



# Weather Tracker IoT

## Group C

21011864 Jeongmin Park

21011910 Jiho Park

22011901 Jiyun Yang

# Weather Tracker IoT

The system receives weather data from KMA(Korea Meteorological Administration) through its API Hub, simulates a custom GPS location, and generates virtual sensors such as temperature, humidity, and light. All collected information will be on the serial monitor in VSCode.

## Key Features

- Fetch and parse weather data in XML format from the KMA API
- Simulate GPS coordinates for location-based weather display
- Map weather data to virtual sensors
- Visualize all information on the Serial Monitor



# Details

Below are the technologies we will use in the project

## 01. VSCode + Wokwi

Software-based IoT  
using VSCode and Wokwi,  
a virtual ESP32 simulator

## 02. HTTPClient

Get Data from KMA APIhub  
<https://apihub.kma.go.kr>

## 03. Sensors

### - GPS

Generate a random  
location in Korea and  
convert coordinates to  
city and district names

### - LEDs

Red LED → Sunny

Blue LED → Rainy

Yellow LED → Cloudy

### - DHT22

Temperature & Humidity

## 04. Serial Monitor


### - Display weather Info


Location

Weather

Temperature

 Weather Tracker IoT

 Location: Seoul, Gangnam-gu

 Weather: Clear Sky

 Temperature: 22.5 °C

# Members' Tasks

Data Communication & Parsing / Virtual Sensors & GPS Simulation / Integration & Serial Output

## Jeongmin Park : Data Parsing

- Use ESP32 HTTPClient.h to request KMA API data
- Parse the XML string and extract weather-related information

## Jiho Park : Sensors & GPS

- Make a random coordinate in Korea
- Convert GPS coordinates to address format
- Send structured data to the main program

## Jiyun Yang : Serial Output

- Merge data codes and sensor codes
- Organize serial monitor output formats
- Oversee project progress and prepare final presentation

# Timeline

Timeline to complete this project (Week 9 - Week 16)

## week 9 - 10

- Decide on the project
- Set up the environment
- Get KMA API data

## week 11 - 12

- Parse and extract weather-related XML data
- Generate simulated GPS coordinates

## week 13 - 14

- Write codes for virtual sensors (LEDs, DHT22)

## week 15 - 16

- Design the Serial Monitor output
- Debug and fix errors after running the program
- Prepare for the final presentation