

# Instructions for EE4216 Project

---

## Project instructions:

1. This is a group project with 3 students each team. You should form your team and sign up in the Canvas project grouping.
2. Each team pick **ONE** project from the EE4216 Project List attached. Each team needs to design and implement a smart IoT system for a specific purpose which are described in the project descriptions and objectives. The implementation should include multiple sensor nodes, data storage and visualization and user access interface.
3. If some extra sensors are needed, please propose to the instructor by the Friday of Week 8.

## Project Assessments:

1. The weightage of the project is 30% of the final marks.
2. The assessment includes a demonstration and a project report. Their weightages are 60% and 40%.

Demonstration (60%)		Project report (40%)	
Your project implementation should encompass the fundamental functions of the developed system. The system should constitute a comprehensive IoT setup that includes multiple sensor nodes, data visualization and user interface. Incorporating additional features is encouraged.		The project report should cover the problem statement, design, implementation, results analysis, and conclusion. The source code must be thoroughly documented to ensure readability.	
System functions (50%)	Extra features (10%)	Reports writing (25%)	Code and documentation (15%)

3. The demonstration will be conducted in Week 13 during the project session (one lecture sessions may be used for demonstration if time is not sufficient). The report submission deadline is 23:59 16 November 2024.

# EE4216 Project List

(One project for a team)

## A. Smart Home Environmental Monitoring System

### Descriptions

Design a Smart Home Security and Environment Monitoring System that tracks various environmental parameters (temperature, humidity, motion, and gas detection) and provides security alerts. This system will use ESP32-S3 DevKits, which will multitask to handle various sensors, connect to Wi-Fi, and send data to an online server. The system will also include real-time alerts for unauthorized entry or hazardous gas levels, allowing users to remotely monitor and control their home environment and security.

### Project Objectives:

1. Environmental Monitoring: Measure temperature, humidity, and gas levels within a home.
2. Security Monitoring: Detect motion and unauthorized entry using PIR sensors.
3. Multitasking: Implement multitasking on the ESP32 to handle multiple sensors and processes simultaneously.
4. Wi-Fi Connectivity: Use the ESP32's Wi-Fi capabilities to connect to the home network and communicate with an online server.
5. Data Logging and Visualization: Store sensor data on a cloud server and provide real-time visualization through a web or mobile interface.
6. Real-Time Alerts: Send alerts for security breaches or hazardous conditions (e.g., smoke, gas leaks).

## **B. Smart Plant Monitoring and Irrigation System**

### **Descriptions**

Design a Smart Plant Monitoring and Irrigation System that helps users maintain optimal growing conditions for their indoor or outdoor plants. This system will monitor soil moisture, temperature, humidity, and light intensity. The ESP32-S3 DevKits will interface with various sensors, connect to Wi-Fi, and communicate data to an online server. When the soil moisture level falls below a predefined threshold, the system will automatically activate a water pump to irrigate the plant.

### **Project Objectives**

1. **Soil Moisture Monitoring:** Measure the moisture level of the soil to determine if irrigation is needed.
2. **Environmental Monitoring:** Measure the temperature, humidity, and light intensity around the plant.
3. **Automated Irrigation:** Automatically water the plant when soil moisture drops below a certain level.
4. **Wi-Fi Connectivity:** Use the ESP32's Wi-Fi capabilities to send sensor data to an online server and receive control commands.
5. **Data Visualization:** Provide real-time data visualization and historical trends through a web or mobile interface.
6. **Remote Control and Alerts:** Allow remote monitoring and control via a web dashboard or mobile app and send alerts when necessary.

## **C. Smart Energy Monitoring and Control System**

### **Description**

Create a Smart Energy Monitoring and Control System that tracks energy consumption of household appliances in real-time. The system will use ESP32-S3 DevKits to interface with current and voltage sensors to measure power usage. It will connect to Wi-Fi and send data to an online server for real-time monitoring, logging, and analysis. Users can also remotely control connected appliances, turning them on or off to manage energy usage effectively.

### **Project Objectives**

1. **Energy Monitoring:** Measure the real-time power consumption of household appliances.
2. **Data Logging:** Store and analyze energy usage data over time to identify usage patterns.
3. **Remote Control:** Allow users to turn appliances on or off remotely to save energy.
4. **Wi-Fi Connectivity:** Use the ESP32's Wi-Fi capabilities to connect to a home network and communicate with an online server.
5. **Data Visualization:** Provide a user-friendly web or mobile interface to display energy usage data.
6. **Notifications:** Send alerts when energy consumption exceeds predefined thresholds.