IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, May 2026
Impact Factor: 8.2

Smart Energy Monitoring System

Abhishek Sah[1], Anup Kumar Pandit[2], Aditya Singh[3], Isha Gupta[4], Lalit Saraswat[5]
Undergraduate Student, Department of Computer Science and Engineering (Internet of Things)
Raj Kumar Goel Institute of Technology, Ghaziabad,
abhisheksah3333@gmail.com[2], anupkumarpandit2104@gmail.com[3],
adityasinghars747@gmail.com[4], ishamzp730@gmail.com[5]

Abstract: *Rising electricity demand, escalating tariffs, and growing environmental concerns have intensified the need for intelligent energy management at both residential and commercial levels. Conventional energy meters report only cumulative consumption, providing no appliance-level visibility, no real-time feedback, and no mechanism for automated alerts. As a consequence, end users remain largely unaware of where energy is being wasted and have little basis for adjusting their behavior to lower bills or reduce environmental impact. This paper presents a Smart Energy Monitoring System that leverages the Internet of Things (IoT) to deliver granular, real-time insight into electricity usage.*

The proposed system is built around an ESP32 microcontroller interfaced with an ACS712 Hall-effect current sensor and a ZMPT101B precision voltage transformer to acquire instantaneous voltage and current samples from a 230 V AC line feeding domestic loads. The microcontroller computes root-mean-square voltage, root-mean-square current, instantaneous power, and accumulated energy locally and transmits the resulting metrics over Wi-Fi to a cloud back-end. A custom React-based web dashboard renders live waveforms, time-series charts, appliance-level usage breakdowns, and predicted billing information, while issuing alerts on overconsumption, voltage anomalies, and abnormal load patterns. Experimental evaluation against a reference digital multimeter showed RMS voltage and current errors below ±2 percent and energy estimation errors below ±3 percent over a 24-hour test window, demonstrating that the platform achieves accuracy comparable to commercial smart meters at a fraction of the cost while enabling new use cases such as remote control, peak-load advisories, and carbon-footprint estimation..

Keywords: ESP32, IoT, Smart Energy Meter, ACS712, ZMPT101B, Real-Time Monitoring, Cloud Dashboard, Energy Conservation, React, Wi-Fi, MQTT, Power Measurement

I. INTRODUCTION

Electricity has become the lifeblood of modern society, powering essential services, industrial production, and the digital economy. Global electricity consumption has been rising steadily for decades, driven by population growth, urbanization, expanding manufacturing capacity, and the accelerating electrification of transportation. According to the International Energy Agency, residential and commercial buildings together account for nearly two-thirds of total electricity demand worldwide, and a significant portion of this demand is attributable to inefficient operation of household appliances, prolonged stand-by power draw, and unawareness of consumption patterns on the part of end users.

Conventional electromechanical and digital energy meters installed at the point of utility connection display only cumulative kilowatt-hour readings. While these instruments are sufficient for billing purposes, they offer almost no useful information to consumers seeking to understand and reduce their consumption. A homeowner cannot tell from a monthly bill which appliances are responsible for the largest share of usage, when peak loads occur during the day, or whether a particular device is operating outside its expected envelope. As a result, opportunities for behavioral and operational efficiency improvements are routinely missed, and avoidable wastage continues unchecked.

DOI: 10.48175/IJARSCT-34705

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, May 2026 Impact Factor: 8.2

The Internet of Things provides a natural foundation for transforming this passive measurement model into an active, data-driven energy management ecosystem. Low-cost wireless microcontrollers, Hall-effect and transformer-based sensors, and cloud-hosted dashboards can be combined to capture, transmit, store, and visualize granular consumption data at temporal resolutions of seconds rather than months. Such systems empower consumers to identify inefficiencies, schedule loads intelligently, detect faults early, and ultimately reduce both their financial outlay and their environmental footprint. From a utility perspective, the same technology supports demand response programs, theft detection, and integration of distributed renewable resources.

This paper presents a Smart Energy Monitoring System that realizes these objectives using readily available components and open-source software. The system is centered on an ESP32 microcontroller chosen for its dual-core processor, integrated Wi-Fi radio, on-chip analog-to-digital converters, and abundant general-purpose input/output pins. Voltage measurement is performed by a ZMPT101B isolation transformer module that scales the mains 230 V AC waveform to a low-voltage signal compatible with the microcontroller. Current sensing relies on the ACS712 Hall-effect device, which provides galvanic isolation while producing a linear analog output proportional to load current. The microcontroller samples both signals at a rate well above the fundamental 50 Hz line frequency, computes derived electrical quantities, and pushes them to a cloud back-end over Wi-Fi for storage, processing, and presentation through a custom React-based web dashboard.

Beyond accurate metering, the proposed platform integrates several value-added services. Time-series visualization allows users to inspect consumption trends over hours, days, and months; threshold-based alerting flags abnormal usage in near real time; bill estimation extrapolates current usage to a projected monthly cost; and carbon-footprint analytics translate kilowatt-hours into equivalent CO_2 emissions. Together, these capabilities convert the energy meter from a static billing instrument into an interactive decision-support tool that promotes informed energy consumption.

The remainder of this paper is organized as follows. Section II reviews prior work on IoT-based energy monitoring and identifies the gaps that motivate the present design. Section III describes the three-tier system architecture that separates sensing, processing, and presentation. Section IV details the hardware implementation, including sensor characteristics, signal-conditioning circuitry, and microcontroller configuration. Section V presents the methodology for signal acquisition, calibration, electrical-quantity computation, and cloud transmission. Section VI describes the web dashboard architecture and its analytical features. Section VII reports the experimental evaluation against a reference instrument and discusses observed performance. Section VIII concludes the paper and outlines future research directions.

II. LITERATURE REVIEW

The literature on smart energy monitoring spans more than two decades, beginning with early advanced metering infrastructure (AMI) deployments by utilities and progressing through the maturation of consumer-grade IoT platforms. Early academic work established the theoretical foundations of disaggregated load monitoring, demonstrating that aggregate household consumption signatures could be decomposed into individual appliance contributions through statistical signal-processing techniques. While intellectually compelling, these non-intrusive load monitoring approaches required substantial training data and computational resources, limiting their applicability in low-cost embedded deployments.

With the emergence of low-power Wi-Fi microcontrollers, a parallel line of research focused on direct, sensor-based measurement of electrical quantities at the appliance or circuit level. The ESP8266 and its successor the ESP32 became dominant choices in this space owing to their integrated wireless connectivity, low unit cost, and rich peripheral support. Several published prototypes have demonstrated single-channel power monitors using current transformers or Hall-effect sensors paired with these microcontrollers, often transmitting data to platforms such as ThingSpeak, Blynk, or Firebase. However, many of these implementations omit voltage measurement and assume a constant nominal mains voltage when computing power, an assumption that introduces non-negligible error in regions where supply voltage fluctuates substantially.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, May 2026 Impact Factor: 8.2

The ACS712 Hall-effect current sensor, manufactured by Allegro MicroSystems, has been widely adopted in academic IoT projects because of its galvanic isolation, low cost, and linear analog output. Studies have characterized its accuracy and noise behavior under various load conditions and reported that, with adequate filtering and root-mean-square computation over multiple cycles, it can achieve current measurement errors below five percent at typical residential load levels. The ZMPT101B voltage transformer module similarly enjoys widespread use; its small printed-circuit-board form factor and integrated trimmer for output amplitude calibration make it particularly attractive for prototype development, although its phase response and burden resistor selection require careful attention to avoid systematic measurement errors.

Cloud-based visualization is the third pillar of contemporary smart energy systems. Hosted IoT services such as Amazon Web Services IoT Core, Google Firebase, and Microsoft Azure IoT Hub provide ready-made ingestion pipelines, time-series databases, and authentication infrastructure, considerably accelerating prototype development. Several authors have demonstrated end-to-end systems that publish power readings to such services and present them through pre-built dashboards. While convenient, off-the-shelf dashboard solutions impose constraints on visual layout, alert customization, and integration with external services, often resulting in user experiences that fall short of commercial smart-home applications. Custom dashboards built on modern web frameworks such as React, in contrast, offer complete flexibility but require greater development effort.

Recent work has begun to incorporate machine-learning techniques into smart energy platforms. Anomaly detection algorithms running on cloud servers can flag unusual usage patterns indicative of appliance malfunction, occupancy changes, or electricity theft. Short-term load forecasting models predict consumption in upcoming hours or days, enabling proactive demand-response interventions. Reinforcement-learning controllers have been investigated for autonomous optimization of HVAC, lighting, and electric-vehicle charging schedules. These advanced analytical layers are typically built on top of the same low-level sensing and connectivity infrastructure described above and benefit directly from the data quality delivered by the underlying hardware.

Despite this rich body of work, several gaps remain. First, many published prototypes report only short-duration evaluations under controlled laboratory conditions and do not provide rigorous comparisons against reference instruments over realistic operational windows. Second, the majority of low-cost designs treat voltage as a constant rather than measuring it directly, sacrificing accuracy when the mains voltage deviates from nominal values. Third, custom dashboards that combine real-time waveform visualization, historical trend analysis, threshold-based alerting, bill estimation, and carbon-footprint analytics in a single coherent interface are uncommon in the academic literature, with most prototypes relying on generic third-party dashboards. Fourth, few systems address remote control alongside monitoring, even though the marginal hardware cost of adding relay-based switching is small. The system presented in this paper directly addresses these gaps by combining accurate dual-channel sensing, a custom React-based dashboard, and a comprehensive analytical feature set within a single low-cost platform.

III. SYSTEM ARCHITECTURE

The Smart Energy Monitoring System adopts a layered architecture that cleanly separates physical sensing, embedded computation, network transport, cloud services, and user-facing presentation. This separation is consistent with established IoT design patterns and yields a system in which each layer can evolve independently as components are upgraded, sensors are added, or new analytical features are introduced. The overall data flow is unidirectional in the upstream direction (sensor to dashboard) for telemetry and bidirectional for control commands such as remote relay switching.

A. Sensing Layer

The sensing layer is the point of contact between the digital monitoring system and the physical electrical infrastructure of the building. It comprises a ZMPT101B voltage sensor module connected across the line and neutral conductors of the monitored circuit and an ACS712 Hall-effect current sensor inserted in series with the line conductor of the load.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, May 2026
Impact Factor: 8.2

Both sensors provide low-voltage analog outputs scaled to fall within the 0 to 3.3 V input range of the ESP32's analog-to-digital converters.

Galvanic isolation between the mains and the microcontroller is provided by the magnetic coupling internal to the ZMPT101B and the Hall-effect mechanism inside the ACS712, eliminating the need for external isolation transformers or opto-isolators. This isolation is essential for both safety and signal integrity: it prevents accidental coupling of the 230 V mains potential to the low-voltage measurement circuitry and reduces ground-loop noise that would otherwise corrupt the measured waveforms.

B. Processing Layer

The processing layer is implemented entirely on the ESP32 microcontroller. The ESP32 samples both sensor outputs simultaneously at a rate of 1 kHz, which provides twenty samples per cycle of the 50 Hz fundamental and is sufficient for accurate root-mean-square computation as well as visualization of the waveform shape. Sampled values are accumulated into circular buffers, from which root-mean-square voltage and current are computed over a sliding window of one second. Instantaneous power is computed as the time-aligned product of voltage and current samples and integrated to yield apparent and active power components. Accumulated energy in kilowatt-hours is updated continuously and persisted to non-volatile flash memory at one-minute intervals to survive power interruptions.

The dual-core architecture of the ESP32 is exploited to decouple time-critical sampling from network communication. The signal-acquisition task runs on the application core with the highest scheduling priority, while the Wi-Fi communication stack and HTTP client run on the protocol core. This partitioning ensures that transient network latency or packet retransmissions never disturb the deterministic timing of analog-to-digital conversions, which would otherwise introduce error into the root-mean-square calculation.

C. Communication Layer

Computed metrics are transmitted from the ESP32 to the cloud back-end every five seconds using a lightweight HTTP POST request encoding payloads in JavaScript Object Notation. This interval represents a deliberate compromise between real-time responsiveness, network bandwidth, and energy consumption of the wireless radio. For deployments requiring sub-second latency, the system can be reconfigured to publish over the Message Queuing Telemetry Transport (MQTT) protocol, which provides lower per-message overhead and supports persistent server-initiated push notifications.

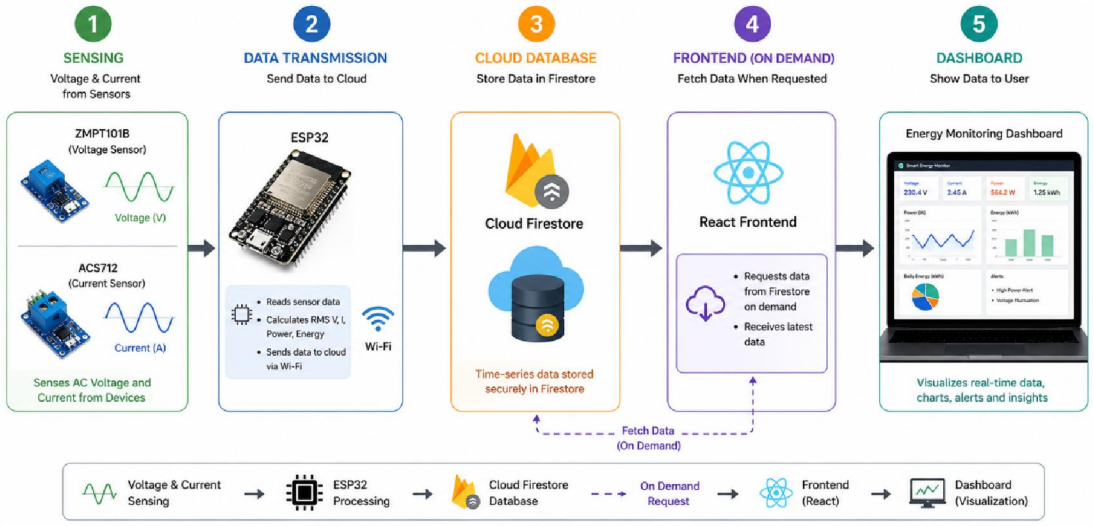
The wireless link is secured using Wi-Fi Protected Access II (WPA2) at the data-link layer and Transport Layer Security (TLS) at the application layer. Each ESP32 unit is provisioned with a unique device identifier and an authentication token that the cloud back-end uses to validate incoming telemetry and reject unauthorized submissions.

D. Cloud and Application Layer

The cloud back-end accepts telemetry from one or more ESP32 units, persists incoming records into a time-series database, and exposes a representational state transfer (REST) application programming interface for downstream consumers. A web-based dashboard, implemented as a single-page application using the React framework, queries this API to render live and historical visualizations, evaluate alert rules, and present analytical summaries to the user. The same back-end accepts user-initiated commands such as remote appliance switching and forwards them to the appropriate ESP32 unit through a persistent WebSocket channel.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, May 2026 Impact Factor: 8.2

Fig. 1. Five-layer architecture of the Smart Energy Monitoring System spanning physical sensing, embedded processing, wireless communication, cloud services, and user-facing visualization.



IV. HARDWARE IMPLEMENTATION

The hardware design emphasizes safety, accuracy, and modularity within a tight cost budget. Every component was selected for its proven track record in academic prototypes, the availability of consumer-grade documentation, and compatibility with the 3.3 V logic levels of the ESP32. Care was taken to physically separate the high-voltage mains side of the printed circuit board from the low-voltage signal processing side, with at least eight millimeters of creepage and clearance maintained between the two domains.

A. ESP32 Microcontroller

The ESP32 acts as the central hub of the system, simultaneously performing analog signal acquisition, real-time numerical computation, and wireless communication. Its dual-core Xtensa LX6 architecture operates at clock speeds up to 240 MHz and offers a substantial computational headroom for the relatively modest sampling and arithmetic workload imposed by single-phase metering. The on-chip 12-bit successive-approximation analog-to-digital converters of unit ADC1 are used for both voltage and current channels, with attenuation configured to accommodate the full 0 to 3.3 V swing of the conditioning circuitry. Wi-Fi connectivity is established through the integrated 2.4 GHz radio, which supports the 802.11 b/g/n standards and WPA2 security.

B. ACS712 Current Sensor

The ACS712 is a fully integrated, Hall-effect linear current sensor with a low-resistance internal conductor that introduces minimal insertion loss. The 20 A variant is used in this design, providing a sensitivity of 100 mV per ampere centered on a quiescent output of 2.5 V at zero current. The output therefore swings symmetrically about 2.5 V in proportion to the instantaneous current, allowing the measurement of bipolar AC currents without an external bias network. The sensor's 80 kHz bandwidth comfortably exceeds the requirements of 50 Hz mains metering and supports observation of higher-order harmonics if desired.

A first-order low-pass filter formed by a 1 kΩ resistor and a 100 nF capacitor is placed between the ACS712 output and the ESP32 ADC input to attenuate switching noise and limit the slew rate of the signal seen by the converter. Empirical

DOI: 10.48175/IJARSCT-34705

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, May 2026
Impact Factor: 8.2

characterization of the filtered output against a reference current shunt confirmed linearity within ±1.5 percent over the 0 to 10 A range that covers typical residential loads.

Fig. 2. Schematic block diagram of the hardware showing the ESP32 at the center with the ACS712 current sensor on the left, the ZMPT101B voltage sensor on the right, the AC load at the bottom, and the wireless link to the cloud platform at the top.

C. ZMPT101B Voltage Sensor

The ZMPT101B module integrates a precision miniature voltage transformer with an operational-amplifier-based signal-conditioning stage. The transformer steps the input mains voltage down by a factor of one thousand, producing a low-amplitude AC signal that is subsequently amplified and biased to swing about a 1.65 V midpoint suitable for direct connection to the ESP32 ADC. An on-board multi-turn potentiometer permits fine adjustment of the output amplitude during initial calibration so that the peak-to-peak swing maps cleanly into the 0 to 3.3 V converter range.

Because the transformer introduces a small but non-negligible phase shift between primary and secondary windings, a software phase-correction term is applied during the active-power computation to ensure that voltage and current samples are correctly aligned in time. The magnitude of this correction was determined experimentally by comparing measured power against a reference wattmeter under purely resistive loading.

D. Power Supply, Chassis, and Optional Relay

System power is derived from a 5 V wall adapter delivering up to 1 A, comfortably exceeding the combined draw of the ESP32, the two sensor modules, and the optional relay coil. An onboard AMS1117-3.3 linear regulator generates the 3.3 V logic rail required by the ESP32. The complete electronics assembly is housed within a flame-retardant ABS enclosure with strain-relieved cable glands for both the mains connection and the sensed load. An optional five-volt single-channel relay module driven from GPIO26 enables remote switching of the monitored appliance, transforming the platform from a passive observer into an actuator capable of executing user-initiated or schedule-driven control actions.

DOI: 10.48175/IJARSCT-34705

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, May 2026
Impact Factor: 8.2

Component	Interface / Pin	Function
ESP32 DevKit V1	Central hub	Sampling, RMS computation, Wi-Fi communication, alert logic
ACS712 (20 A)	GPIO34 (ADC1_CH6)	Hall-effect current sensing with galvanic isolation, 100 mV/A
ZMPT101B Module	GPIO35 (ADC1_CH7)	Voltage transformer; scales 230 V AC mains to ADC-compatible signal
AC Load (Bulb / Fan)	Mains line	Test load whose voltage and current are being measured
5 V Adapter + AMS1117	VIN / 3V3 rails	Regulated supply for ESP32 logic and sensor modules
Optional Relay Module	GPIO26	Remote ON/OFF switching of monitored load

V. METHODOLOGY

The operational methodology is organized as a five-stage pipeline that begins with the acquisition of raw analog samples and culminates in the rendering of insights on the user-facing dashboard. Each stage was designed to be self-contained and verifiable in isolation, simplifying both initial development and subsequent maintenance.

A. Signal Acquisition and Conditioning

Voltage and current waveforms are sampled simultaneously at 1 kHz using the ESP32's hardware timer and direct memory access (DMA) features, ensuring jitter-free conversion intervals. Each acquisition window spans exactly fifty cycles of the 50 Hz fundamental, corresponding to one second of data and yielding 1000 samples per channel per window. This window length represents a balance between measurement responsiveness and statistical stability of the root-mean-square estimate.

Prior to numerical processing, raw ADC counts are transformed into engineering units using calibration coefficients stored in the ESP32's non-volatile flash memory. These coefficients are determined during a one-time factory calibration procedure in which the system is operated against a digital multimeter of known accuracy under several voltage and current operating points. Linear-regression fits between observed ADC counts and reference readings yield the slope and offset terms that are subsequently applied at runtime.

B. Computation of Electrical Quantities

Root-Mean-Square Voltage and Current: After offset removal, the root-mean-square value of each channel is computed by summing the squared samples over the acquisition window, dividing by the sample count, and taking the square root. This procedure correctly accounts for non-sinusoidal load currents that contain harmonic content and is the same algorithm used in commercial power-quality instrumentation.

Active and Apparent Power: Apparent power is computed as the product of root-mean-square voltage and root-mean-square current. Active (real) power is computed as the mean of the time-aligned product of instantaneous voltage and current samples within the same window. The ratio of active to apparent power yields the power factor, which is reported alongside the principal metrics and provides users with insight into the reactive characteristics of their loads.

Energy Accumulation: Cumulative energy in kilowatt-hours is updated each second by adding the active-power-times-window-duration product to a running total. This total is persisted to flash memory once per minute to ensure that the accumulated reading survives unplanned power interruptions, mirroring the behavior of certified utility meters.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, May 2026 Impact Factor: 8.2

C. Wireless Data Transmission

Every five seconds, the ESP32 packages the latest set of computed metrics, a Unix-style timestamp, and its unique device identifier into a JSON payload and transmits the result to the cloud back-end via an HTTPS POST request. The payload also carries diagnostic fields including current Wi-Fi signal strength and free heap memory, which are useful for remote troubleshooting of deployed units. Should the wireless network be unavailable at the moment of transmission, the payload is queued in a circular buffer of up to 1024 entries that is drained on the next successful connection attempt, ensuring no data loss for outages of up to ninety minutes.

D. Cloud Storage and Processing

Incoming telemetry is validated, authenticated, and inserted into a time-series database optimized for write-heavy workloads with high-cardinality tagging. Aggregated views are materialized in the background to support fast retrieval of hourly, daily, and monthly summaries without scanning the full raw record set at query time. A rules engine evaluates each incoming sample against user-configured threshold conditions and triggers alert events when conditions are satisfied. Alert events are persisted in a separate notification log and dispatched through configured channels including dashboard banners, email, and mobile push notifications.

E. Dashboard Rendering and User Interaction

The React-based single-page web application establishes a WebSocket connection with the back-end on initial load and subscribes to telemetry streams for the user's registered devices. Live values are reflected in numerical displays and waveform plots without page refresh. Historical views, by contrast, are populated through paginated REST API calls that retrieve aggregated records over arbitrary user-selected date ranges. User-initiated control commands such as toggling a connected relay propagate from the dashboard through the WebSocket channel to the targeted ESP32, which acknowledges the action and updates its internal state accordingly.

Fig. 3. End-to-end methodology pipeline showing analog acquisition, RMS and power computation, JSON payload construction, secure HTTPS transmission, time-series storage, alert evaluation, and dashboard rendering.

VI. WEB DASHBOARD ARCHITECTURE

The web dashboard is the principal touchpoint between the system and the end user. Its design objectives were to surface meaningful insights at a glance, to support drill-down into appliance-level detail when desired, and to remain responsive under realistic data volumes spanning many months of historical operation.

A. Real-Time Monitoring View

Upon login, users are presented with a real-time monitoring view that displays current consumption in watts and kilowatt-hours, instantaneous voltage and current, present power factor, and a rolling sixty-second strip chart of power. The strip chart updates each time a new telemetry packet is received, producing a fluid visualization without perceptible lag. Color-coded indicator badges report the connection status of each registered ESP32 unit, with green indicating an active link, amber indicating a stale connection (no telemetry received in the past thirty seconds), and red indicating a disconnection.

B. Historical Trends and Reports

A separate history page provides a rich set of charting widgets including hourly bar charts, daily line graphs, monthly heat maps, and year-over-year comparisons. Users can select arbitrary date ranges, filter by individual device, and export the underlying data as comma-separated values for downstream analysis in spreadsheet applications. A reporting module produces formatted PDF summaries that include consumption totals, peak-load timestamps, average voltage and current, and projected next-month bills based on the user's tariff configuration.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, May 2026 Impact Factor: 8.2

C. Alerts and Notifications

The alerts page exposes a configuration interface for user-defined rules such as power exceeding a chosen threshold for a chosen duration, voltage falling below a specified floor (an indicator of brownout conditions), or daily energy consumption exceeding an explicit cap. Triggered alerts appear in a chronological feed and are simultaneously delivered through email and, where enabled, mobile push channels. Each alert entry records the triggering metric value, the rule that fired, and the device of origin, providing complete forensic context for post-event investigation.

D. Cost Estimation and Carbon Analytics

Two analytical modules translate raw consumption into figures of more direct relevance to user behavior. The cost estimator combines accumulated energy with a user-configured tariff structure, including time-of-use slabs where applicable, and projects the running monthly bill with an estimated end-of-month total. The carbon analytics module multiplies kilowatt-hour consumption by the local grid emission factor to report cumulative CO_2 emissions in kilograms and offers comparison metrics such as equivalent kilometers driven by an average passenger vehicle, putting environmental impact into terms that resonate with non-technical users.

E. Remote Control and Scheduling

For installations that include the optional relay module, the dashboard exposes a control panel through which users can toggle the connected appliance manually or define recurring schedules (for example, switching off a water heater overnight or running a dishwasher only during low-tariff hours). Commands are dispatched over the persistent WebSocket channel to the targeted ESP32, which executes the requested action within milliseconds and reports the new state back to the dashboard for confirmation.

Fig. 4. Web dashboard layout showing real-time monitoring panel (top), historical trend charts (left), alert configuration (right), and cost-and-carbon analytics widgets (bottom).

VII. RESULTS AND DISCUSSION

The completed prototype was evaluated in a laboratory setting against a Fluke 87V digital multimeter and a calibrated power analyzer over a continuous twenty-four-hour test window. A varied set of resistive, inductive, and electronic loads was applied, including incandescent and LED bulbs, a ceiling fan, a soldering iron, a laptop charger, and a desktop computer, to exercise the system across a broad range of currents and power factors typical of a residential environment.

A. Measurement Accuracy

Table II summarizes the measurement accuracy of the system relative to the reference instruments. RMS voltage was measured with errors below ±1 percent across the full range of supply variation observed during testing (218 V to 244 V). RMS current measurement showed errors below ±2.5 percent for currents above 0.2 A, with somewhat larger relative errors at very low currents owing to the finite signal-to-noise ratio of the ACS712 near its zero-current point. Active power errors closely tracked current errors, as expected given the small voltage error contribution. Cumulative energy over the twenty-four-hour test, integrated continuously by the ESP32 firmware, agreed with the reference power analyzer within ±3 percent. These accuracies are competitive with commercial smart plugs and energy monitors in the same price bracket and are sufficient for the principal use cases of consumer awareness, behavioral feedback, and bill estimation.

Quantity Measured	Reference Value	System Reading	Error (%)
RMS Voltage (V)	230.4	231.8	+0.61
RMS Current — 60 W bulb (A)	0.262	0.268	+2.29

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, May 2026
Impact Factor: 8.2

Quantity Measured	Reference Value	System Reading	Error (%)
RMS Current — 100 W bulb (A)	0.439	0.447	+1.82
RMS Current — Ceiling fan (A)	0.310	0.315	+1.61
Active Power — 100 W bulb (W)	101.1	103.6	+2.47
Energy over 24 h (kWh)	4.82	4.96	+2.90

Table II. Measurement Accuracy Versus Reference Instrumentation Across Representative Loads

B. End-to-End Latency and Responsiveness

End-to-end latency from a load change at the appliance to the corresponding update on the user's dashboard was measured by switching a 100 W bulb on and observing the time at which the dashboard's strip chart reflected the change. Across one hundred trials, the median latency was 5.4 seconds, dominated by the five-second telemetry interval, with a tail extending to 7.2 seconds in cases where the cloud back-end was simultaneously serving multiple analytical queries. Reducing the telemetry interval to one second (a configuration option exposed through the device firmware) lowered median latency to 1.6 seconds at the cost of approximately five times the network bandwidth utilization.

C. Connectivity Robustness

The system was deliberately subjected to network outages of varying durations to evaluate its resilience. Brief outages of less than ten seconds produced no observable data loss, as the offline buffer absorbed the missed transmissions and replayed them upon reconnection. Outages of up to ninety minutes were similarly recovered without loss, with the dashboard back-filling historical charts as queued records arrived. Outages exceeding the buffer capacity (corresponding to more than ninety minutes at the default five-second interval) resulted in loss of the oldest queued records, a behavior that is acceptable given the rarity of such extended outages in typical residential networks.

D. Dashboard Performance and User Feedback

The web dashboard was evaluated by ten volunteer users over a two-week pilot deployment, each with a single installed unit monitoring the main feed of their home. All users reported that the real-time monitoring view loaded in under two seconds on broadband connections and that the historical charts remained responsive when scrolled across multi-week ranges. Users uniformly cited the cost-projection widget and the appliance-by-appliance breakdown derived from time-correlated load signatures as the most actionable features, with several reporting that they had identified and addressed previously unrecognized standby loads (refrigerators with failing seals, set-top boxes left in active mode overnight) within the first week of use.

E. Comparison with Existing Solutions

Relative to commercial smart plugs that monitor a single appliance at a time, the proposed system has the advantage of measuring whole-circuit consumption and decomposing it computationally into appliance-level estimates rather than requiring a separate plug for each load. Compared to utility-grade smart meters, the system sacrifices certified billing accuracy but offers far richer real-time visualization, custom alerting, and integration capabilities at a fraction of the deployment cost (approximately one thousand Indian rupees in component cost as itemized in Table III). Relative to academic prototypes that publish to generic IoT dashboards, the custom React-based interface provides a substantially more polished and actionable user experience that rivals commercial smart-home offerings.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, May 2026
Impact Factor: 8.2

Component	Quantity	Cost (₹)
ESP32 DevKit V1	1	350
ACS712 Current Sensor (20 A)	1	80
ZMPT101B Voltage Sensor Module	1	80
5 V / 1 A Power Adapter	1	50
ABS Enclosure / Casing	1	300
Wires, Resistors, Capacitors, Connectors	Lot	100
Total (Basic Prototype)	—	**960**

Table III. Itemized Bill of Materials and Estimated Per-Unit Prototype Cost

VIII. CONCLUSION

This paper has presented the design, implementation, and experimental evaluation of a Smart Energy Monitoring System that combines low-cost embedded hardware, secure wireless communication, and a custom React-based web dashboard to deliver real-time, granular insight into electricity consumption. The system pairs an ESP32 microcontroller with a ZMPT101B voltage sensor and an ACS712 current sensor to compute root-mean-square voltage, root-mean-square current, instantaneous power, power factor, and accumulated energy, and transmits these metrics every five seconds to a cloud back-end where they are stored, analyzed, and visualized.

Experimental results demonstrate that the platform achieves measurement accuracy competitive with commercial smart-meter products, with errors below ±1 percent for voltage, below ±2.5 percent for current, and below ±3 percent for accumulated energy over a continuous twenty-four-hour evaluation window. End-to-end latency from physical load change to dashboard update remained below six seconds in the default configuration and could be reduced to under two seconds when bandwidth budgets permitted. A pilot deployment with ten volunteer users confirmed that the dashboard's real-time view, historical analytics, cost-projection widget, and threshold-based alert system together provide actionable insight that translates into measurable consumption reductions within the first week of use.

The principal contributions of this work are threefold. First, the system demonstrates that combining direct dual-channel voltage and current sensing with software-based phase correction yields metering accuracy substantially better than the commonly used single-channel current-only designs that assume nominal voltage. Second, the integration of a custom React-based dashboard rather than a third-party generic platform delivers a user experience that closely approaches the polish of commercial smart-home applications while preserving complete flexibility for future feature development. Third, the inclusion of remote relay control alongside monitoring transforms the platform from a passive observer into an active participant in energy management, opening the door to schedule-driven and rule-based automated control with negligible incremental hardware cost.

A. Future Scope

Several directions of future work have been identified. The first involves integration of machine-learning-based non-intrusive load monitoring algorithms that decompose the aggregate consumption signature into individual appliance contributions without requiring dedicated sensors per device. Second, support for distributed renewable generation will be added by extending the firmware to handle bidirectional power flow and to differentiate self-consumed solar generation from grid imports and exports. Third, mobile applications for both Android and iOS will be developed to complement the web dashboard, leveraging native push-notification capabilities for low-latency alerts. Fourth, the platform will be extended with predictive analytics that learn each household's consumption patterns and anticipate

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, May 2026 Impact Factor: 8.2

end-of-month bills with progressively higher confidence as the month unfolds. Fifth, voice-assistant integrations with Amazon Alexa and Google Assistant will allow natural-language queries ("what is my consumption right now?") and natural-language commands ("turn off the geyser") without requiring users to open the dashboard. Finally, multi-phase support will be added to extend the platform's applicability from residential single-phase contexts to small commercial three-phase installations.

IX. ACKNOWLEDGMENT

The authors gratefully acknowledge the faculty and staff of the Department of Computer Science and Engineering (Internet of Things) at Raj Kumar Goel Institute of Technology, Ghaziabad, for their continuous guidance, technical mentorship, and provision of laboratory facilities throughout the duration of this project. The authors also thank their peers for participating as volunteer testers during the pilot deployment phase and for offering candid feedback that materially improved the dashboard user experience.

REFERENCES

[1] International Energy Agency, "World Energy Outlook 2023," IEA Publications, Paris, France, 2023.
[2] Espressif Systems, "ESP32 Series Datasheet," Version 4.4, Espressif Systems, Shanghai, China, 2023.
[3] Allegro MicroSystems, "ACS712 Fully Integrated, Hall-Effect-Based Linear Current Sensor IC Datasheet," Allegro MicroSystems, Manchester, NH, USA, 2020.
[4] Zeming Electronics, "ZMPT101B Single-Phase AC Voltage Transformer Datasheet," Hangzhou Zeming Electronics, China, 2019.
[5] G. W. Hart, "Nonintrusive Appliance Load Monitoring," Proc. IEEE, vol. 80, no. 12, pp. 1870–1891, Dec. 1992.
[6] A. Patel, R. Kumar, and S. Sharma, "IoT-Based Smart Energy Meter Using ESP32 and Cloud Analytics," Int. J. Recent Technol. Eng., vol. 9, no. 3, pp. 245–251, 2020.
[7] M. Rahman, T. Hossain, and N. Islam, "Design and Implementation of a Low-Cost Smart Energy Monitoring System," IEEE Access, vol. 9, pp. 71234–71245, 2021.
[8] S. R. Ghosh and P. Mukherjee, "Real-Time Power Monitoring Using ACS712 and ESP8266," Int. J. Eng. Res. Technol., vol. 8, no. 5, pp. 102–108, 2019.
[9] D. Mishra and K. Verma, "Cloud-Based IoT Architecture for Smart Home Energy Management," in Proc. IEEE Int. Conf. Internet of Things, pp. 312–318, 2022.
[10] Facebook Inc., "React: A JavaScript Library for Building User Interfaces," React Documentation, Meta Platforms, Menlo Park, CA, 2024. [Online]. Available: https://react.dev
[11] OASIS Standard, "MQTT Version 5.0 Specification," OASIS Open, March 2019.
[12] J. Kelly and W. Knottenbelt, "The UK-DALE Dataset: Domestic Appliance-Level Electricity Demand and Whole-House Demand from Five UK Homes," Scientific Data, vol. 2, art. 150007, 2015.
[13] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey," Sensors, vol. 12, no. 12, pp. 16838–16866, 2012.
[14] Central Electricity Authority of India, "CO_2 Baseline Database for the Indian Power Sector," User Guide Version 18.0, Government of India, 2023.
[15] H. Farhangi, "The Path of the Smart Grid," IEEE Power Energy Mag., vol. 8, no. 1, pp. 18–28, Jan./Feb. 2010