# AWS IoT Backend based Soil Quality Tracker

2015-12-18
Final Presentation

Team Lustitia
Members : 고성빈, 권현석, 함지웅

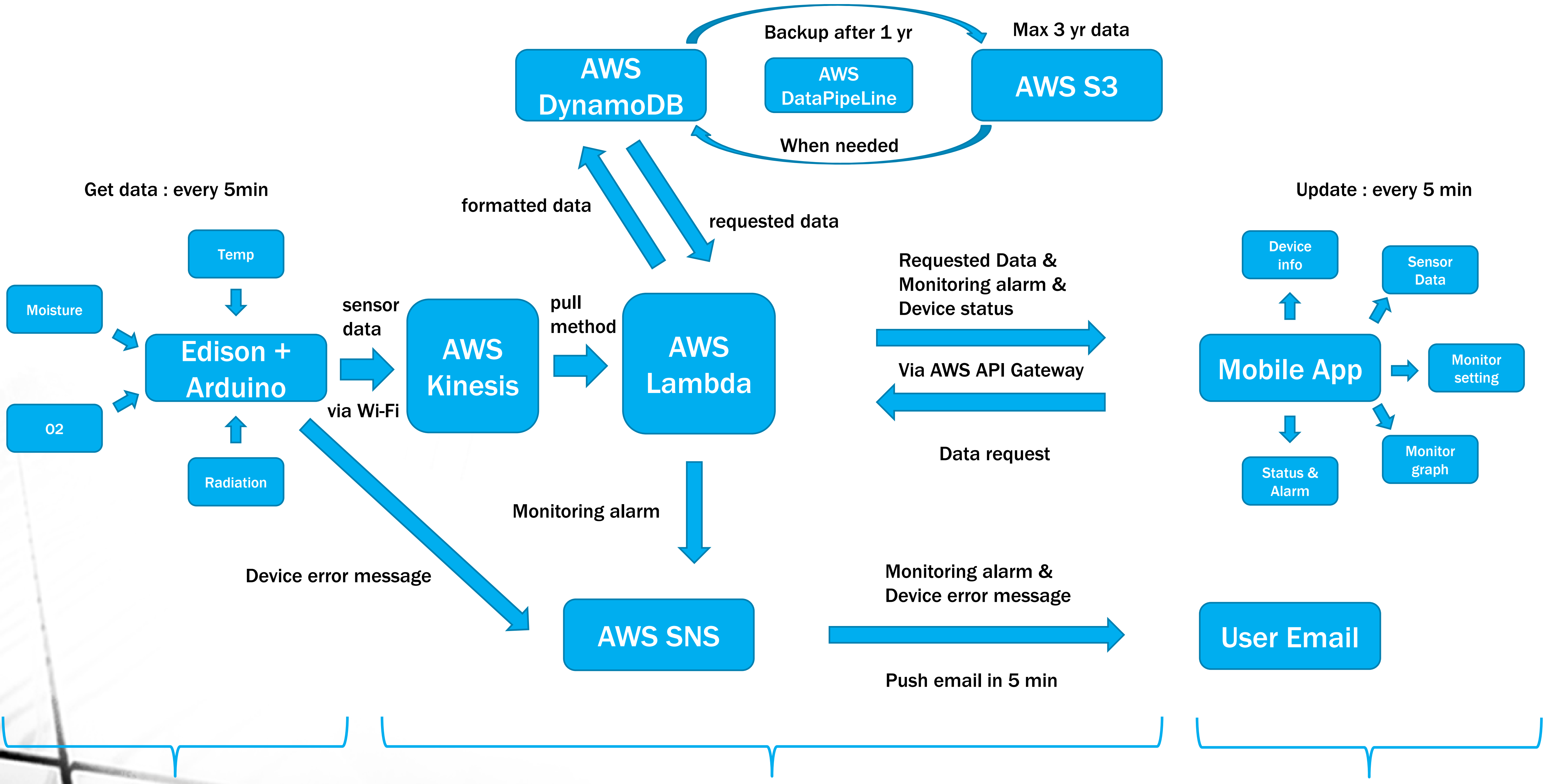
## Contents

## Project overview

Our project aims…

"AWS IoT Backend based Soil Quality Tracker System,
which will use the flexibility & scalability of AWS
to provide users with useful information."

AWS IoT Backend를 이용한 토양 품질 트래커는
Intel Edison과 Arduino 보드를 사용해 수집한 정보를
다양한 AWS 서비스를 사용하는 Backend를 통해
저장하고 원하는 방식으로 가공하여
사용자에게 토양에 대한 여러 정보를 제공하는 것을 목적으로 한다.

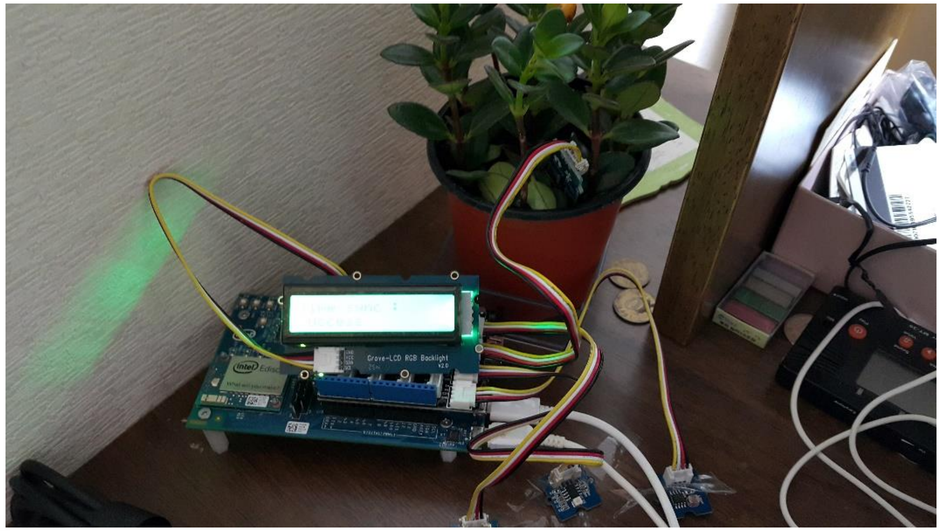


Edison + Arduino

AWS Services

Mobile App + User

## Detailed Architecture – Intel Edison + Arduino Board



Sensors : Temperature, Moisture, UV, Light

LCD : Grove LCD RGB Display

Data Get Interval : 5min

Error Indication : Sensor, Wi-Fi, AWS, Time Sync

## Detailed Architecture – Intel Edison + Arduino Board (cont'd)

Error indication algorithm :

1) Sensor : checking abnormal Sensor values
2) Wi-Fi : checking Wi-Fi connection
3) Time Sync : Connection with NTP server
4) AWS : AWS initialization, service connection

Data flow :

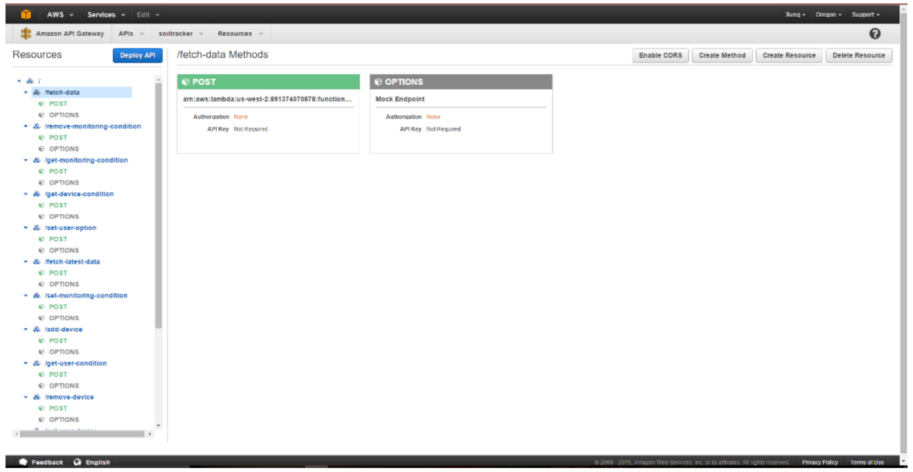
1) Since sensors take some time to get stable, take values every 5 sec
2) Every 5 min, take current values and post them to Kinesis with appropriate format
3) If error occurs, set error indicator variable to specified error's code.

LCD View :

1) While setup, show setup status.
2) While monitoring, display current error status.
3) If no change for 30 secs, turn off light.

# Detailed Architecture – AWS

API Gateway



Resources

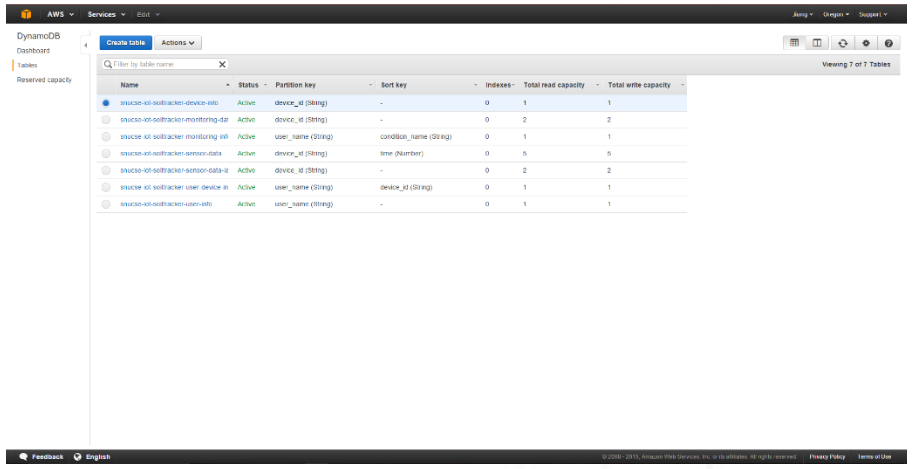
/fetch-data Methods

- /
  - /fetch-data
    - POST
    - OPTIONS
  - /remove-monitoring-condition
    - POST
    - OPTIONS
  - /get-monitoring-condition
    - POST
    - OPTIONS
  - /get-device-condition
    - POST
    - OPTIONS
  - /set-user-option
    - POST
    - OPTIONS
  - /fetch-latest-data
    - POST
    - OPTIONS
  - /set-monitoring-condition
    - POST
    - OPTIONS
  - /add-device
    - POST
    - OPTIONS
  - /get-user-condition
    - POST
    - OPTIONS
  - /remove-device
    - POST
    - OPTIONS

POST: arn:aws:lambda:us-west-2:891374070878:function...
Authorization None
API Key Not Required

OPTIONS: Mock Endpoint
Authorization None
API Key Not Required

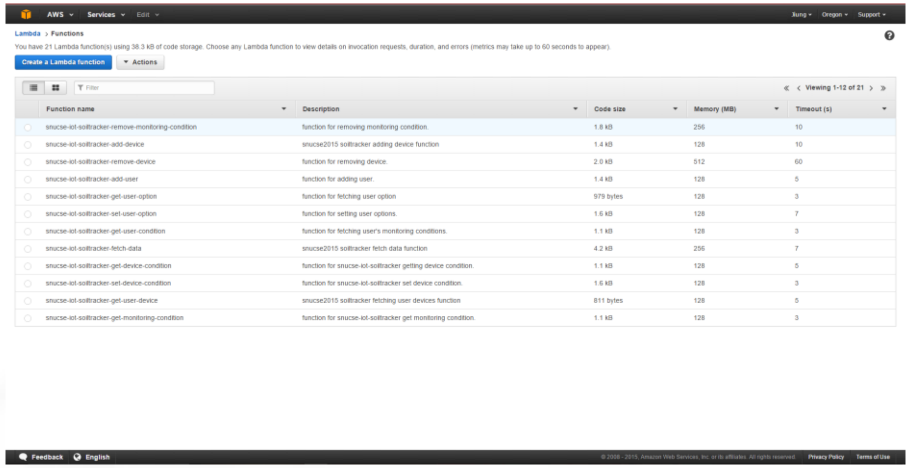
DynamoDB



| Name | Status | Partition key | Sort key | Indexes | Total read capacity | Total write capacity |
|---|---|---|---|---|---|---|
| snucse-iot-soiltracker-device-info | Active | device_id (String) | - | 0 | 1 | 1 |
| snucse-iot-soiltracker-monitoring-dat | Active | device_id (String) | - | 0 | 2 | 2 |
| snucse-iot-soiltracker-monitoring-inf | Active | user_name (String) | condition_name (String) | 0 | 1 | 1 |
| snucse-iot-soiltracker-sensor-data | Active | device_id (String) | time (Number) | 0 | 5 | 5 |
| snucse-iot-soiltracker-sensor-data-la | Active | device_id (String) | - | 0 | 2 | 2 |
| snucse-iot-soiltracker-user-device-in | Active | user_name (String) | device_id (String) | 0 | 1 | 1 |
| snucse-iot-soiltracker-user-info | Active | user_name (String) | - | 0 | 1 | 1 |

Viewing 7 of 7 Tables

Lambda



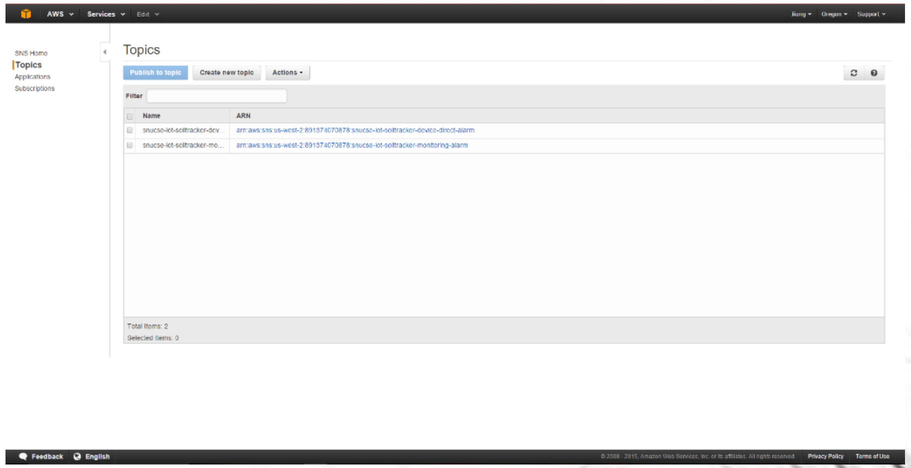
Lambda > Functions

You have 21 Lambda function(s) using 38.3 kB of code storage. Choose any Lambda function to view details on invocation requests, duration, and errors (metrics may take up to 60 seconds to appear).

Viewing 1-12 of 21

| Function name | Description | Code size | Memory (MB) | Timeout (s) |
|---|---|---|---|---|
| snucse-iot-soiltracker-remove-monitoring-condition | function for removing monitoring condition. | 1.8 kB | 256 | 10 |
| snucse-iot-soiltracker-add-device | snucse2015 soiltracker adding device function | 1.4 kB | 128 | 10 |
| snucse-iot-soiltracker-remove-device | function for removing device. | 2.0 kB | 512 | 60 |
| snucse-iot-soiltracker-add-user | function for adding user. | 1.4 kB | 128 | 5 |
| snucse-iot-soiltracker-get-user-option | function for fetching user option | 979 bytes | 128 | 3 |
| snucse-iot-soiltracker-set-user-option | function for setting user options. | 1.6 kB | 128 | 7 |
| snucse-iot-soiltracker-get-user-condition | function for fetching user's monitoring conditions. | 1.1 kB | 128 | 3 |
| snucse-iot-soiltracker-fetch-data | snucse2015 soiltracker fetch data function | 4.2 kB | 256 | 7 |
| snucse-iot-soiltracker-get-device-condition | function for snucse-iot-soiltracker getting device condition. | 1.1 kB | 128 | 5 |
| snucse-iot-soiltracker-set-device-condition | function for snucse-iot-soiltracker set device condition. | 1.6 kB | 128 | 3 |
| snucse-iot-soiltracker-get-user-device | snucse2015 soiltracker fetching user devices function | 811 bytes | 128 | 5 |
| snucse-iot-soiltracker-get-monitoring-condition | function for snucse-iot-soiltracker get monitoring condition. | 1.1 kB | 128 | 3 |

SNS



Topics

| Name | ARN |
|---|---|
| snucse-iot-soiltracker-dev... | arn:aws:sns:us-west-2:891374070878:snucse-iot-soiltracker-device-direct-alarm |
| snucse-iot-soiltracker-mo... | arn:aws:sns:us-west-2:891374070878:snucse-iot-soiltracker-monitoring-alarm |

Total Items: 2
Selected Items: 0

## Detailed Architecture – AWS (cont'd)

AWS Lambda

1) All functions for processing data, management of data, device and user management
2) Center for Device – AWS – Mobile App system.
3) Function codes are written in Node.js
4) Total 15 functions are implemented

AWS Dynamo DB

1) All data for sensors, devices, users, monitoring conditions
2) Total 7 DB tables are used

## Detailed Architecture – AWS (cont’d)

AWS API Gateway

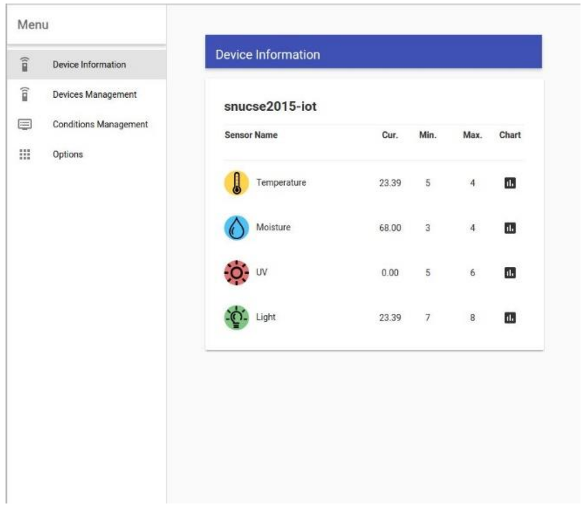
1) API for Mobile app to use lambda functions
2) Total 14 APIs are implemented.
3) Each API is in charge of each Lambda function.

AWS SNS

1) Topic publishing for alarms.
2) Total 2 Topics are implemented.
3) Alarms for device-direct alarms, sensor-monitoring alarms are provided.
4) When triggered, send email to subscribed users.

## Detailed Architecture – Mobile App

Current Info

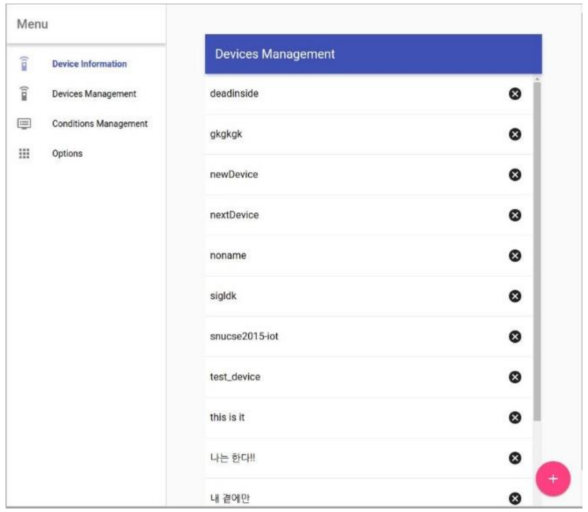


Device Information

**snucse2015-iot**

| Sensor Name | Cur. | Min. | Max. | Chart |
|---|---|---|---|---|
| Temperature | 23.39 | 5 | 4 | 📊 |
| Moisture | 68.00 | 3 | 4 | 📊 |
| UV | 0.00 | 5 | 6 | 📊 |
| Light | 23.39 | 7 | 8 | 📊 |

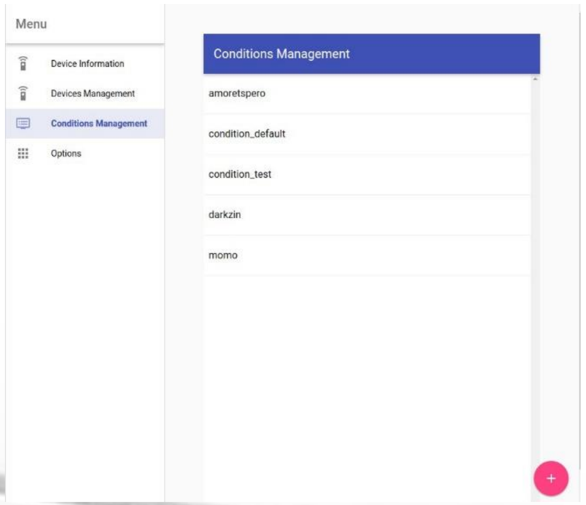
Device Management



Devices Management

- deadinside
- gkgkgk
- newDevice
- nextDevice
- noname
- sigldk
- snucse2015-iot
- test_device
- this is it
- 나는 한다!!
- 내 곁에만

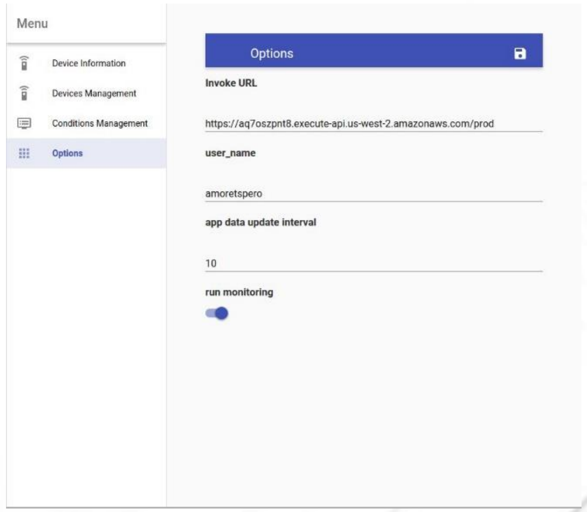
Monitoring Condition Management



Conditions Management

- amoretspero
- condition_default
- condition_test
- darkzin
- momo

Application Options



Options

**Invoke URL**

https://aq7oszpnt8.execute-api.us-west-2.amazonaws.com/prod

**user_name**

amoretspero

**app data update interval**

10

**run monitoring**

[✓ on]

## Detailed Architecture – Mobile App (cont'd)

Device Information

1) Current status of selected device
2) If no device is chosen, or no data is available, show error message
3) Clicking chart button will change display to graphical chart of selected sensor data.

Device Management

1) Add, Delete, Select device
2) Plus button will make new device
3) Cross(X) button will delete selected device.
4) Clicking device name will select that device for current device.

## Detailed Architecture – Mobile App (cont'd)

Conditions Management

1) Add, Delete, Select condition for current device.
2) Plus button will make new monitoring condition
3) Clicking condition name will show detailed information
4) Check button will set this condition as current device's condition
5) Pen button will show editing mode, where cross(x) button is there for deleting condition

Options

1) Manage API invoke URL, user name, app data update interval, monitoring(Y/N)
2) Devices, monitoring conditions changes with user name.
3) User can set update interval(number) and monitoring(Y/N).

## Schedule

| Work / Week | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Intel Edison, AWS Study | 🟩 | 🟩 | 🟩 |  |  |  |  |  |  |  |  |  |  |  |
| Spec Presentation |  |  | 🟩 |  |  |  |  |  |  |  |  |  |  |  |
| Simplified Ver. of prog. for Arduino Dev. |  |  | 🟩 | 🟩 | 🟩 | 🟩 |  |  |  |  |  |  |  |  |
| Simplified Ver. Of AWS IoT Backend Dev. |  |  | 🟩 | 🟩 | 🟩 | 🟩 |  |  |  |  |  |  |  |  |
| Mid-term Presentation |  |  |  |  |  | 🟩 |  |  |  |  |  |  |  |  |
| Finalize specification |  |  |  |  |  | 🟩 | 🟩 |  |  |  |  |  |  |  |
| Complete Ver. of prog. For Arduino Dev. |  |  |  |  |  |  |  | 🟩 | 🟩 | 🟩 | 🟩 |  |  |  |
| Complete Ver. of AWS IoT Backend Dev. |  |  |  |  |  |  |  | 🟩 | 🟩 | 🟩 | 🟩 |  |  |  |
| Mobile App Dev. |  |  |  |  |  |  |  |  |  | 🟩 | 🟩 |  |  |  |
| System Test |  |  |  |  |  |  |  |  |  |  |  | 🟩 | 🟩 |  |
| Demo, final improvements |  |  |  |  |  |  |  |  |  |  |  |  | 🟩 | 🟦 |
| Final Presentation |  |  |  |  |  |  |  |  |  |  |  |  |  | 🟦 |

DEMO

Thank You :)