IoT System Design (Industrial Predictive Maintenance)

Our IoT application is an industrial predictive maintenance system designed to monitor a machine during operation and predict failure risk before a breakdown occurs. The machine is equipped with sensors that capture key operating signals. In our dataset, the main sensor measurements are air temperature, process temperature, rotational speed, torque, and tool wear. These signals represent the machine's condition and workload and are commonly used in manufacturing to detect early signs of abnormal behavior.

Data Collection and Sampling

In a real deployment, the sensors would collect readings at a fixed interval (for example, every 1 minute). The dataset does not include timestamps, so we treat tool wear (minutes) as a practical proxy for time progression. This allows us to structure the data as sequential observations and supports time-aware modeling for the time-series requirement.

Edge Layer and Communication

After data collection, sensor readings would be sent to an edge device (embedded controller or industrial gateway) located near the machine. The edge device performs lightweight processing such as filtering noise, validating ranges, and flagging simple anomalies (for example, sudden spikes in torque or temperature).
The edge device then sends the cleaned data to the cloud server using a lightweight messaging method such as **MQTT** or **HTTP/REST**, depending on the site network. Connectivity could be via Ethernet/Wi-Fi in a lab environment or industrial networking in a real factory. The data flows in the following sequence:

Sensors → Edge Device → Secure Communication Channel → Cloud Server → Database → Machine Learning Model → Dashboard.

Cloud Server Storage and Processing

On the cloud server, incoming data is stored in a database designed for ongoing sensor records. A reasonable architecture is either:

- a **relational (SQL) database** for structured tables and reporting, or
- a **time-series/NoSQL style storage** for high-frequency sensor streams.

This cloud layer also supports data cleaning logs, long-term storage for trend analysis, and model output storage (predictions + flags) so the dashboard can display both raw signals and ML results.

Machine Learning Inference and Retraining

The cloud server runs machine learning in two ways:

- **Batch inference** (e.g., run predictions every hour/day on the latest data)

- **Near real-time inference** (predict as each new minute of sensor data arrives).

For this project, we use:

1. a time-series regression approach (using tool wear as the time proxy) to model future behavior/trends, and

2. a deep learning feedforward neural network to classify machine state (failure vs. normal).

A simple retraining cycle is scheduled, for example, weekly or monthly, using new accumulated data. If performance drops, “model drift”, the model is retrained and redeployed on the cloud server.

Security and Access Control

Because industrial IoT data can be sensitive, the system assumes basic security controls:

- data is encrypted in transit between edge and cloud server

- API keys or device credentials are required for device-to-cloud communication

- role-based access is used so only authorized staff can view dashboards and predictions

End-User Dashboard and Decision Support

The final outputs are presented to the maintenance team through a dashboard. The dashboard shows key sensor trends, highlights abnormal patterns, and displays model predictions (failure risk / classification). This helps maintenance staff schedule inspections, prioritize maintenance tasks, and reduce unplanned downtime by acting proactively rather than reacting after a breakdown.