

# Mini BuT

MBT

오상화, 김수은, 민태준, 박재효, 박종민



# CONTENTS

---

01  
프로젝트 수정 사항

02  
프로젝트 수행 내용

03  
향후 추진 계획



## 01. 프로젝트 수정 사항

### 1. 외부 미세먼지 데이터 수집 방식 추가

공공데이터 Open API + KMU Crowd Sensor Cloud

=> 외부 미세먼지 데이터의 대표성과 지역성을 확보

=> 프로젝트의 방향 : 실내 공기질의 전반적 관리 및 개선



실내 미세먼지 농도 관리 및 개선

## 01. 프로젝트 수정 사항

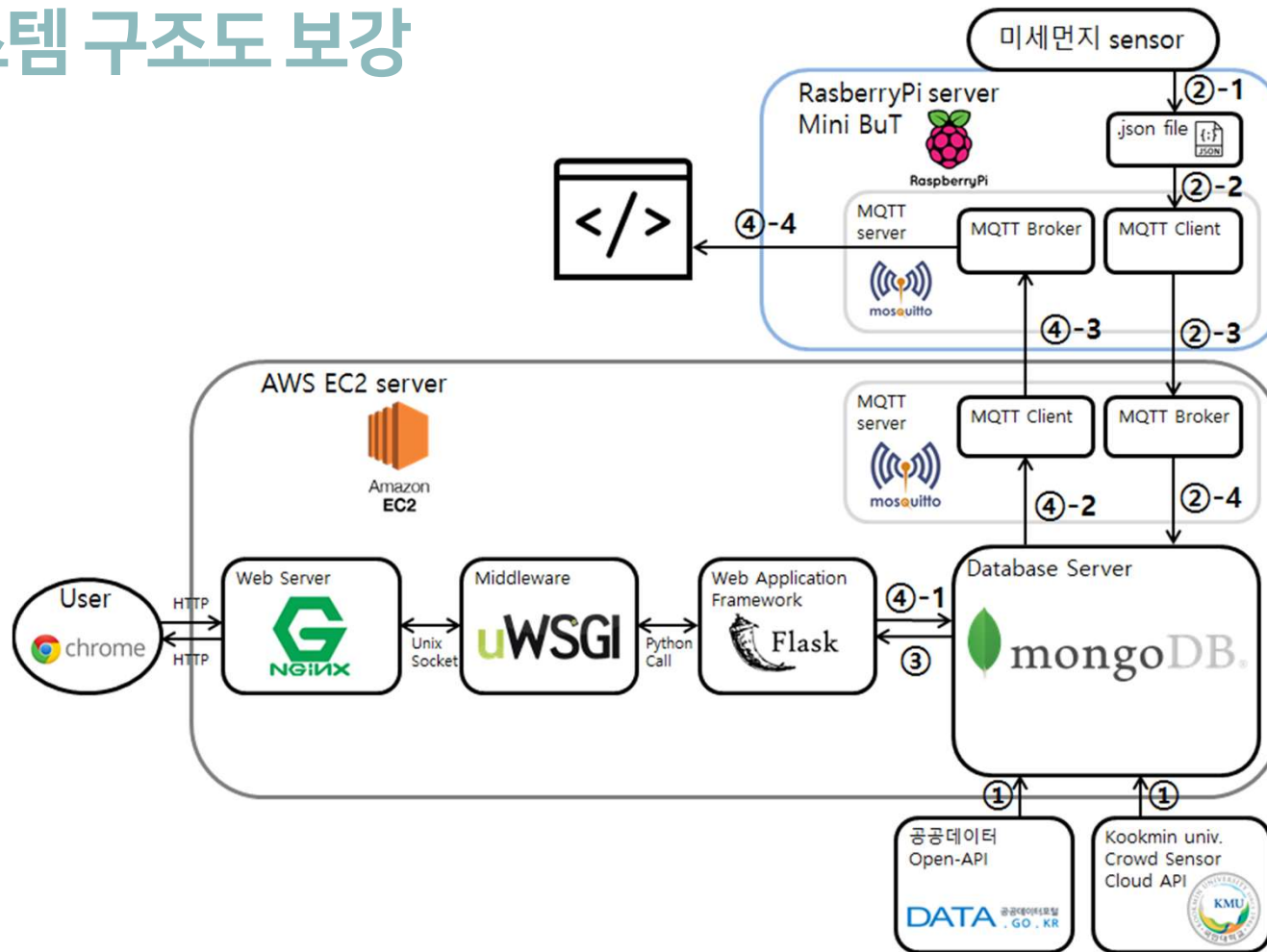
## 2. 프로젝트 시연 방향성 변경

IR 통신을 통한 창문의 제어 -> 시연용 웹 페이지 제작으로 대체

=> Mini BuT이 연동 기기를 얼마나 디테일하게 관리  
하는지를 더욱 직관적이고 가시적으로 확인 가능

## 01. 프로젝트 수정 사항

### 3. 시스템 구조도 보강



## 02. 프로젝트 수행 내용

### 1. Server 구현

서버 기본 환경 구성 완료

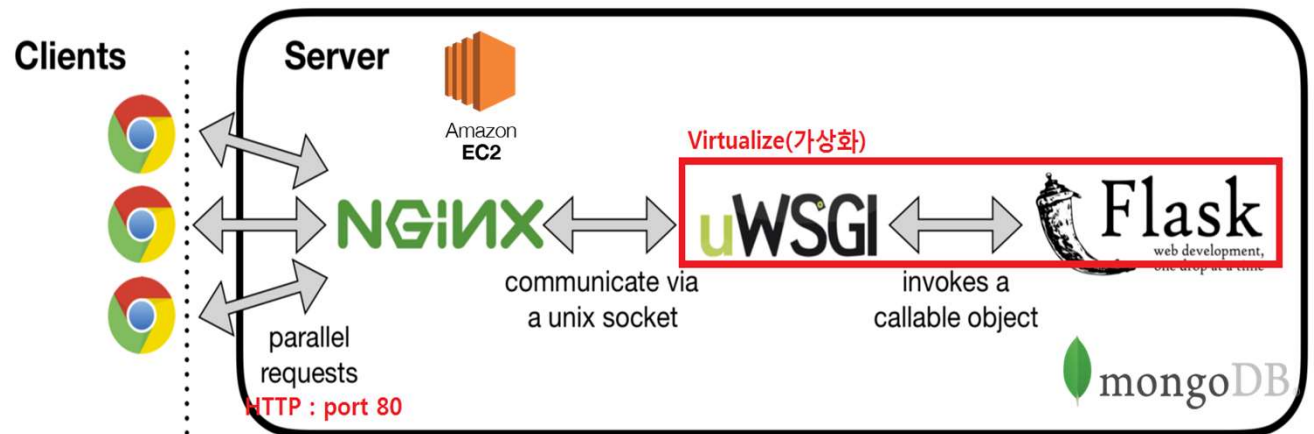
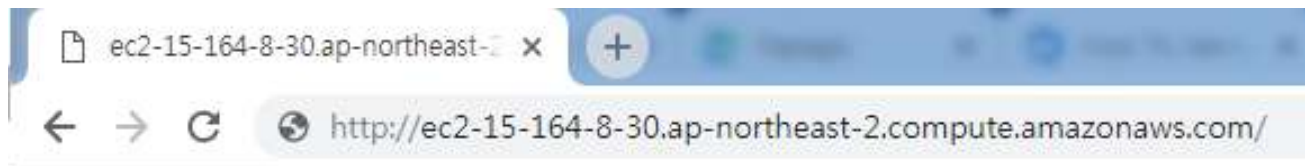
DB 설계 및 DB 서버 생성 완료

실외 데이터 수집 모듈 일부 구현 완료

서버와 라즈베리파이 간 MQTT 통신 환경 구현 완료

## 02. 프로젝트 수행 내용

### 1. Server 구현



## 02. 프로젝트 수행 내용

### 1. Server 구현

Databasename = dust

dust.recent

userid: 'abc'  
internaladdress:

lat: '111'  
lng: '122'

internaldata:

pm10: '10'  
pm25: '11'  
pm10grade: '1'  
pm25grade: '1'  
date: '2019-04-15 19:00:00'

externaladdress:

lat: '123'  
lng: '130'

externaldata:

pm10: '75'  
pm25: '50'  
pm10grade: '2'  
pm25grade: '2'  
date: '2019-04-15 19:00:00'

dust.internaldust

userid: 'abc'  
pm10: '10'  
pm25: '11'  
pm10grade: '1'  
pm25grade: '1'  
date: '2019-04-15 19:00:00'

...

dust.externaldust

pm10: '75'  
pm25: '50'  
pm10grade: '2'  
pm25grade: '2'  
date: '2019-04-15 19:00:00'

...

```
ubuntu@ip-172-31-25-41: ~  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
> db.internaldust.find().pretty()  
{  
  "_id" : ObjectId("5ca6e7f3e3727cdd8dd83b20"),  
  "userid" : "ABC",  
  "pm10" : 10,  
  "pm25" : 20,  
  "date" : "2019-04-05 14:00"  
}  
{  
  "_id" : ObjectId("5ca6e7f3e3727cdd8dd83b21"),  
  "userid" : "DEF",  
  "pm10" : 30,  
  "pm25" : 40,  
  "date" : "2019-04-05 15:00"  
}  
> db.externaldust.find().pretty()  
{  
  "_id" : ObjectId("5ca6ec40e3727cdd8dd83b22"),  
  "userid" : "ABC",  
  "pm10" : 100,  
  "pm25" : 200,  
  "date" : "2019-04-05 16:00"  
}  
{  
  "_id" : ObjectId("5ca6ec40e3727cdd8dd83b23"),  
  "userid" : "DEF",  
  "pm10" : 300,  
  "pm25" : 400,  
  "date" : "2019-04-05 17:00"  
}  
> □  
switched to db dust  
> db.recent.find().pretty()  
{  
  "_id" : ObjectId("5ca6fc74e3727cdd8dd83b24"),  
  "userid" : "ABC",  
  "internalAddress" : {  
    "lat" : 137,  
    "lng" : 147  
  },  
  "internalData" : {  
    "pm10" : 10,  
    "pm25" : 20,  
    "date" : "2019-04-05 14:00"  
  },  
  "externalAddress" : {  
    "lat" : 138,  
    "lng" : 148  
  },  
  "externalData" : {  
    "pm10" : 11,  
    "pm25" : 21,  
    "date" : "2019-04-05 14:00"  
  }  
}  
{  
  "_id" : ObjectId("5ca6fcd6e3727cdd8dd83b25"),  
  "userid" : "DEF",  
  "internalAddress" : {  
    "lat" : 222,  
    "lng" : 333  
  },  
  "internalData" : {  
    "pm10" : 32,  
    "pm25" : 32,  
    "date" : "2019-04-05 14:00"  
  },  
  "externalAddress" : {  
    "lat" : 222,  
    "lng" : 333  
  },  
  "externalData" : {  
    "pm10" : 23,  
    "pm25" : 23,  
    "date" : "2019-04-05 14:00"  
  }  
}
```



## 02. 프로젝트 수행 내용

### 1. Server 구현

```
kmusw@kmusw-ThinkPad-T440:~/바탕화면$ python3 api.py
```

```
data time : 2019-04-17 14:00
```

```
pm10 Value : 59
```

```
pm25 Value : 19
```

```
pm10 Grade : 2
```

```
pm25 Grade : 2
```

이전 db 값

```
{'_id': ObjectId('5c9d87767b02410ce0a6447e'), 'date': '2019-03-28 22:00', 'pm10value': '35', 'pm25value': '20', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5c9da0887d4339717f936bda'), 'date': '2019-03-29 13:00', 'pm10value': '45', 'pm25value': '18', 'pm10grade': '2', 'pm25grade': '1'}
{'_id': ObjectId('5ca4630d7d43390ef4f18238'), 'date': '2019-04-03 16:00', 'pm10value': '32', 'pm25value': '16', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5caffa777d43391058228aa7'), 'date': '2019-04-12 11:00', 'pm10value': '69', 'pm25value': '28', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5cb006f47d4339115d746b22'), 'date': '2019-04-12 12:00', 'pm10value': '55', 'pm25value': '29', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5cb019657d433913fa2a9627'), 'date': '2019-04-12 13:00', 'pm10value': '62', 'pm25value': '33', 'pm10grade': '2', 'pm25grade': '3'}
{'_id': ObjectId('5cb019707d433913fa2a9629'), 'date': '2019-04-12 13:00', 'pm10value': '62', 'pm25value': '33', 'pm10grade': '2', 'pm25grade': '3'}
```

이후 db 값

```
{'_id': ObjectId('5c9d87767b02410ce0a6447e'), 'date': '2019-03-28 22:00', 'pm10value': '35', 'pm25value': '20', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5c9da0887d4339717f936bda'), 'date': '2019-03-29 13:00', 'pm10value': '45', 'pm25value': '18', 'pm10grade': '2', 'pm25grade': '1'}
{'_id': ObjectId('5ca4630d7d43390ef4f18238'), 'date': '2019-04-03 16:00', 'pm10value': '32', 'pm25value': '16', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5caffa777d43391058228aa7'), 'date': '2019-04-12 11:00', 'pm10value': '69', 'pm25value': '28', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5cb006f47d4339115d746b22'), 'date': '2019-04-12 12:00', 'pm10value': '55', 'pm25value': '29', 'pm10grade': '2', 'pm25grade': '2'}
{'_id': ObjectId('5cb019657d433913fa2a9627'), 'date': '2019-04-12 13:00', 'pm10value': '62', 'pm25value': '33', 'pm10grade': '2', 'pm25grade': '3'}
{'_id': ObjectId('5cb019707d433913fa2a9629'), 'date': '2019-04-12 13:00', 'pm10value': '62', 'pm25value': '33', 'pm10grade': '2', 'pm25grade': '3'}
{'_id': ObjectId('5cb6c2977d43395aefef69b0'), 'date': '2019-04-17 14:00', 'pm10value': '59', 'pm25value': '19', 'pm10grade': '2', 'pm25grade': '2'}
```

## 02. 프로젝트 수행 내용

### 1. Server 구현

```
message qos= 0
message retain flag= 0
message recived 3,{'pm10': 11.7, 'time': '29.03.2019 07:51:59', 'pm2.5': 8.5}
message topic= vds1/data
message qos= 0
message retain flag= 0
message recived 4,{'pm10': 13.3, 'time': '29.03.2019 07:53:02', 'pm2.5': 9.3}
message topic= vds1/data
message qos= 0
message retain flag= 0
^CTraceback (most recent call last):
  File "mqtt.py", line 18, in <module>
    client1.loop_forever()
  File "/usr/local/lib/python3.6/dist-packages/paho/mqtt/client.py", line 1578,
in loop_forever
    rc = self.loop(timeout, max_packets)
  File "/usr/local/lib/python3.6/dist-packages/paho/mqtt/client.py", line 1057,
in loop
    socklist = select.select(rlist, wlist, [], timeout)
KeyboardInterrupt
knusw@knusw-ThinkPad-T440:~/바탕화면$ python3 mqtt.py
message recived 1,{'pm2.5': 8.2, 'time': '29.03.2019 08:05:04', 'pm10': 12.5}
message topic= vds1/data
message qos= 0
message retain flag= 0
```

## 02. 프로젝트 수행 내용

### 2. Mini BuT 본체 구현

실내 미세먼지 데이터 수집 모듈 구현

측정된 실내 미세먼지 데이터 전송 모듈 구현

1602 LCD를 12C로 연결하여 동작 확인 및 모듈 구현

LED 동작 확인 및 모듈 구현

## 02. 프로젝트 수행 내용

### 2. Mini BuT H/W 구현



## 02. 프로젝트 수행 내용

### 3. 웹 페이지 구현



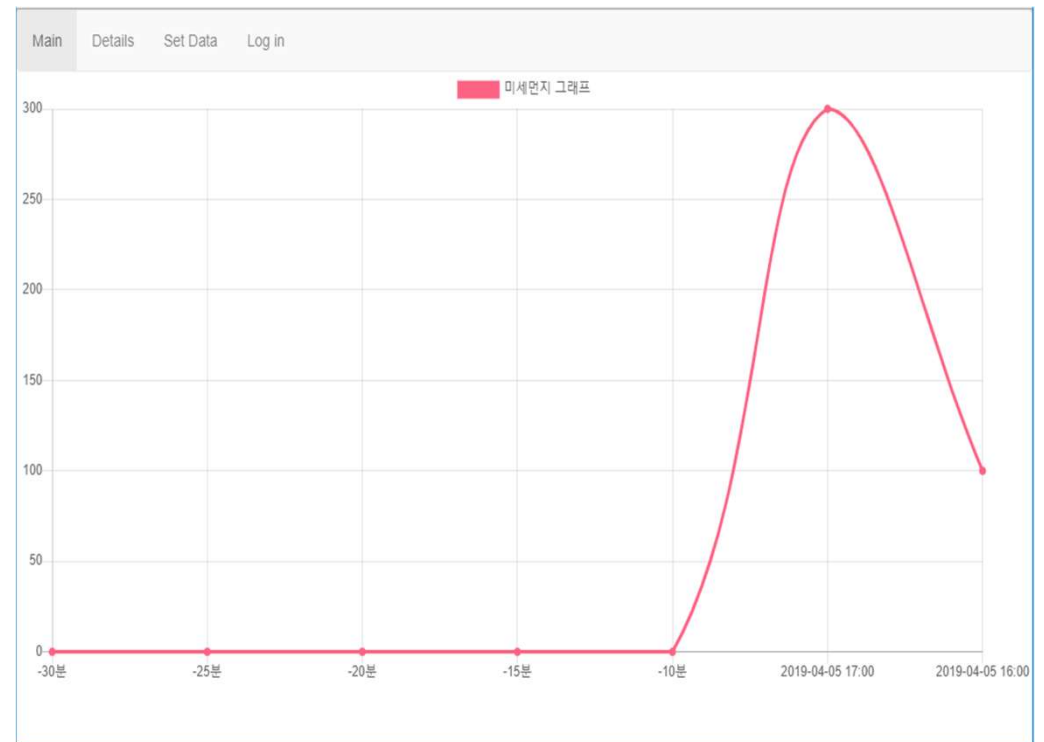
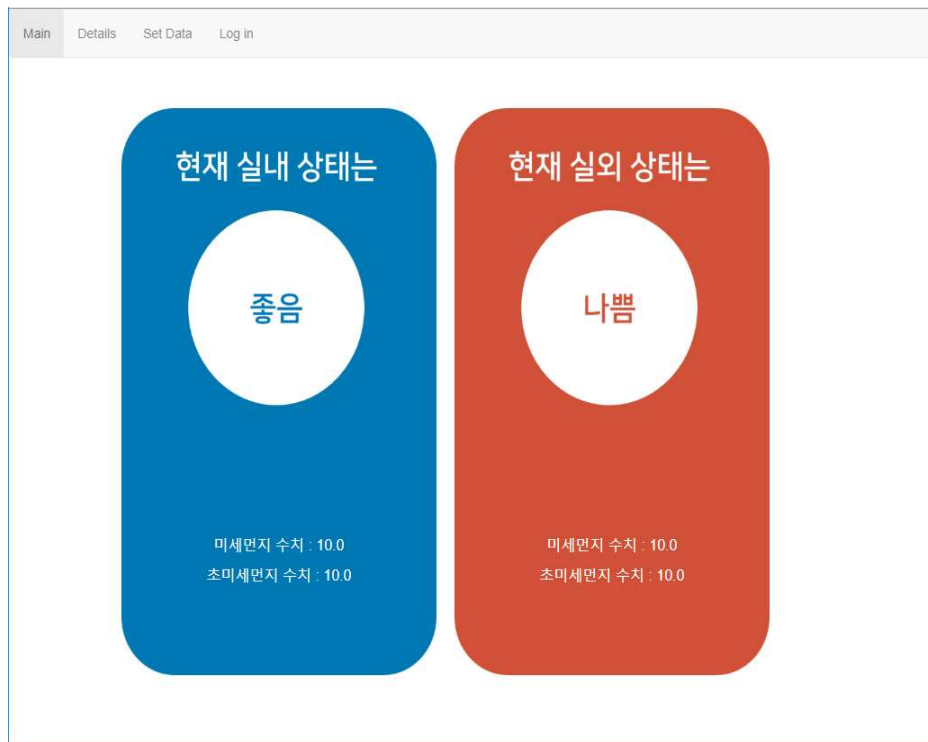
Database



Web

## 02. 프로젝트 수행 내용

### 3. 웹 페이지 구현





## 03.향후 추진 계획

---

- 1. Sever
  - 2. Mini BuT 본체 구현
  - 3. 웹 페이지
- 

## 03. 향후 추진 계획

### 1. Server

실외 데이터 수집 추가

DB 설계 수정 및 적용

Mini BuT에서 받은 측정 데이터 DB에 저장

수집 데이터 분석 모듈 설계 수정 및 구현

서버에서 라즈베리파이로 제어 명령 전달 통신 환경 구현





### 03. 향후 추진 계획

## 2. Mini BuT 본체 구현

서버에서 받은 명령어 처리 구현

Mini BuT 전체 프로그램 및 H/W완성



## 03. 향후 추진 계획

### 3. 웹 페이지

사용자 관리를 위한 계정 페이지 구현

수집한 데이터를 자세하게 확인 가능하고 다용도로 사용하기 위한 통계 페이지 구현

외부 기기의 측정 상황을 보여주어 기기가 종료되거나 연결에 문제가 생기는 등, 측정 값 오류를 파악하거나 사용자 관리를 위한 관리자용 페이지 구현



# Thank you

MBT