

# Smart AI-Powered Multi-Sensor Vibration and Environmental Monitoring System

## Project Contributors

- Mouhib Ben Salah
- Oussema Rhim
- Aziz Dakhlaoui
- Fedi Ben Hessine

This presentation introduces a cutting-edge system that utilizes the ESP32 microcontroller, AI-based prediction, and multiple sensors to monitor vibration patterns, environmental factors, and potential anomalies in real-time. The system analyzes data locally using AI algorithms, providing real-time insights and triggering alerts for potential faults and equipment failures.



# Applications of the Vibration Monitoring System

## Industrial Use

Predictive maintenance for machinery to identify potential issues and schedule preventative maintenance, reducing downtime and increasing operational efficiency.

## Structural Health Monitoring

Detecting wear and tear in bridges, buildings, and towers to ensure structural integrity and prevent catastrophic failures.

# Key Features and Benefits

## 1 AI-Powered Vibration Analysis

Analyzes vibration data in real-time, classifying patterns into "Normal," "Warning," or "Critical" states, enabling timely interventions.

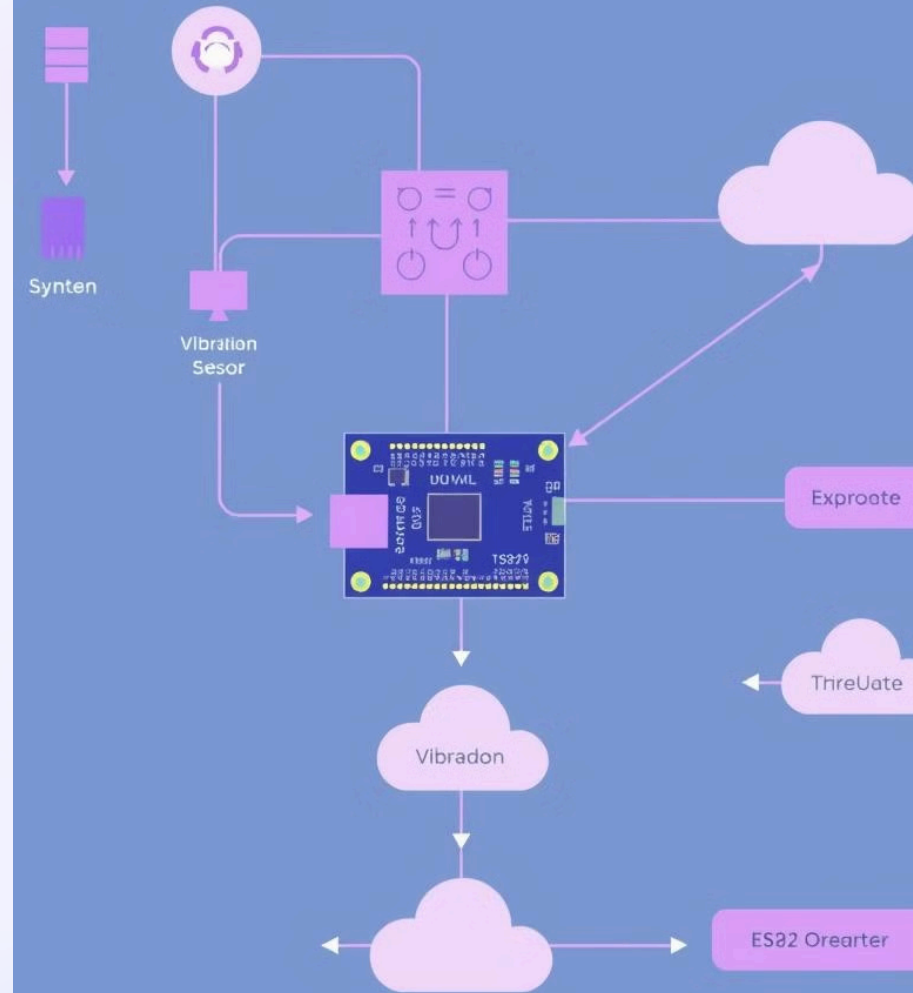
## 2 Multi-Sensor Integration

Combines vibration, temperature, humidity, pressure, and impact sensors to provide a comprehensive understanding of system health and environmental influences.

## 3 Local and Cloud Processing

Performs real-time AI inference on the ESP32 and logs data to a cloud platform for long-term analysis and visualization, providing valuable insights over time.

Allcoversto yl.apntoeer scngacer monitoring



# How It Works: Data Collection and AI Prediction



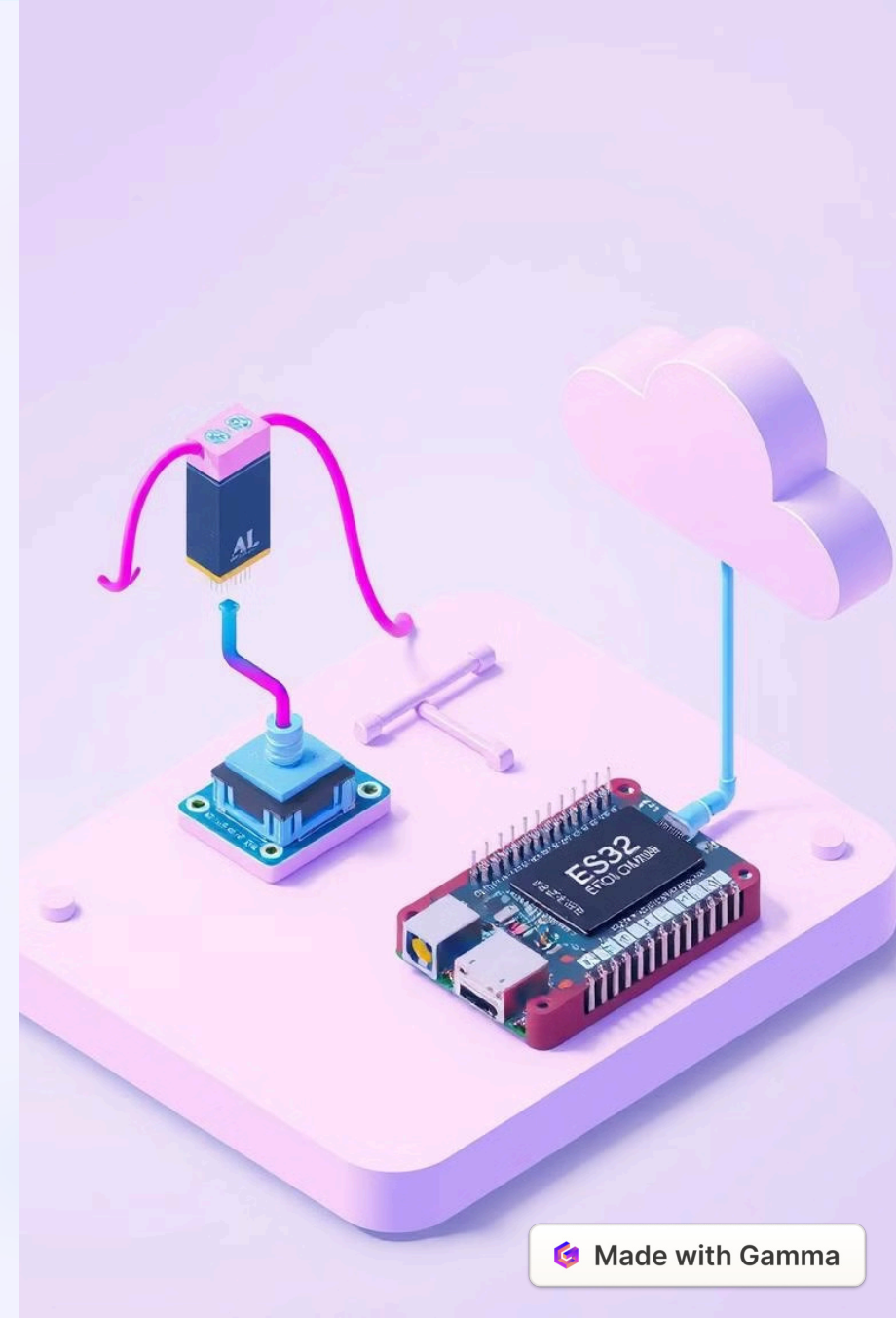
## Data Collection

The MPU6050 collects real-time vibration data, while DHT22, BMP180, and SW-420 sensors monitor environmental conditions and impact events.



## AI Prediction

The ESP32 analyzes vibration patterns using an optimized AI model (e.g., TensorFlow Lite Micro) to classify states as normal, warning, or critical.







# Alerts and Visualization

## Local Alerts

The ESP32 triggers local alarms using a buzzer and LEDs to indicate critical events and alert nearby personnel.

1

2

## Remote Notifications

Sends vibration trends and classification results to a cloud dashboard, enabling remote monitoring and providing valuable insights.



# Hardware Components and Cost

Component	Purpose	Estimated Cost (TND)
ESP32	Core controller	Already owned
MPU6050	Vibration/acceleration sensor	~15
DHT22	Temperature and humidity sensor	~15
BMP180	Pressure sensor	~15
SW-420 or Piezo Sensor	Sudden shock/impact detection	~10
Buzzer + LEDs	Local alert system	~5
SD Card Module	Offline data logging	~10
Misc. (wires, resistors, etc.)	Connections	~5

# Building the System: Steps to Success

1

## Hardware Setup

Connect the MPU6050, environmental sensors, impact sensor, local alert system, and SD card module to the ESP32.

2

## AI Model Development

Collect training data for different vibration states, train an AI model (e.g., TensorFlow or Scikit-learn), and optimize for ESP32 deployment.

3

## Code Development

Write code in Arduino IDE or PlatformIO to collect data, perform AI inference, trigger alerts, and manage offline logging.

4

## Cloud Integration

Use platforms like ThingSpeak, Blynk, or Node-RED for real-time visualization, configurable alerts, and long-term data analysis.



# Outcome and Future Upgrades

## 1

### Early Fault Detection

Provides early warning signs of potential equipment failures, enabling preventative maintenance and reducing downtime.

## 2

### Improved Reliability

Increases the reliability and longevity of machinery by identifying and addressing potential issues before they escalate.

## 3

### Actionable Insights

Generates actionable insights for maintenance teams, leading to more informed decisions and optimized maintenance strategies.

