WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

SmartGrid AI: An AI-Based Smart Electricity Theft Detection and Real-Time Monitoring System

HARISH K[1], JAISANTHOSH S[2], KOVARSHINI A N[3], LATHIKA G[4], MADHUMITHA R K[5]

Department of Computer Science and Engineering Sri Shakthi Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India

ABSTRACT :

Electricity theft and unauthorized power consumption are significant challenges for electricity distribution systems. Conventional monitoring often relies on periodic meter readings and manual inspection, which can delay the identification of unusual consumption patterns. This paper presents SmartGrid AI, a web-based platform designed to support electricity consumption monitoring and the identification of potentially suspicious usage. The platform combines consumer and grid-location management, smart-meter asset records, consumption-data handling,

machine-learning-based anomaly detection, live telemetry visualization, alert management, investigation tracking, and report generation. The anomaly detection component uses the Isolation Forest algorithm to identify records that differ from learned consumption patterns. A software-based meter simulator generates sample readings and demonstrates continuous data ingestion and live updates through WebSocket communication. The application is implemented using React, Vite, Tailwind CSS, Python, FastAPI, Scikit-learn, SQLite, and WebSocket. The system is intended as a decision-support tool: an anomaly indicator is not proof of theft and requires examination by authorized personnel. The current demonstration uses simulated meter data; connection to physical meters requires an appropriate meter or IoT gateway integration.

Index Terms—Electricity theft detection, smart grid, Isolation Forest, anomaly detection, machine learning, smart meter simulation, real-time monitoring.

INTRODUCTION

Electricity is essential to residential, commercial, and industrial activity. Distribution organizations must monitor consumption, maintain service reliability, and identify irregularities that may indicate equipment faults, data-quality problems, unauthorized connections, or electricity theft. Conventional monitoring processes commonly use periodic meter readings, manual inspection, and comparisons against expected consumption. These approaches remain useful, but they may not provide timely visibility when large numbers of consumers and meters must be monitored.

A sudden decrease in recorded consumption, an unexpected zero reading, or a substantial difference from a consumer's reference pattern may warrant further investigation. However, an unusual reading can also arise from legitimate changes in occupancy, seasonal demand, meter malfunction, communication interruption, or incorrect data entry. A monitoring platform should therefore identify records that deserve attention without automatically treating every anomaly as confirmed theft.

Artificial intelligence and machine learning offer methods for examining consumption records and identifying patterns that differ from normal behavior. Anomaly detection is particularly relevant when confirmed examples of theft are limited or unavailable. SmartGrid AI applies this approach within a centralized web platform. It combines data administration, consumption analysis, real-time visualization, alert handling, and investigation records so that monitoring personnel can review suspicious activity in a structured workflow.

The main contribution of this work is the integration of these operational functions in one demonstrator. The platform includes a software-based telemetry simulator to show the flow of meter readings and the response of the monitoring interface. The paper describes the system design, processing workflow, implemented modules, technology stack, and practical limitations. It does not claim field validation using physical smart meters or provide numerical accuracy claims that have not been measured.

PROBLEM STATEMENT AND OBJECTIVES

Electricity consumption information is often distributed across meter records, consumer details, location data, and inspection notes. When these records are reviewed separately, personnel may spend additional time identifying the relevant consumer, understanding the meter context, and recording follow-up actions. Periodic inspection can also delay the discovery of unusual patterns between scheduled readings. The problem addressed by SmartGrid AI is the lack of a unified monitoring workflow that connects consumption records, anomaly indicators, live visibility, and investigation management.

The platform is designed to address the following operational challenges:

- Identifying unusual consumption patterns that may be difficult to notice through manual review alone.
- Reducing the delay between the arrival of a reading and its visibility to monitoring personnel.

- Maintaining consumer, grid-location, and meter information in a centralized application.
- Organizing potentially suspicious records as alerts that can be reviewed and tracked.
- Maintaining investigation details and generating summaries for monitoring and analysis.
- Providing controlled access to application functions and retaining human verification of AI-generated indicators.

A. Objectives

- Develop a centralized web-based electricity monitoring platform.
- Apply the Isolation Forest algorithm to selected consumption-related features.
- Display incoming simulated meter readings through a live monitoring interface.
- Manage consumers, grid locations, smart-meter assets, and consumption records.
- Generate suspicious-activity indicators and support alert review.
- Maintain investigation records, reports, notifications, and audit information.
- Use role-based access to restrict application functions to authorized users.

RELATED WORK AND BACKGROUND

A. *Conventional Electricity Monitoring*

Traditional electricity monitoring relies on meter readings, billing information, and field inspections. Electricity theft detection has been studied using different data-driven and meter-based approaches [3]. These sources provide essential operational data, but manual comparisons can become time-consuming as the number of consumers increases. Periodic readings also provide only a snapshot of consumption and may not reveal a sudden change until the next reading cycle. Automated data collection and centralized dashboards can improve visibility, while anomaly detection can help prioritize records for review.

B. *Smart Grid Technologies*

Smart-grid environments use digital measurement, communication networks, and data-processing systems to improve the observation and management of electricity distribution [2]. Smart meters can provide more frequent consumption information than conventional manual reading processes. The usefulness of such information depends on reliable communication, data quality, appropriate access controls, and clear procedures for responding to unusual readings. SmartGrid AI models the software side of this workflow and demonstrates the handling of incoming telemetry.

C. *Machine Learning for Anomaly Detection*

Machine-learning methods can identify consumption records that differ from historical or expected patterns. Supervised classification generally requires labelled examples, whereas unsupervised anomaly detection can be applied when reliable labels are scarce. Anomaly detection does not determine the cause of an unusual reading; it highlights observations that may require additional context. This distinction is important in electricity monitoring because legitimate operational changes and data errors can resemble suspicious behavior.

D. *Isolation Forest*

Isolation Forest is an unsupervised algorithm that isolates observations through randomly selected features and split values [1]. Observations that are different from the majority can often be isolated using fewer partitioning steps. The algorithm is suitable for anomaly-screening tasks because it does not require a labelled theft example for every pattern. In SmartGrid AI, it is applied to selected consumption-related features and produces an anomaly indicator for review. The output depends on data preparation, selected features, model configuration, and the characteristics of the data being analyzed.

E. *Research Gap*

A practical monitoring application requires more than an isolated machine-learning model. It must connect data ingestion, consumer and meter context, anomaly screening, live visualization, alert review, and investigation tracking. SmartGrid AI addresses this integration requirement by placing these functions in one application. The current implementation is a software demonstrator; its effectiveness in a live distribution environment would require field data, operational validation, and evaluation against confirmed inspection outcomes.

PROPOSED SYSTEM

SmartGrid AI is a centralized web application designed to support electricity consumption monitoring and the identification of potentially abnormal usage patterns. The frontend provides interfaces for administration and monitoring. Backend services handle application requests, data validation, persistence, and machine-learning operations. Consumption records are processed to prepare features for anomaly detection. When a record is considered unusual, the system makes the resulting indicator available to the alert and review workflow.

The platform supports consumer management, grid-location management, smart-meter asset management, consumption-record handling, AI-based anomaly detection, simulated real-time monitoring, suspicious alerts, investigation management, reporting, notifications, audit logs, and system settings. The software-based simulator is used to demonstrate a continuous stream of readings. It should not be interpreted as a connection to physical

utility meters. Physical deployment would require authorized access to meter hardware or an IoT gateway, secure communication, and integration with the relevant utility data systems.

A. Design Principles

- Centralized context: consumer, meter, location, and reading information are associated to support review.
- Modular implementation: frontend, backend, database, and machine-learning responsibilities are separated.
- Human-in-the-loop operation: alerts support prioritization and do not automatically establish wrongdoing.
- Traceability: investigation and audit records help document follow-up activity.
- Demonstration transparency: simulated readings are distinguished from measurements obtained from physical equipment.

SYSTEM ARCHITECTURE

The architecture consists of presentation, application, data-processing, AI-detection, database, real-time communication, and human-review components. These components exchange information through backend services and defined application interfaces. The architecture separates the user-facing monitoring experience from the processing and persistence functions so that each responsibility can be maintained independently.

A. *Presentation Layer*

The presentation layer is implemented with React and Vite. It provides the dashboard and module interfaces through which authorized users access consumer details, grid locations, meter assets, consumption records, alerts, investigations, and reports. Tailwind CSS is used for consistent interface styling. Dashboard cards and charts summarize the information available from backend services.

B. *Application and API Layer*

The backend is implemented using Python and FastAPI. It exposes API endpoints for application operations such as authentication, consumer management, location management, meter management, consumption records, anomaly detection, alerts, investigations, and reports. The API layer validates incoming requests and coordinates interactions between the frontend, database, and detection component.

C. *Data Processing and AI Layer*

The data-processing component checks incoming consumption information and prepares selected features for the model. The AI component uses Scikit-learn to apply Isolation Forest to the prepared records. The output is represented as an anomaly indicator and passed to the application workflow. It is important that the model output remains interpretable as a screening result rather than a final determination of electricity theft.

D. *Database Layer*

SQLite is used for persistent storage in the current implementation. It maintains application information relating to consumers, grid locations, smart-meter assets, consumption readings, alerts, investigations, and audit activities. The database provides a central source of application records for the frontend and backend services. A production deployment may require a database configuration selected according to expected data volume, availability, backup, and concurrency requirements.

E. *Real-Time Communication Layer*

A software-based meter simulator generates sample readings at configurable intervals. The backend receives the readings through an ingestion interface and distributes live updates through WebSocket communication. The monitoring interface can therefore display incoming telemetry without requiring the user to refresh the page for every update. The simulator supports demonstration scenarios, including normal readings and selected abnormal patterns.

F. *Human Review and Authorization Layer*

The human-review layer presents suspicious records to authorized users. Personnel can inspect the associated consumer and meter context, review the alert, and maintain investigation information. Role-based access is used to restrict application functions. This review stage is necessary because an anomaly can have several causes, including legitimate consumption changes, meter faults, or data transmission problems.

METHODOLOGY

The SmartGrid AI workflow follows a sequence of data collection, validation, preprocessing, feature preparation, anomaly detection, alert generation, and investigation. The steps are designed to connect the incoming data with the operational modules that support review.

A. *Data Collection*

Consumption information is obtained from stored records and the software-based smart-meter simulator. The simulator generates sample electrical parameters such as active power, voltage, current, power factor, and consumption values. Readings are associated with meter and consumer information where those records are available. In the current demonstration, generated values are used to test the ingestion and visualization workflow.

B. *Data Preprocessing*

Before analysis, the collected readings are checked for the presence of required values and prepared for feature extraction. Preprocessing is important because missing, invalid, or inconsistent values can affect anomaly detection. The application should distinguish an invalid or incomplete reading from a valid but unusual consumption pattern. Where required information is unavailable, the record should be flagged for data-quality review rather than treated as confirmed suspicious activity.

C. *Feature Engineering*

The system uses selected features to represent electricity consumption behavior. These features provide different views of the current reading and its relationship to a reference value:

- Units consumed: the recorded electricity consumption, expressed in kilowatt-hours where the source data uses that unit.
- Consumption ratio: the relationship between current consumption and a corresponding reference consumption.
- Z-score: the deviation of a consumption value relative to a statistical reference distribution.
- Load utilization: the proportion of available load capacity being used, where capacity information is available.
- Drop percentage: the percentage decrease relative to a selected reference consumption value.
- Zero-reading indicator: a flag identifying a reading in which the recorded consumption is zero.

The usefulness of each feature depends on consistent units, a meaningful reference period, and the quality of the source data. For example, a drop percentage is only meaningful when the current and reference values are comparable. Features should therefore be calculated using documented rules and checked for division-by-zero or missing-reference conditions.

D. *Isolation Forest-Based Anomaly Detection*

Isolation Forest builds randomized partitioning trees and examines how easily observations can be isolated. An observation that differs substantially from the majority may require fewer splits to become isolated. The algorithm uses path length as part of its anomaly-scoring process. Conceptually, the anomaly score can be expressed as:

$$s(x, n) = 2^{-E(h(x))/c(n)}$$

where x represents an observation, n represents the sample size, E(h(x)) is the average path length required to isolate the observation, and c(n) is the average path length of an unsuccessful search in a binary search tree. The model output is used to identify records for further examination. The actual detection threshold and model parameters should be documented when experimental evaluation is performed.

E. *Alert Generation and Investigation*

When the detection component identifies a potentially unusual record, the application makes an alert available to authorized users. The alert should retain enough context to support review, such as the relevant meter, consumer, reading time, and anomaly information available from the system. Personnel can examine the record and maintain investigation details. The investigation workflow helps distinguish an alert that is awaiting review from one that has been examined or requires additional action.

F. *Real-Time Monitoring and Reporting*

The simulator produces sample readings at configurable intervals. The backend ingests the readings and publishes updates through WebSocket communication. The live monitoring interface displays current telemetry and a stream of incoming records. Reporting functions summarize consumption information, suspicious records, and investigation activity to support monitoring and analysis. Since the demonstration data is simulated, reports generated from it should be understood as application test outputs rather than utility billing or field-inspection evidence.

G. *End-to-End Monitoring Workflow*

A typical monitoring workflow begins when a user signs in and selects the relevant monitoring area. Consumer, location, and meter records provide the context needed to interpret incoming readings. A reading may enter the platform through stored consumption data or the software simulator. The backend validates the record, prepares the selected consumption features, and passes the prepared values to the anomaly detection component. If the model identifies an unusual pattern, the platform makes the associated record available through the suspicious-alert interface. An authorized user can inspect the available context, record the review outcome, and create or update an investigation entry when follow-up is required.

This workflow connects automated screening with a documented human review process. It also makes clear where further information may be needed: an unusual reading may require a check of the meter, communication link, reference consumption, or consumer circumstances. The system can organize

the information available to the user, but it cannot independently establish the cause of the anomaly. This distinction is maintained throughout the application so that AI output is used as decision support rather than an automatic enforcement decision.

FUNCTIONAL MODULES

The application is divided into functional modules so that user tasks can be performed through a consistent interface. The principal modules are summarized below.

Module	Function
User authentication	Provides the login interface and restricts access to authorized application functions through role-based permissions.
Executive dashboard	Presents a high-level overview of consumers, meters, consumption records, suspicious alerts, and investigation activities.
Consumer management	Maintains consumer information needed to associate consumption records and meter assets with the relevant account.
Grid location management	Organizes distribution-location information and helps associate consumers and meters with their locations.
Smart-meter asset management	Maintains registered meter details and their associated consumer or location records.
Consumption record management	Supports the handling of electricity consumption records, including manual entry and CSV-based import and export.
AI anomaly detection	Processes selected consumption features using Isolation Forest and provides anomaly indicators for further examination.
Live monitoring	Displays incoming simulated meter readings and changing telemetry through WebSocket-based updates.
Suspicious alerts	Lists potentially abnormal records so authorized personnel can examine the associated information.
Investigation management	Supports the tracking of suspicious activities and the maintenance of investigation details and status.
Reports and export	Provides summaries and data export for consumption information, suspicious activities, and investigation records.
Notifications	Displays in-application notifications about relevant system activities and updates.
Audit logs	Maintains records of selected application activities to support traceability and review.
System settings	Provides access to configurable application options and administrative preferences.

TECHNOLOGIES AND IMPLEMENTATION

SmartGrid AI combines frontend, backend, database, real-time communication, and machine-learning technologies. The selected tools support a modular web application and a software demonstration of the monitoring workflow.

Technology	Purpose in the system
React	Development of the web frontend and reusable interface components.
Vite	Frontend development server and build tooling.
Tailwind CSS	Styling of the user interface.
JavaScript	Frontend application logic and interaction.
Python	Backend services and machine-learning implementation.
FastAPI	Backend API development and request handling.
Scikit-learn	Implementation of the Isolation Forest algorithm.
SQLite	Persistent storage of application records.
WebSocket	Real-time communication of incoming telemetry updates.

A. *Frontend Implementation*

The frontend is implemented using React with Vite. The application is organized into interfaces for dashboard monitoring, data administration, anomaly review, and reporting. Tailwind CSS provides a consistent visual system across these interfaces. The frontend requests data from backend services and presents the returned records in tables, summary cards, and charts.

B. *Backend Implementation*

FastAPI provides REST API endpoints and coordinates the application workflow. It receives requests from the frontend, validates the submitted information, interacts with the database, and invokes the anomaly detection component when required. Separating the API layer from the user interface supports maintainability and allows the backend logic to be tested independently.

C. *Database Implementation*

SQLite stores persistent application information. The stored entities include consumers, locations, meter assets, consumption records, alerts, investigations, and audit information. Relationships between these records provide the context required to review a suspicious reading and follow its investigation status.

D. *Machine-Learning and Simulator Implementation*

Scikit-learn is used to implement the Isolation Forest model. The model analyzes selected consumption features and produces indicators for unusual records. The software-based simulator generates sample meter readings and supports selected normal and abnormal scenarios for demonstration. These functions allow the data-ingestion and monitoring workflow to be exercised without claiming direct physical-meter connectivity.

E. *Integration*

The frontend communicates with backend services through API requests. The backend coordinates data validation, database operations, and anomaly detection. WebSocket communication distributes live telemetry updates to the monitoring interface. The combined workflow connects data entry and ingestion with visualization, alert review, and investigation management.

SECURITY, PRIVACY, AND RESPONSIBLE USE

Electricity consumption records can reveal information about consumer activity and should be handled with appropriate care. Smart-grid deployments must consider security and privacy risks across connected devices and communication systems [4]. SmartGrid AI includes role-based access to restrict the functions available to authorized users. In a production deployment, authentication credentials should be protected, network communication should use secure transport, and access to consumer and meter information should follow the organization's authorization policy.

Audit records can support traceability by documenting selected application actions. Operational deployments should define retention periods, backup procedures, access reviews, and incident-response processes. Data should be collected only for legitimate monitoring purposes and shared only with personnel who require it for their responsibilities. Test or demonstration data should be clearly distinguished from real consumer records.

The anomaly detection output must be interpreted responsibly. An unusual reading is not conclusive evidence of electricity theft. Before any enforcement or customer-impacting decision, authorized personnel should review the meter condition, communication status, consumption history, relevant operational context, and any available field evidence. The system is intended to assist prioritization and record management, not to replace human judgment or established investigation procedures.

FUNCTIONAL VERIFICATION AND EVALUATION FRAMEWORK

The implementation can be evaluated through functional checks of the main application workflow. The checks should verify that records can be created and retrieved, incoming simulated telemetry is displayed, anomaly indicators are surfaced, alerts can be reviewed, and investigation information can be maintained. These are functional verification activities; they should not be confused with a measured assessment of detection accuracy.

Area	Suggested verification	Evidence to retain
Authentication	Confirm authorized login and role-based access to relevant screens.	Test account roles and access results.
Data management	Create, view, update, and export consumer, meter, and consumption records.	Screenshots or test logs for each operation.
Telemetry	Run the simulator and confirm that new readings appear in the live interface.	Timestamped simulator and dashboard captures.
Anomaly screening	Submit representative normal and injected abnormal records and inspect model output.	Input samples, model configuration, and output records.
Alert workflow	Confirm that suspicious records are available for authorized review.	Alert record and review status.
Investigation	Create and update an investigation record linked to an alert.	Investigation record and audit trail.
Reporting	Generate a summary and verify that it reflects the stored test records.	Exported report and source record count.

A complete performance evaluation would require a labelled dataset containing both normal and confirmed abnormal consumption cases. Suitable metrics may include precision, recall, F1-score, false-positive rate, and detection latency. Results should be reported only after the dataset, train/test protocol, feature definitions, model parameters, and evaluation procedure have been documented. No numerical accuracy or field-performance result is claimed in this paper because such evidence was not included in the supplied project document.

LIMITATIONS AND FUTURE ENHANCEMENTS

A. *Current Limitations*

- The live telemetry demonstration uses software-generated readings rather than direct measurements from deployed physical smart meters.
- Anomaly indicators depend on the quality and representativeness of the input data and selected features.
- The Isolation Forest output identifies unusual patterns but does not determine the cause of those patterns.
- The current implementation uses SQLite; deployment requirements may call for a database architecture designed for larger-scale concurrent workloads.
- Detection performance has not been quantified in this paper using a labelled field dataset.

B. *Future Enhancements*

- Integrate authorized smart meters or IoT gateways to receive real consumption readings securely.
- Evaluate the anomaly detection model using representative labelled datasets and report standard performance metrics.
- Introduce configurable thresholds and model monitoring while documenting model versions and feature definitions.
- Extend dashboards with location-based summaries and time-series comparisons for authorized monitoring teams.
- Strengthen deployment security through secure transport, improved secrets management, backups, and access auditing.
- Explore scalable data storage and message-processing components if the volume of telemetry increases.
- Provide investigation workflow guidance and clearer explanations of the features contributing to an anomaly indicator.

CONCLUSION

This paper presented SmartGrid AI, a web-based platform designed to support electricity consumption monitoring and the identification of potentially suspicious usage patterns. The system integrates consumer and meter administration, consumption-record handling, Isolation Forest-based anomaly screening, simulated real-time telemetry, suspicious alerts, investigation management, reporting, and role-based access. Its modular implementation combines React, FastAPI, Python, Scikit-learn, SQLite, and WebSocket communication.

The platform demonstrates how machine-learning indicators can be connected to a practical monitoring and review workflow. The software-based simulator provides a controlled way to demonstrate data ingestion and live visualization, while the investigation module keeps human review central to the process. The current system should be regarded as a functional software demonstrator rather than a field-validated electricity theft detection solution. Future work will focus on physical meter integration, evaluation with labelled consumption datasets, and stronger operational validation.

ACKNOWLEDGMENT

The authors thank the Department of Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore, for the academic support and facilities provided during the development of this work.

REFERENCES

[1] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in Proc. IEEE International Conference on Data Mining, 2008, pp. 413–422.

[2] V. C. Gungor et al., "Smart Grid Technologies: Communication Technologies and Standards," IEEE Transactions on Industrial Informatics, vol. 7, no. 4, pp. 529–539, Nov. 2011.

[3] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, "Electricity theft: Overview, issues, prevention and a smart meter based approach to control theft," Energy Policy, vol. 39, no. 2, pp. 1007–1015, 2011.

[4] P. McDaniel and S. McLaughlin, "Security and Privacy Challenges in the Smart Grid," IEEE Security & Privacy, vol. 7, no. 3, pp. 75–77, May–June 2009.

[5] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "On Detecting Clustered Anomalies Using SCiForest," in Proc. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 2010.