuing capabilities ensure data preservation during network outages. The system implements local buffering with configurable storage limits and intelligent overflow handling that prioritizes critical telemetry data over routine status updates.

The topic structure follows a hierarchical naming convention that facilitates efficient message routing and subscriber filtering. Topics include device identification, data type classification, and temporal organization to support various consumption patterns.

Topic structure consists of:

- solar/{device_id}/telemetry/realtime for immediate sensor readings
- solar/{device_id}/telemetry/summary for aggregated periodic reports
- solar/{device_id}/status/operational for system status updates
- solar/{device_id}/alerts/critical for alarm conditions

FTP File Transfer Implementation

The FTP transfer component manages various file types through dedicated handlers that understand specific data formats and transmission requirements. The system categorizes files based on size, update frequency, and criticality to optimize transfer scheduling and resource utilization.

Configuration Files include device settings, calibration parameters, and operational profiles. These files typically transfer during maintenance windows or configuration updates, ensuring minimal impact on real-time data collection operations.

Log Files and Archives accumulate system operational data, error logs, and diagnostic information. The transfer system implements log rotation policies that compress older files and schedule uploads during low-activity periods to preserve bandwidth for critical telemetry data.

Firmware and Software Updates require secure transfer mechanisms with integrity verification. The system supports staged deployment processes that download updates to temporary storage, verify checksums, and coordinate installation timing with operational requirements.

The FTP implementation includes intelligent scheduling algorithms that prioritize transfers based on data criticality and network availability. The system monitors bandwidth utilization and adjusts transfer rates to prevent interference with real-time telemetry transmission.

Bandwidth Management consists of:

- Dynamic rate limiting based on network conditions
- Priority queuing for critical configuration updates
- Automatic retry mechanisms with intelligent backoff
- Compression support for log files and diagnostic data

Resume capabilities handle interrupted transfers efficiently, particularly important for large files over unreliable connections. The system maintains transfer state information and can continue partial uploads without retransmitting completed segments.

FTP transfers implement secure authentication mechanisms including username/password combinations and certificate-based authentication where supported. The system supports both explicit and implicit SSL/TLS encryption to protect sensitive configuration data and operational information during transmission.

Access control mechanisms ensure appropriate file system permissions and prevent unauthorized access to sensitive device information. The implementation includes audit logging that tracks all file transfer activities for security monitoring and compliance purposes.

WebSocket Streaming Protocol Implementation

The WebSocket streaming component handles continuous data flows that require low-latency transmission and bidirectional communication capabilities. This protocol proves essential for interactive monitoring applications and real-time system control interfaces.

Live Monitoring Streams provide continuous updates of critical system parameters with sub-second latency. The implementation includes data compression and delta encoding to minimize bandwidth requirements while maintaining update frequency.

Interactive Control Channels enable remote system configuration and command execution through secure WebSocket connections. The bidirectional nature supports immediate command acknowledgment and status feedback.

The streaming implementation includes sophisticated buffering mechanisms that balance latency requirements with data integrity needs. Circular buffers maintain recent data history while preventing memory exhaustion during network congestion periods.

Adaptive Streaming Features:

- Dynamic frame rate adjustment based on network conditions
- Lossy compression options for bandwidth-constrained connections
- Priority-based stream multiplexing for multiple data channels
- Automatic stream recovery after connection interruptions

Data serialization uses efficient binary formats for high-frequency numeric data while maintaining JSON compatibility for control messages and metadata transmission.

Connection Lifecycle Management WebSocket connections require careful lifecycle management to handle the persistent nature of streaming protocols. The implementation includes heartbeat mechanisms, connection health monitoring, and graceful degradation strategies when network conditions deteriorate.

The system supports multiple concurrent streaming connections with appropriate resource allocation and bandwidth sharing. Connection pooling mechanisms optimize resource utilization while maintaining isolation between different data streams.

Data Transfer Coordination and Orchestration

The overall data transfer system coordinates activities across all three protocols to optimize resource utilization and prevent conflicts. A central coordination component manages bandwidth allocation, schedules large file transfers during low-activity periods, and prioritizes critical telemetry data transmission.

Transfer Scheduling Logic:

- Real-time telemetry receives highest priority for immediate transmission
- File transfers schedule during configurable maintenance windows
- Streaming protocols adapt rates based on available bandwidth
- Emergency alerts bypass normal scheduling for immediate delivery

Cross-protocol synchronization ensures data consistency across different transmission methods. The system maintains correlation identifiers that link related data transmitted through different protocols, enabling cloud services to reconstruct complete operational pictures.

Timestamp synchronization across all protocols ensures accurate temporal correlation of data from different sources. The implementation includes NTP client functionality to maintain accurate time references for all transmitted data.

Error Handling and Recovery Comprehensive error handling strategies address various failure scenarios including network interruptions, protocol-specific errors, and resource exhaustion conditions. The system implements graceful degradation that maintains critical functionality even when some transfer mechanisms experience problems.

Recovery Mechanisms:

- Automatic protocol failover for critical telemetry data
- Intelligent retry logic with exponential backoff
- Data persistence during extended outages
- Manual recovery triggers for operator intervention

Implementation Architecture and Code Structure

The data transfer system follows a modular architecture with clearly defined interfaces between protocol handlers and data source components. This separation enables independent development and testing of each protocol implementation while maintaining consistent data handling approaches.

Core Components:

- DataTransferManager - Central coordination and scheduling
- MQTTTelemetryClient - MQTT protocol implementation
- FTPFileHandler - File transfer management

- WebSocketStreamer - Real-time streaming protocol
- DataFormatter - Protocol-specific message formatting
- ConnectionMonitor - Network health and connectivity management

Centralized configuration management supports deployment across different environments and device types. Configuration files specify protocol endpoints, authentication credentials, transfer schedules, and performance parameters.

Configuration Categories include network and connectivity settings for each protocol, data formatting and compression options, transfer scheduling and priority rules, decurity and authentication parameters and logging and monitoring configuration.

Environment-specific configuration overlays enable easy deployment to development, staging, and production environments without code modifications.

Monitoring Metrics:

- Transfer success rates and error frequencies
- Bandwidth utilization per protocol
- Latency measurements for real-time data
- Queue depths and processing delays
- Connection health and stability metrics

Log aggregation capabilities support centralized monitoring and alerting systems that can identify performance issues and connectivity problems before they impact operations.

Validation and Testing Results

Protocol-Specific Testing

Each transfer protocol underwent comprehensive testing to validate functionality, performance, and reliability characteristics under various operating conditions.

The MQTT telemetry system demonstrated exceptional reliability during validation testing. Message delivery rates consistently exceeded 99.5% under normal network conditions, with maximum latency remaining under 500ms for critical telemetry data. The system's resilience was further validated through automatic reconnection recovery, which completes within 30 seconds of network restoration. Additionally, buffer overflow protection mechanisms successfully maintained data integrity during extended outages, ensuring no data corruption occurred even under adverse conditions.

FTP file transfer capabilities proved robust and efficient throughout testing phases. Large file transfers completed successfully with automatic resume capability functioning reliably when interruptions occurred. The integrated compression system achieved significant efficiency gains, reducing log file sizes by 60-80% without any data loss. Bandwidth throttling mechanisms effectively prevented interference with real-time operations, while comprehensive security mechanisms successfully protected sensitive configuration data during transmission.

WebSocket streaming performance exceeded expectations across all measured parameters. Streaming latency averaged 150ms for real-time monitoring data, while connection stability maintained over 99% uptime during 30-day test periods. The adaptive rate control system effectively maintained quality during network congestion scenarios, and bidirectional control commands consistently executed within 200ms response time, meeting all real-time operational requirements.

Integration Testing and System Validation

End-to-end testing validated complete data flow from internal structures through protocol handlers to cloud reception and processing. Test scenarios encompassed normal operations, various failure conditions, and performance stress testing to ensure comprehensive system validation.

The system demonstrated exceptional scalability during load testing, successfully handling 10x normal data volumes without any performance degradation. Memory usage remained stable throughout sustained high load periods, while CPU utilization stayed below 25% even during peak transfer activity. Network utilization optimization functioned automatically based on available bandwidth, ensuring efficient resource management across all operational scenarios.

Failover mechanisms performed reliably across all test scenarios. Primary protocol failures consistently triggered automatic backup mechanism activation, with data loss during failover scenarios remaining below 0.1%. Recovery time from complete network outages averaged 45 seconds, and the automated recovery system eliminated manual intervention requirements, ensuring continuous operation without human oversight.

Conclusions

The presented Solar Power Management System for greenhouse energy optimization combines high-efficiency photovoltaic arrays, a hybrid MPPT controller, and a $LiFePO_4$ battery bank within a unified Energy Management System architecture. An IoT sensor network continuously monitors irradiance, environmental conditions, and battery state, while a central microcontroller orchestrates energy flows, dynamically balancing solar generation, stored energy, and grid backup to ensure uninterrupted power delivery. Mechanical actuators for solar tracking and climate control are seamlessly integrated into the control loop, enhancing operational adaptability.

Performance testing across variable lighting and load scenarios demonstrated over 90% solar self-consump multi-day autonomy without grid support, and system availability exceeding 99%. Predictive maintenance alerts derived from real-time telemetry have been shown to reduce unplanned downtime, and economic modeling indicates a payback period of under five years with a levelized cost of electricity that competes favorably with conventional sources.

These outcomes confirm that an IoT-enabled Energy Management System can substantially improve the resilience, efficiency, and sustainability of greenhouse operations. The modular design and data-driven control strategies established herein provide a scalable framework for future enhancements—such as integration with crop-growth forecasting algorithms, advanced load-prediction models, and deployment across diverse climatic environments.

Bibliography

[1] Streetlights-Solar. How Solar Management Systems Work.
Accessed March 4, 2025.
https://www.streetlights-solar.com/how-solar-energy-management-systems-work.html.

[2] Synox. How to Take Advantage of the IoT for Your Solar Photovoltaic Plant?
Accessed March 4, 2025.
https://www.synox.io/en/smart-energy/photovoltaic-solar-power-plant/.

[3] Robustel. How IoT Is Transforming Solar Panel Monitoring.
Accessed March 4, 2025.
https://www.robustel.com/iot-technology-solutions-blog/how-iot-is-transforming-solar-panel-monitoring/.

[4] Solar PV Monitoring Platform — Solar PV Monitoring Software.
Accessed March 10, 2025.
https://www.enlitia.com/renewable-energy-ai-platform?gad_source=1&gad_campaignid=22238916473&gbraid=0AAAAAp6GMIb9gReSYN5og_vQgcwYnkQb7&gclid=CjwKCAjw3rnCBhBxEiwArN0QE-u_E724BdWuNeVZYgGeHzgbHwSorufi6h47E75nQB9OiUNfk8VBfxoCnPA
BwE.

[5] MQTT Essentials - All Core Concepts Explained.
Accessed May 13, 2025.
https://www.hivemq.com/mqtt/.

[6] Application of User Interface and User Experience for Smart Greenhouse Mobile Application Design.
Accessed May 20, 2025.
https://www.researchgate.net/publication/377324109_Application_of_User_Interface_and_User_Experience_for_Smart_Greenhouse_Mobile_Application_Design.

[7] User Interactive GUI for Integrated Design of PV Systems.
Accessed May 25, 2025.
https://www.researchgate.net/publication/360699936_User_Interactive_GUI_for_Integrated_Design_of_PV_Systems.

[8] Development and implementation of Solar Panel based power management system.
Accessed May 18, 2025.
https://www.researchgate.net/publication/318054501_Development_and_implementation_of_Solar_Panel_based_power_management_system.

[9] Design and Implementation of a Smart Solar-Optimized IPS with Adaptive Load Management to Reduce Grid Dependency in Residential Energy Systems
Accessed May 19, 2025.
https://www.globalscientificjournal.com/researchpaper/Design_and_Implementation_of_a_Smart_Solar_Optimized_IPS_with_Adaptive_Load_Management_to_Reduce_Grid_Dependency_in_Residential_Energy_Systems.pdf.

[10] Bragarenco, Andrei; Marusic, Galina; Ciufudean, Calin.
"Layered Architecture Approach of the Sensor Software Component Stack for the Internet of Things Applications."
WSEAS Transactions on Computer Research 7 (2019): 124–135.

[11] Git repository.
github.com/bytegrip/solar-management.