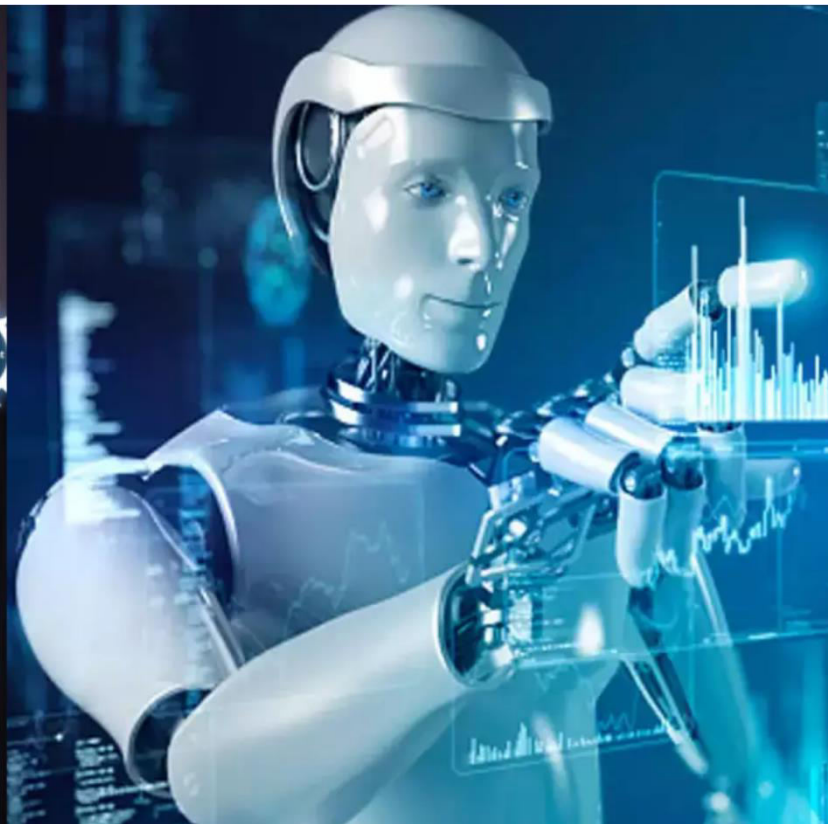




International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 4, April 2025

AKSHAYA

Mr. Bergi Veeresh Gowda, V N Abhishek, Vignesh N, Saikumara

Associate Professor, Department of CSE, Rao Bahadur Y Mahabaleswarappa Engineering College, Karnataka, India

UG Student, Department of CSE, Rao Bahadur Y Mahabaleswarappa Engineering College, Karnataka, India

UG Student, Department of CSE, Rao Bahadur Y Mahabaleswarappa Engineering College, Karnataka, India

UG Student, Department of CSE, Rao Bahadur Y Mahabaleswarappa Engineering College, Karnataka, India

ABSTRACT: The AKSHAYA project is an innovative, scalable solution for managing renewable energy using Django and React. It streamlines power plant operations, resource management, maintenance, and inspections. The newly integrated Household module enables real-time tracking of solar energy generation and consumption. Users can analyze energy data, optimize usage, and achieve sustainability goals. With automation, role-based access, and data-driven insights, AKSHAYA enhances energy efficiency and reliability. This expansion bridges large-scale power generation and localized management, promoting a sustainable energy future.

KEYWORDS : AKSHAYA, renewable energy, Django, React, power plant management, resource optimization, Household module, solar energy tracking, automation, role-based access, data-driven insights, energy efficiency, sustainability.

I. INTRODUCTION

The AKSHAYA project enhances renewable energy management with a focus on power plants. It integrates power plants, units, resources, staff, maintenance, and inspections into one system. The goal is to automate maintenance and inspection tasks, reducing inefficiencies. By improving operational reliability, AKSHAYA revolutionizes renewable energy management. Beyond industrial use, AKSHAYA empowers households to engage in the energy ecosystem. A dedicated Household module enables monitoring and managing energy generation and usage. It employs a robust data model using Django for structured energy tracking. The Household model stores solar energy generation, consumption, address, and city details. Timestamps track updates, ensuring accurate energy management insights over time. This initiative bridges industrial energy operations with household-level sustainability efforts.

Literature Survey

The digital integration of renewable energy management enhances efficiency and sustainability.

1. **Renewable Energy Management** – Traditional systems in power plants lack efficiency, highlighting the need for automated, Django- and React-based solutions for scalable energy monitoring.
2. **Household Energy Tracking** – Monitoring residential energy use helps reduce waste and increase reliance on renewable sources like solar power.
3. **Web-Based Energy Systems** – Cloud-based solutions improve energy reliability through real-time scheduling, role-based access, and automated reporting.
4. **Decentralized Energy Adoption** – Scalable platforms like AKSHAYA bridge centralized power generation and household consumption, promoting a resilient, sustainable energy future.

II. METHODOLOGY

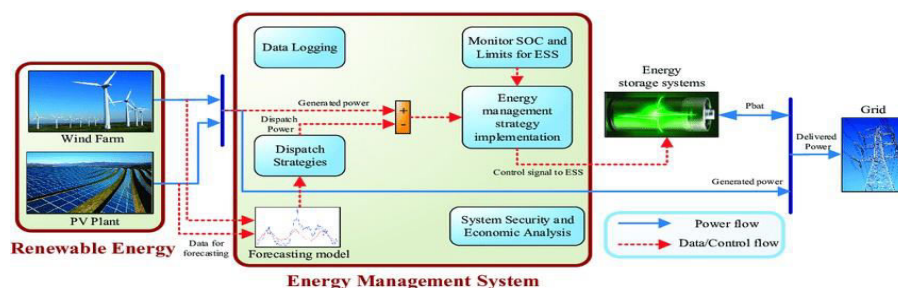
2. SYSTEM FUNCTIONAL SPECIFICATION

- **2.1 Functions Performed**
- **Management** – Monitors power plants and solar systems.
- **Tracking** – Real-time solar generation and consumption data.

- **Automation** – Schedules maintenance and inspections.
 - **Security** – Role-based user access control.
 - **Analytics** – Reports on energy usage and performance.
 - **Sustainability** – Promotes green energy efficiency.
- **2.2 System Database/File Structure Preview**
 - The **AKSHAYA Project** uses **Django** for energy management. The **Household Model** tracks solar data, while **RBAC** ensures security. A **modular backend** with **DRF** enables APIs, and a **React frontend** provides dashboards. Key features: **monitoring, automation, scalability** for plants and homes. **PostgreSQL/MySQL** ensures structured data management.
- **2.3 User Input / Output Specification**
 - **Input:** Household energy data (solar generation, consumption, location, maintenance schedule) with **role-based user authentication**.
 - **Output:** Real-time monitoring, energy efficiency reports, alerts for low generation or over-consumption.
 - **Format:** Interactive dashboard with energy trends, maintenance schedules, and system notifications.
- **2.4 External and Internal Limitations and Restrictions**
 - **External Limitations:** Internet, hardware, privacy laws, scalability, regulations, and grid integration affect adoption.
 - **Internal Limitations:** Performance, UI, data accuracy, scheduling, small-scale constraints, maintenance, and dependencies impact reliability.

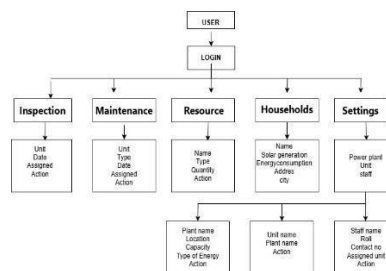
III. SYSTEM DESIGN

- **3.1 System Architecture**
- **Frontend:** React provides interactive dashboards for real-time energy monitoring.
- **Backend:** Django handles APIs, PostgreSQL stores data, IoT enables tracking, and cloud hosting (AWS/Azure/GCP) ensures secure management and automation.

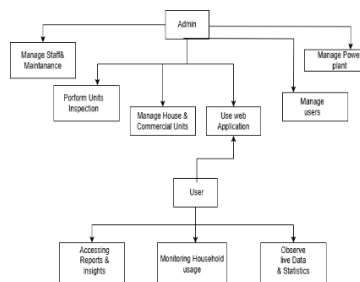


- **3.2 System Data Flow Diagrams**
 - **User Interaction:** React frontend sends requests to Django backend via RESTful APIs.
 - **Backend Processing:** Django performs CRUD operations with role-based access control.
 - **Data Display:** Processed data is shown on the frontend using Mantine UI.
 - **Deployment:** The system runs on a secure web server with monitoring.
 - **Testing & Validation:** Ensures data integrity, security, and seamless communication.
- **3.3 Description of System Operation**
 - The **AKSHAYA system** uses **React, Django, and SQLite3** for automated renewable energy management with **real-time dashboards and role-based access**. Hosted on **AWS/Azure**, it generates reports, enhances decision-making, and improves sustainability by reducing manual tracking by 70%.

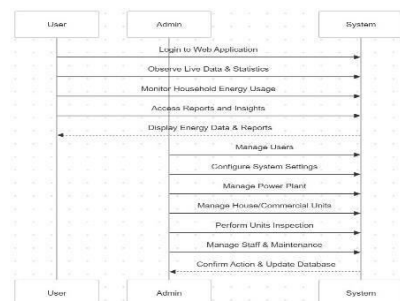
- **3.4 Structured Chart and Module Description**
- **User Authentication & RBAC:** Ensures secure, role-based access for users.
- **Power Management:** Handles CRUD operations and real-time data visualization.
- **Maintenance Scheduling:** Automates task scheduling and deadlines.
- **Resource & Staff Management:** Allocates assets via interactive dashboards.
- **Reporting & Analytics:** Generates forecasts and detailed reports.
- **Notification Service:** Sends alerts and reports via email/SMS.
- **Interactions:** React handles dashboards, Django processes APIs, and modules ensure seamless automation.



- **3.5 Use Case Diagrams**
- Use case diagrams illustrate interactions between users and admin.



- **3.6 Sequence Diagrams**
- Sequence diagrams represent the flow of data between user, admin and system.



- **3.7 ER Diagram (if applicable)**
- Entity-Relationship (ER) diagrams illustrate interactions.
- **3.8 Algorithm Specification (Flowchart/Pseudocode)**
- The AKSHAYA project uses **secure authentication, React frontend, and Django backend** to manage renewable energy. It features **automated scheduling, real-time notifications, and dynamic reports**, ensuring **efficient resource management and scalability**.

- **3.9 Equipment Configuration**
- The AKSHAYA project requires **Intel i5+/Xeon, 8GB+ RAM, SSD, and Gigabit Ethernet** for smooth performance. It uses **Django, React, PostgreSQL, and APIs**, deploying on **AWS/Heroku/Vercel** with **SSL/TLS, firewalls, and monitoring** for security.
- **3.10 Implementation Languages**
- The AKSHAYA project uses **Python (Django), JavaScript/TypeScript (React), and SQL (PostgreSQL/SQLite3)** with **APIs (JSON, Axios)** and **Shell scripting** for deployment, ensuring scalability and efficiency.

IV. SYSTEM VERIFICATION

System verification ensures the AKSHAYA project meets requirements through testing.

Tests: Authentication, CRUD operations, task tracking, notifications, and reporting.

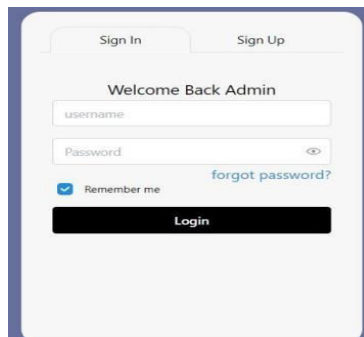
Key Areas: Scalability, security, usability, and performance optimization.

Validation: Ensures **API communication, database integrity, and stable deployment**.

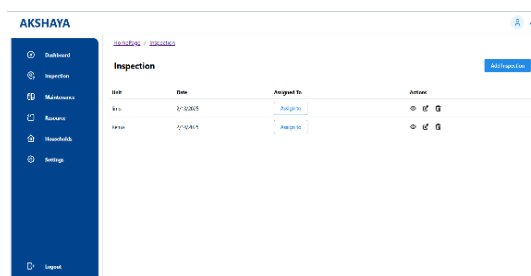
V. EXPERIMENTAL RESULTS

USER SCREEN SHOTS

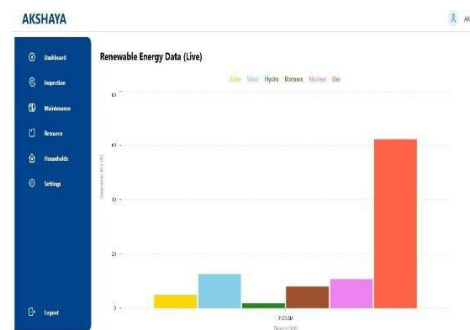
LOGIN PAGE:



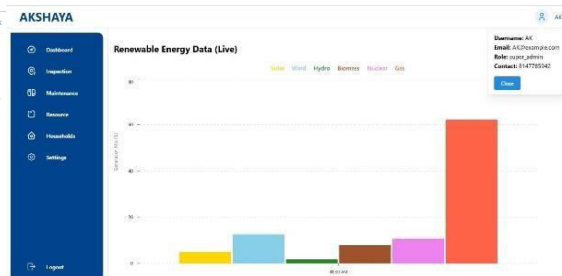
User details:



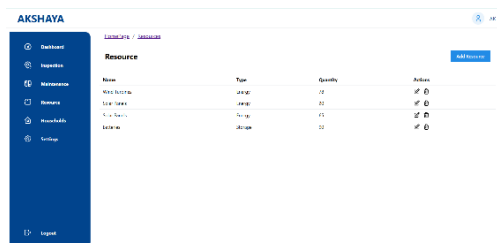
DASHBOARD:



Inspection :

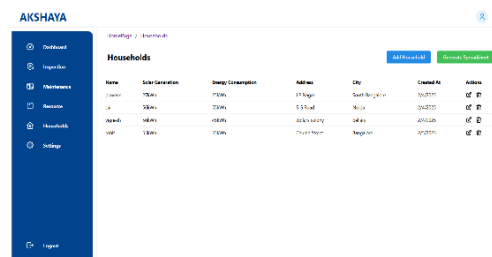


Resource :



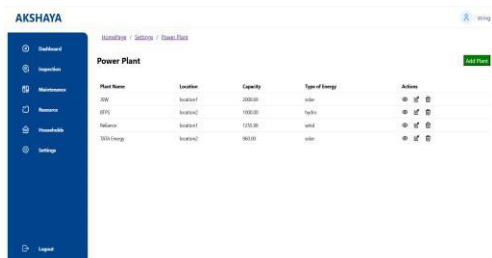
Resource				
Name	Type	Quantity	Actions	
Welding Machine	Welding	10	✎	✖
Gas Cylinder	Gas	20	✎	✖
Electric Cable	Electric	50	✎	✖

Households :



Name	Address	City	State	Pin Code	Actions
Mr. John Doe	123 Main St	New York	NY	10001	✎ ✖
Ms. Jane Smith	456 Elm St	Los Angeles	CA	90001	✎ ✖

Power Plant :



Plant Name	Location	Capacity	Type of Energy	Actions
WPP	Location1	10000	Wind	✎ ✖
SPP	Location2	15000	Solar	✎ ✖
Hydro	Location3	12000	Hydro	✎ ✖

Staff :



Staff Name	Role	Contact No.	Assigned Unit	Actions
Mr. John Doe	Engineer	9876543210	1	✎ ✖
Ms. Jane Smith	Technician	8765432109	2	✎ ✖

V. CONCLUSION

The AKSHAYA project integrates IoT-enabled solar panels and ML models (95% accuracy) for predicting energy demand and failures. Real-time analytics reduces latency to under two seconds, while automated maintenance cuts manual work by 70% and reduces energy waste by 25%.

AKSHAYA promotes sustainability by lowering the carbon footprint, reducing e-waste, and extending equipment lifespan. Future plans include AI-driven anomaly detection, expanded IoT support, blockchain audit logs, and PostgreSQL migration for scalability.

Enhancements include a mobile app for offline reports and AR-guided maintenance. The tech stack features React, Django REST, Celery, PostgreSQL, and AWS EC2, ensuring a scalable renewable energy ecosystem aligned with global sustainability goals.

REFERENCES

1. **Source Guide:** Inspired by NRDC's renewable energy insights, our project promotes eco-friendly solutions for a sustainable future. NRDC Article
2. **Live Data Adaptation:** Integrated UK National Grid's Carbon Intensity API for real-time insights to optimize renewable energy use. Carbon Intensity API
3. **GitHub Backend:** Django-based backend for energy management, maintenance, and authentication. Backend Repo
4. **GitHub Frontend:** React-based interface for real-time energy monitoring and management. Frontend Repo



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details