EnergyOS AI

The AI Energy Operating System for MSMEs

Vision

EnergyOS AI is an Industrial AI + IoT platform that helps manufacturing industries monitor, analyze, and optimize their electricity consumption in real time.

Instead of simply displaying electricity readings, EnergyOS transforms raw electrical data into intelligent insights, recommendations, and predictive analytics that help factories reduce operational costs and improve efficiency.

EnergyOS works as the **intelligence layer** between industrial electrical systems and business decision-makers.

What is EnergyOS?

Think of EnergyOS as

Google Analytics + Industrial IoT + Artificial Intelligence + Power BI = EnergyOS AI

It continuously collects energy data from factory equipment, processes it using AI, and presents meaningful insights through an interactive dashboard.

Target Industries

1. Commercial Buildings
2. Manufacturing Factories
3. Textile Industries
4. Food Processing Plants
5. Cold Storage
6. Warehouses
7. Hospitals
8. Hotels

Key Objectives

1. Monitor electricity usage in real time
2. Reduce energy costs
3. Improve machine efficiency
4. Predict future electricity bills
5. Detect abnormal energy consumption
6. Enable preventive maintenance
7. Generate AI-powered reports
8. Human Safety

Why EnergyOS?

Traditional energy monitoring systems only display data.
EnergyOS AI goes one step further by answering questions like:

- Why did electricity consumption increase?
- Which machine is wasting energy?
- How much can we save this month?
- Which machine requires maintenance?
- What action should be taken today?

Problem Statement & Solution

Problem Statement

Many MSMEs rely only on their monthly electricity bill to understand energy consumption.
As a result, they cannot identify:

- Which machine consumes the most electricity
- Why electricity bills suddenly increase
- Peak demand timings
- Idle machines wasting electricity
- Poor power factor
- Inefficient equipment
- Preventive maintenance requirements

Our Solution

EnergyOS AI continuously monitors electrical parameters from the factory using Industrial IoT devices.
The collected data is securely transferred to the cloud where Artificial Intelligence analyzes energy consumption patterns and generates intelligent recommendations.
Instead of reacting after receiving the monthly bill, factory managers can take action every day.

Key Features

✓ Lower Electricity Bills
✓ Better Machine Utilization
✓ Reduced Downtime
✓ AI-Based Decision Making
✓ Sustainable Manufacturing
✓ Carbon Emission Analysis
✓ Smart Recommendations
✓ Preventive Maintenance
✓ Energy Waste Detection

Live Monitoring
1. Voltage
2. Current
3. Power
4. Energy
5. Frequency
6. Power Factor

Business Features
1. PDF Reports
2. Email Alerts
3. Mobile Dashboard
4. Admin Panel
5. Historical Analytics
6. Enable Customization

Budget & Revenue

Estimated Budget Utilization

Category	Estimated Cost (₹)
Product Design (UI/UX)	1,00,000
Software Development	2,50,000
AI & Analytics Development	2,00,000
IoT Sensors & Prototyping	1,50,000
Edge Devices & Hardware	1,00,000
Cloud Infrastructure (1 Year)	1,50,000
Pilot Deployment & Testing	2,00,000
Documentation & Miscellaneous	1,00,000
Total	**₹ 12,50,000**

Revenue Model

SaaS Subscription Model

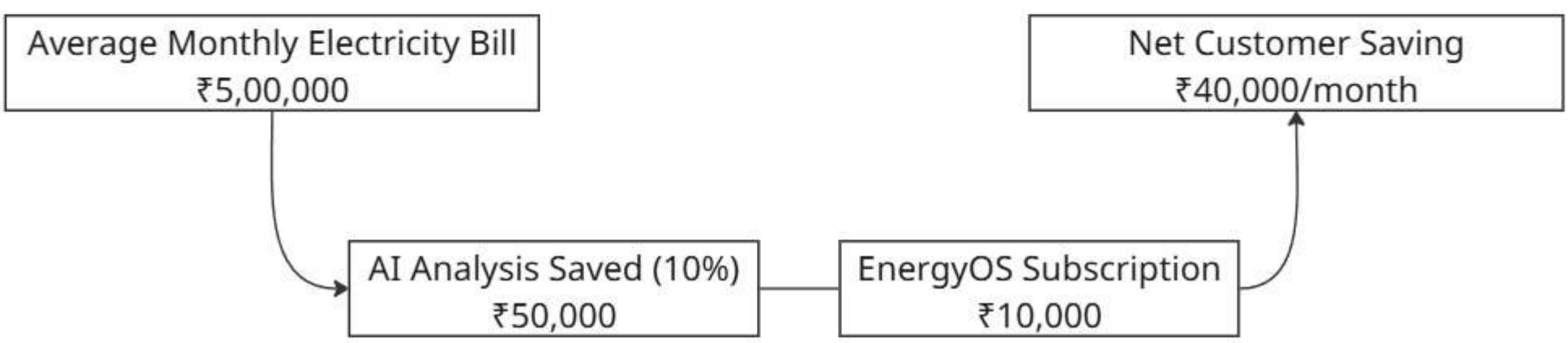
Factories subscribe to EnergyOS AI based on the number of machines and monitoring requirements.

Plan	Target Customer	Monthly Fee
Starter	Small MSME (1-10 Machines)	₹ 2,000/ month
Growth	Medium factory (1-50 Machines)	₹ 10,000/ month
Enterprise	Large Industry (50+ Machines)	₹ 25,000 - ₹50,000/ month

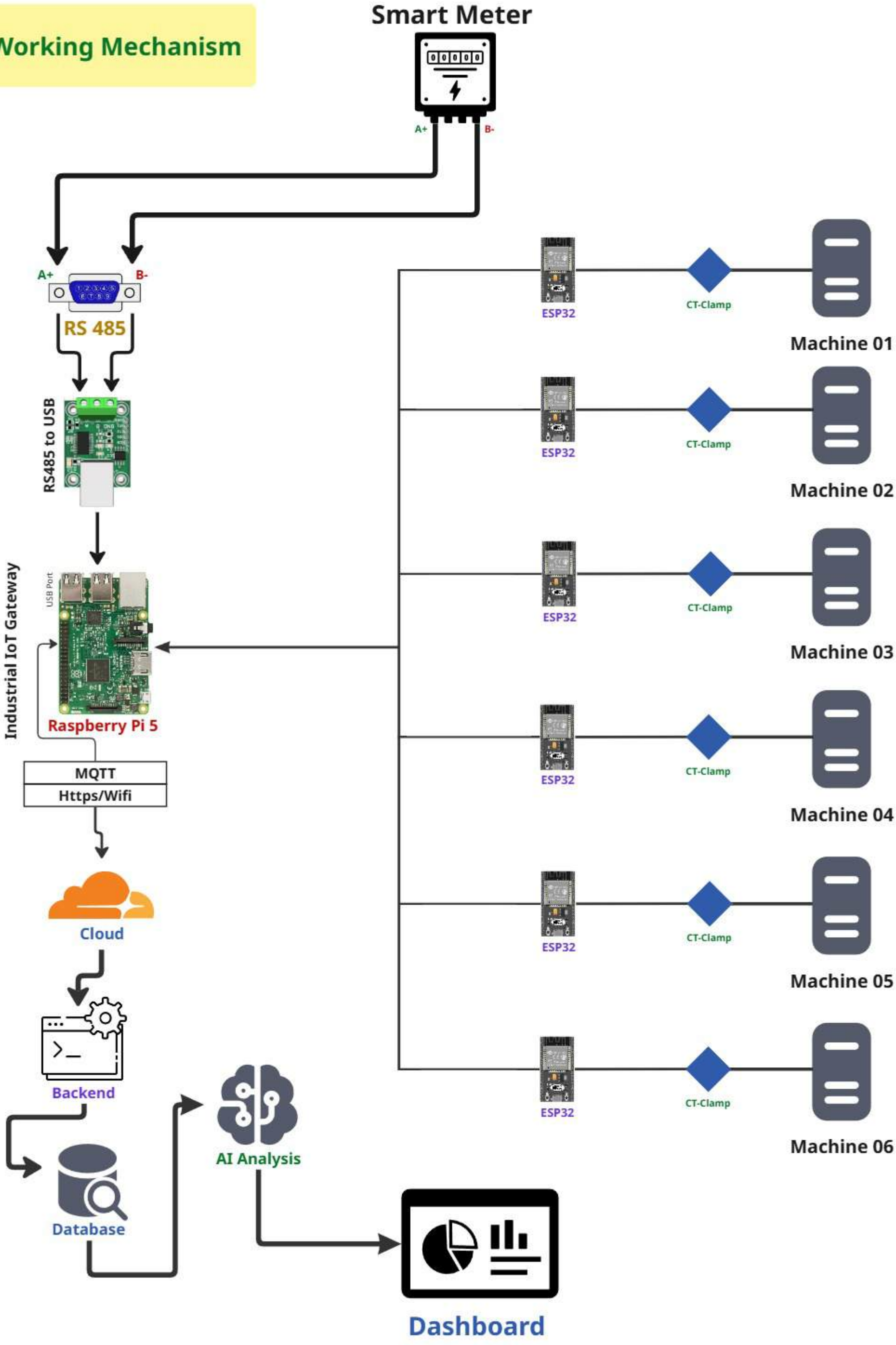
Additional Revenue Sources

- One-time hardware installation charges
- Annual maintenance contracts (AMC)
- AI-powered energy audit services
- Carbon footprint reporting
- Custom dashboard development
- PLC and SCADA integration
- Solar energy monitoring module
- Premium analytics & predictive maintenance

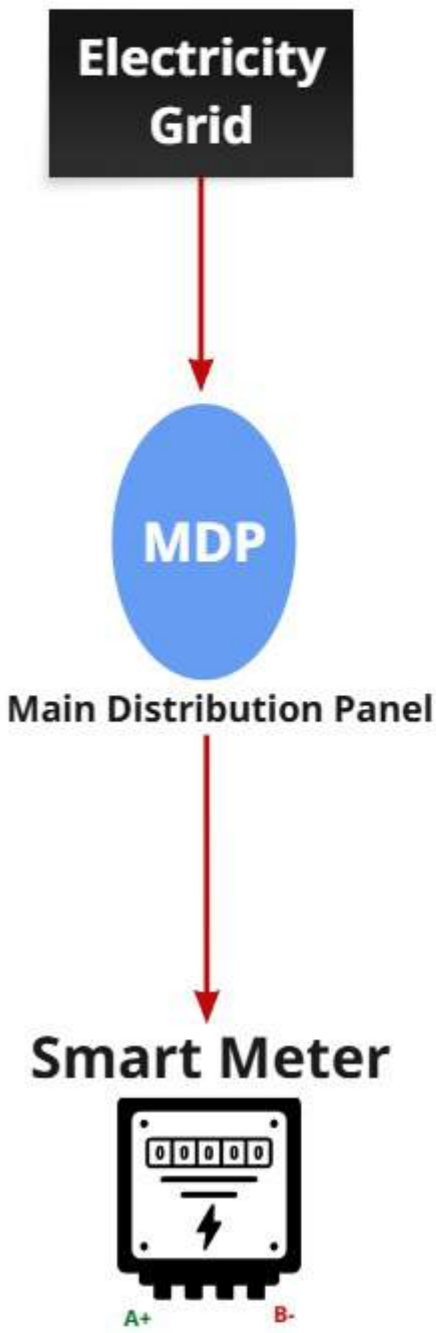
Customer Value Proposition



Working Mechanism



Smart Energy Meter



Purpose

The Smart Energy Meter is installed at the **Main Distribution Panel (MDP)** of the factory. It continuously measures the electricity entering the entire factory.

Unlike a normal electricity meter that only provides a monthly bill, the Smart Meter provides r**eal-time electrical parameters** every few seconds.

It acts as the first data collection device in the **EnergyOS AI system.**

Electrical Parameters Measured

Every few seconds, the meter provides the following values:

- Voltage
- Current
- Power
- Active Power (kW)
- Reactive Power (kVAR)
- Energy (kWh)
- Frequency
- Power Factor

Output

The Smart Meter converts these electrical values into digital data using the **Modbus RTU Protocol.**

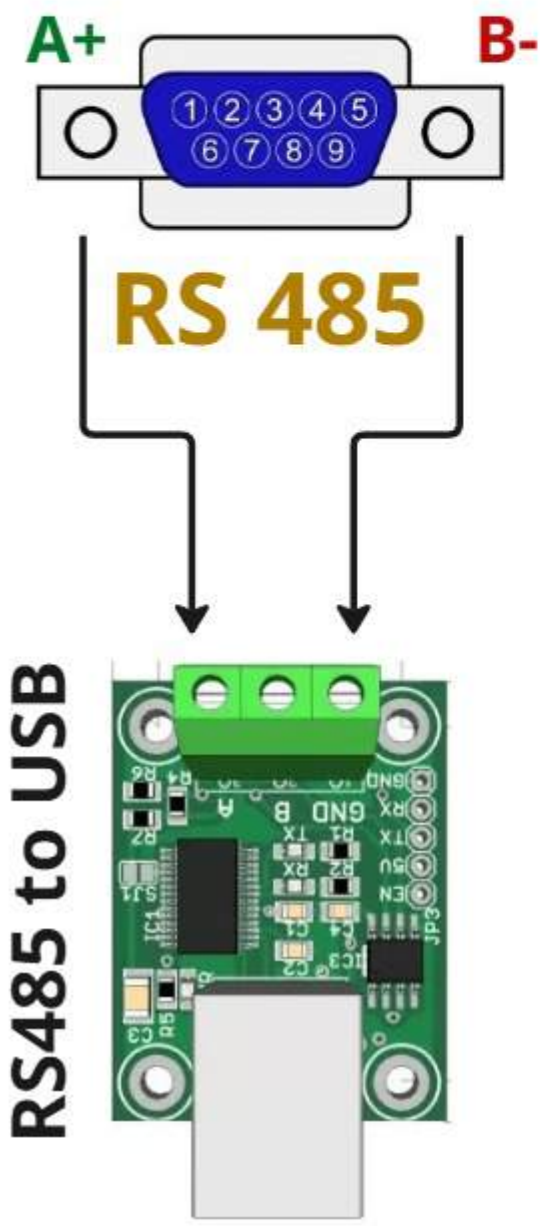
Next Connection

The smart meter connects to the RS485 network using the **positive (A+)** and **negative (B−)** terminals. If a GND terminal is available, it should also be connected. Most smart meters clearly label these terminals.

Example

This solution is compatible with **various smart meters**, including the Schneider EM6400, Siemens PAC2200, ABB EQmatic, L&T EM101+, and Secure Elite 440.

RS485 Communication Network



Purpose

RS485 is **not a sensor.** It is an industrial communication standard used to transfer data from the Smart Meter to the IoT Gateway. Think of RS485 as a secure communication cable between devices.

Previous Connection

The RS485 cable connects to the smart meter using the **positive (A+)** and **negative (B−)** terminals. Some meters also provide a GND terminal.

Next Connection

The RS485 data is converted to USB using an **RS485-to-USB converter** because the **Industrial IoT Gateway** (Raspberry Pi 5) does not have a native RS485 interface. The converter allows the Raspberry Pi 5 to receive **Modbus RTU** data through its USB port.

Why RS485?

Factories contain heavy electrical equipment that creates electrical noise. Normal USB communication becomes unreliable.

RS485 supports:

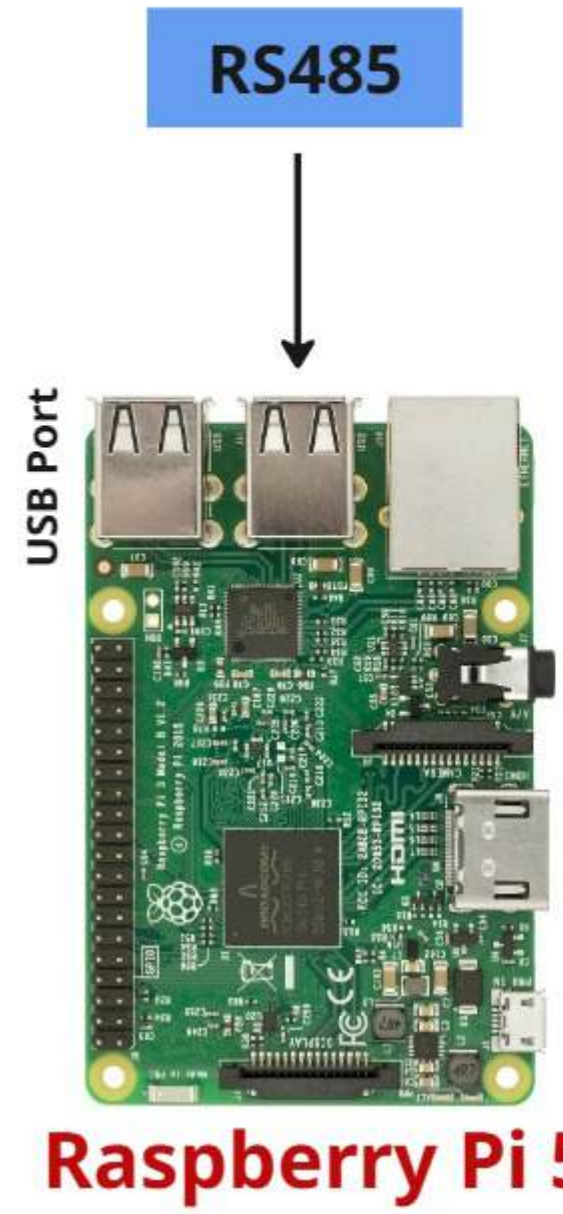
- Long-distance communication (**up to 1200 m**)
- High **noise immunity**
- Multiple devices on one communication bus
- **Reliable industrial** networking

Communication Protocol

The system uses the **Modbus RTU communication** protocol. Below is an example of a data packet.

Voltage = 415V
Current = 96A
Power = 52kW
Energy = 18,920kWh

Industrial IoT Gateway



Purpose

The Raspberry Pi acts as the central gateway of **EnergyOS** AI.
It collects data from:

- Main **Smart Meter**
- **ESP32** Machine Nodes

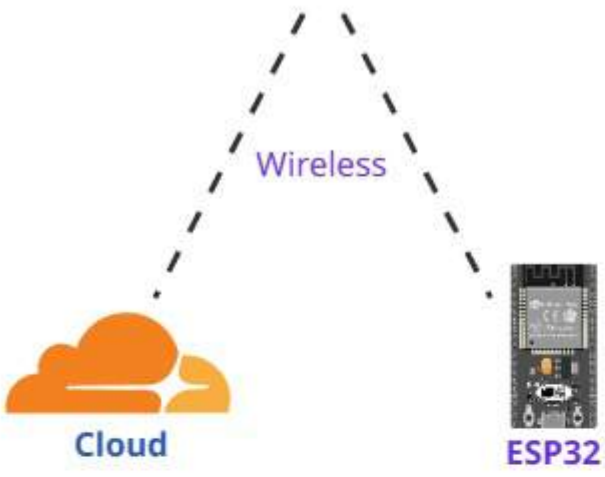
Then sends all data securely to the cloud.

Function

- Reads Modbus data
- Collects machine sensor data
- Converts **data into JSON**
- Adds timestamps
- Adds **Machine IDs**
- Buffers data if the internet is unavailable
- **Sends data** to the cloud

Example JSON

The gateway sends the collected data to the **cloud in JSON format**.
The backend uses the **machineId** field to identify the source device, such as the main meter or a **specific machine** (e.g., M-01 or M-02).




```
{
  "factoryId":"ABC01",
  "machineId":"MAIN",
  "voltage":415,
  "current":94,
  "power":51,
  "energy":18540
}
```

Previous Connection

The **Raspberry Pi** connects to two types of devices:

- The **RS485** network through a **USB port** using an RS485-to-USB converter.
- ESP32 devices over the **same Wi-Fi network** that connects the gateway to the cloud.

Next Connection

The gateway collects data from both the **RS485** network and the **ESP32** devices, then sends it to the cloud using **MQTT** or HTTPS over Wi-Fi.

Machine Monitoring

Machine

CT-Clamp

ESP32

Purpose

The Main Smart Meter only measures the total factory electricity.
To identify which machine consumes how much energy, every important machine is equipped with a monitoring node.

Components

- CT Clamp
- ESP32

CT Clamp Function

The CT Clamp is placed around the machine's power cable.
It measures the electrical current without cutting the wire.
It is **safe** and **non-invasive**.

ESP32 Function

ESP32 receives current measurements from the monitoring circuit.
It:

- **Reads sensor** values
- Calculates power (if voltage is known)
- Assigns Machine ID
- **Sends data** wirelessly

Data Collected

- Machine ID
- Current
- Estimated Power
- Temperature (optional)
- Vibration (optional)
- Running Status

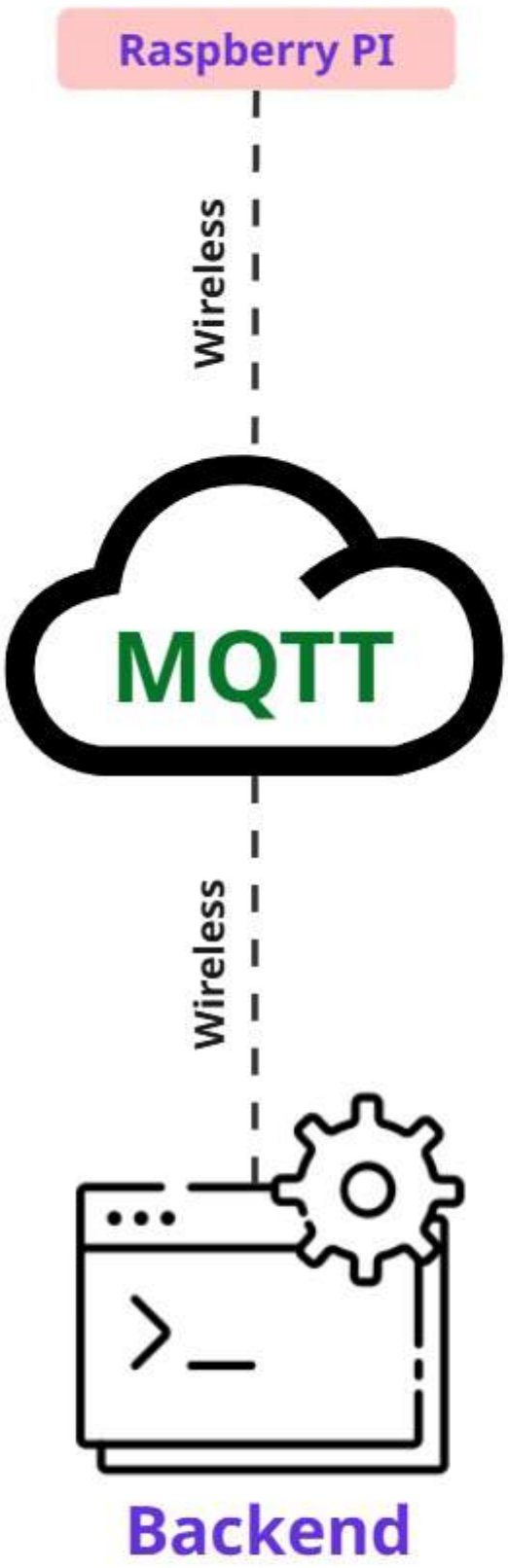
Previous Connection

Each machine is connected to a **CT Clamp**, which is connected to an **ESP32**. The ESP32 sends the data **wirelessly** to the cloud over Wi-Fi.

Next Connection

The ESP32 collects data from **Machines**, then sends it to the cloud using **MQTT** or HTTPS over Wi-Fi.

Cloud Communication (MQTT / HTTPS)



Purpose

After collecting all sensor data, the **Raspberry Pi** sends all the data to the cloud.

HTTPS Function

HTTPS uses standard **REST API** communication. It is simple to implement, suitable for MVPs, and **easier to integrate**. However, it is generally less efficient than MQTT for IoT applications.

MQTT Function

- Designed for IoT.
- Lightweight.
- Fast.
- Reliable.
- Supports thousands of devices.

Data Sent

```
{
 "gatewayId":"GW01",
 "machineId":"M03",
 "power":14.2,
 "current":22.5,
 "timestamp":"..."
}
```

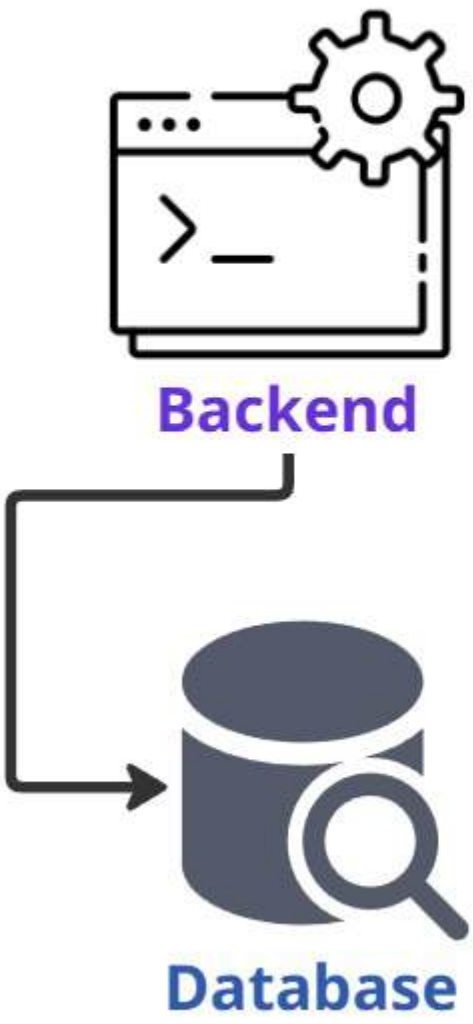
Previous Connection

It connects wirelessly to the Raspberry Pi 5 **IoT Gateway** over the same Wi-Fi network. It collects data from the **Main Smart Meter** and all machines, then **sends** the data to the cloud.

Next Connection

The Raspberry Pi publishes machine and meter data to the cloud using **MQTT.** (MQTT is a protocol, not the collector.)

Backend Processing & Data Storage



Purpose

The backend receives all **incoming sensor data** from the Main Smart Meter and machines, then stores it for analysis by the **AI engine.**

Backend Responsibilities

- Validate data
- Authenticate devices
- Store readings
- Detect missing data
- Trigger alerts
- Generate reports

Database Stores

- Factory
- Machine
- Sensor
- Energy Reading
- Alerts
- Reports
- Users

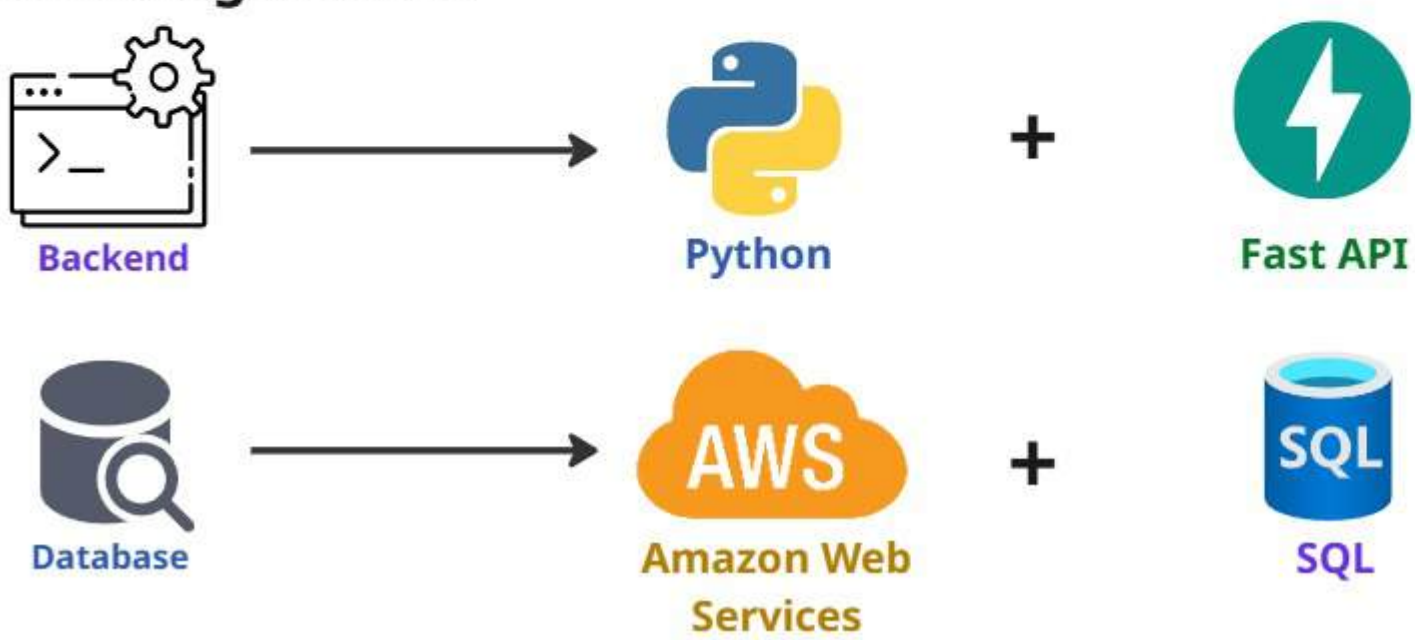
Previous Connection

It connects to the cloud wirelessly over Wi-Fi. It receives data from the Main **Smart Meter** and all **machines**, then stores it in the **database for AI analysis.**

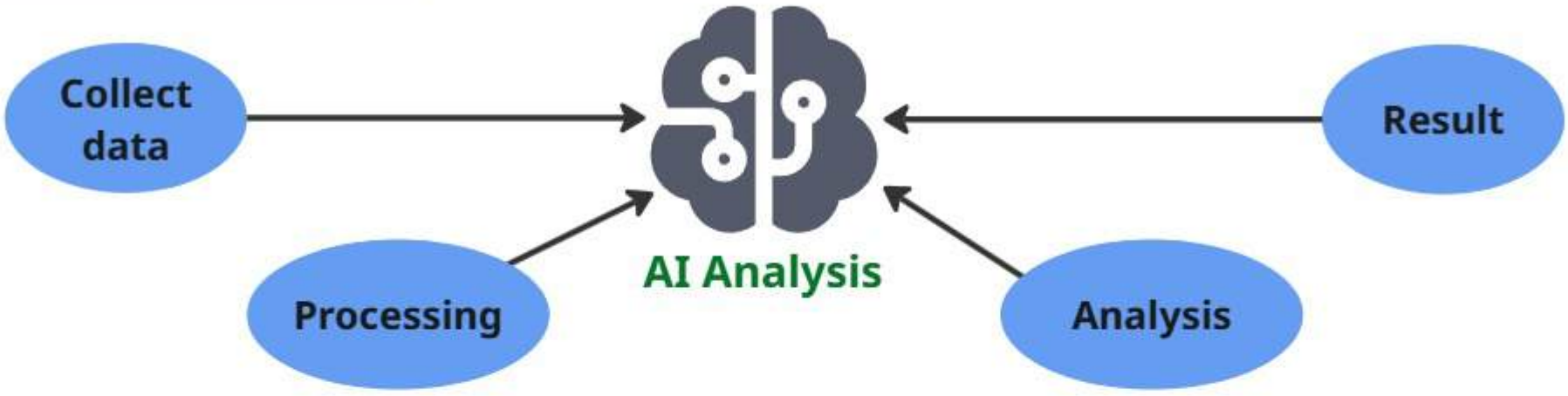
Next Connection

The **database stores** all collected machine and meter data. The **AI system** connects to the database and analyzes the collected data.

Technologies Used



AI Analysis Engine



Purpose

This is the **brain** of EnergyOS.
Instead of only displaying numbers, **AI converts data** into actionable recommendations.

Previous Connection

The AI model is connected to the **database**, where all data from the Main Smart Meter and machines is stored.

AI Tasks

- Predict monthly electricity bill
- **Detect abnormal energy** usage
- Compare machine efficiency
- Identify idle machines
- Forecast peak demand
- **Recommend** energy-saving actions
- Estimate carbon emissions
- Suggest preventive maintenance

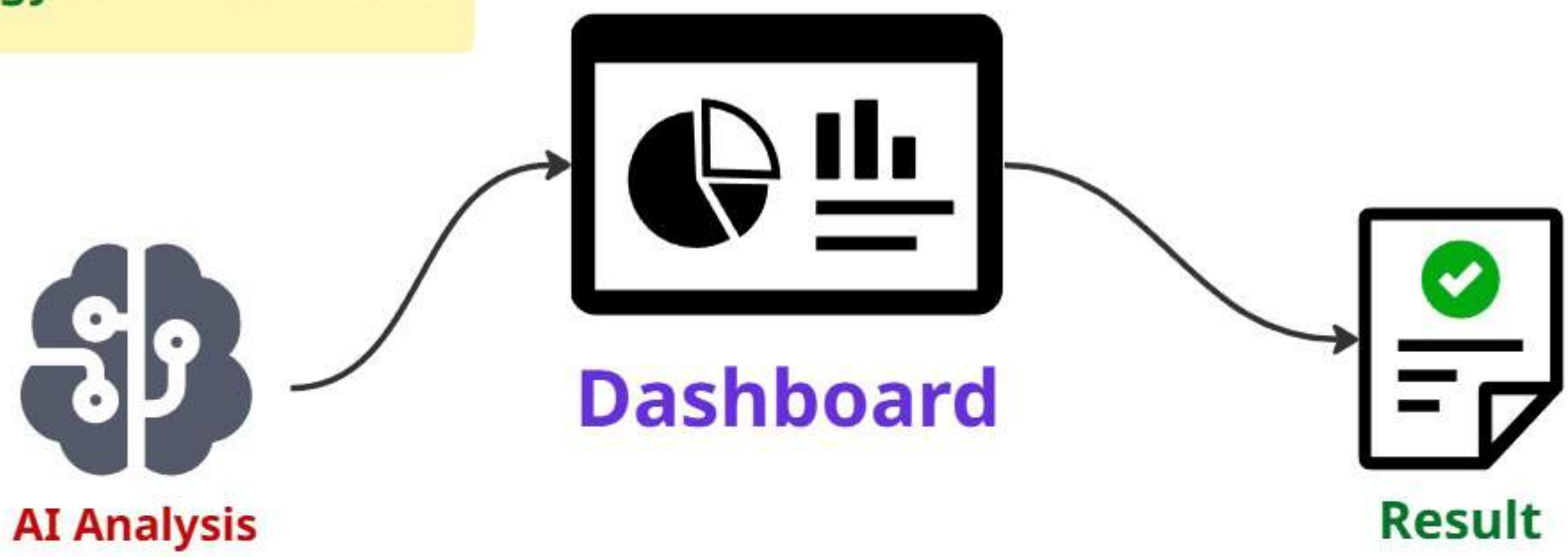
Example Insight

- Motor 1 is **consuming 18% more power** than usual. This may indicate bearing wear or overload.
- Boiler started earlier than scheduled today, increasing electricity cost.
- If you delay Compressor-2 by 15 minutes, this month's demand charge could **reduce by ₹18,000**.

Next Connection

The **AI engine** is connected to the **frontend** dashboard**.** It analyzes industrial electricity usage using data from the database and displays insights, **recommendations**, and **alerts** on the dashboard.

EnergyOS AI Dashboard



Purpose

The **Dashboard presents** all information in a simple, visual format for factory managers. Instead of reading raw electrical values, users immediately understand where **energy** is being consumed and what **actions** to take.

Information Displayed

Live Monitoring
1. Factory Power
2. Machine Power
3. Voltage
4. Current

Reports
1. Daily Energy Report
2. Weekly Report
3. Monthly Report
4. PDF Export

AI Insights
1. Energy Waste Alerts
2. Peak Demand Alerts
3. Monthly Bill Prediction
4. Maintenance Recommendations

Charts
1. Real-Time Power Graph
2. Machine Comparison
3. Energy Trend
4. Peak Demand Timeline

Customer Benefits

The Dashboard enables factory managers to:

- Monitor **energy consumption** in real time.
- Identify inefficient machines.
- **Reduce electricity costs** through AI recommendations.
- Improve operational efficiency.
- Make data-driven decisions instead of relying only on monthly electricity bills.

Demo Link :- energyos-ai.vercel.app