

IoT 팀프로젝트 최종보고서

작품명 : 건물 출입 기록 관리 시스템

| | | |
|------|---------|--------|
| 팀 명 | | 아는형님들 |
| 담당교수 | | 방 대 욱 |
| 참가자 | 학과 및 전공 | 컴퓨터공학과 |
| | 대표학생 | 김 동 준 |

- 목 차 -

| | |
|-----------------------------|----|
| 1. 일반정보 | 3 |
| 2. 참여학생 및 업무분담표 | 3 |
| 3. 수행배경 | 4 |
| 3-1. 개요 | 4 |
| 3-2. 센서 구성요소 | 4 |
| 4. 시스템 구성 | 4 |
| 5. 시스템 설계 | 5 |
| 5-1. Usecase Diagram | 5 |
| 5-2. Sequence Diagram | 6 |
| 6. 시스템 구성 요소 | 6 |
| 6-1. 라즈베리파이 | 6 |
| 6-2. 출입 기록 관리 프로그램 | 7 |
| 7. 결과물 | 8 |
| 7-1. 라즈베리파이 | 8 |
| 7-2. 파이썬 | 9 |
| 8. 기대효과 및 개선 사항 | 10 |

1. 일반 정보

| | | | | |
|--------------------|--------------------|---------------|-----|--------|
| 팀 명 | 아는형님들 | | | |
| 작품 명 | 건물 출입 체온 기록 관리 시스템 | | | |
| 대 표 참 여 학 생 | 성 명 | 학 번 | | 소속학과 |
| | 김 동 준 | 5293291 | | 컴퓨터공학과 |
| | 연 락 처 | 010-9224-1427 | | |
| 지도교수 | 성 명 | 방 대 욱 | 학 과 | 컴퓨터공학과 |

2. 참여학생 및 업무분담표

| No | 소속학과 | 성 명 | 학번 | 업무내용 |
|----|--------|-----------------|---------|-------------------------------|
| 1 | 컴퓨터공학과 | 김동준 | 5293291 | IoT Makers 통신 개발 및 센서 기능 개발 |
| 2 | 컴퓨터공학과 | 김영란 | 5416212 | 출입 기록 관리 프로그램 개발(Python GUI) |
| 3 | 컴퓨터공학과 | 박찬섭 | 5293550 | 출입 기록 관리 프로그램 개발(데이터 처리 및 통신) |
| 4 | 컴퓨터공학과 | 이강희 | 5360123 | 라즈베리파이 센서 기능 개발 및 외형 제작 |
| | 총 | (대표 김동준 외 3명) | | |

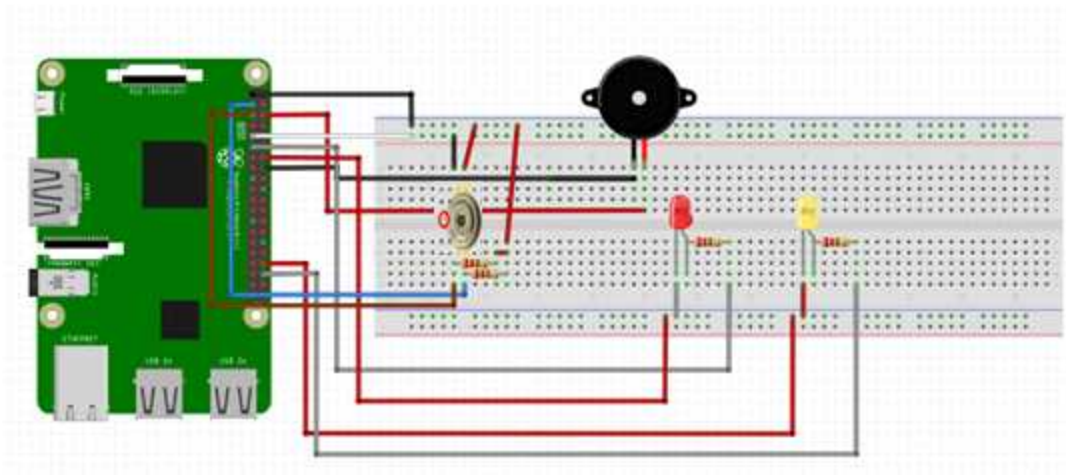
3. 수행 배경

3-1 개요

현재 코로나 사태로 인해 건물 출입 시 체온 측정과 명부 작성을 하고 있습니다. 기존의 수기로 작성하는 방식은 신뢰도가 낮고 해당 건물 관리자가 작성된 문서를 보관하고 있어 실시간으로 정보를 얻을 수가 없으므로 열 환자 데이터를 관리할 수 있는 시스템을 구축하고자 합니다.

건물 관리자는 프로그램을 통해 건물의 위치정보를 입력하고 체온계를 건물 출입구에 설치합니다. 건물에 입장하는 사람은 출입구에 설치된 체온계를 통해 체온을 잽니다. 체온을 잦후 QR코드를 이용해 출입자 정보와 해당 건물의 위치 정보 및 입력된 체온 정보는 플랫폼(IoT Makers)으로 전송됩니다. 건물관리자는 실시간으로 플랫폼에 기록된 데이터를 읽어와 출입 체온 기록을 관리합니다.

3-2 센서 구성요소

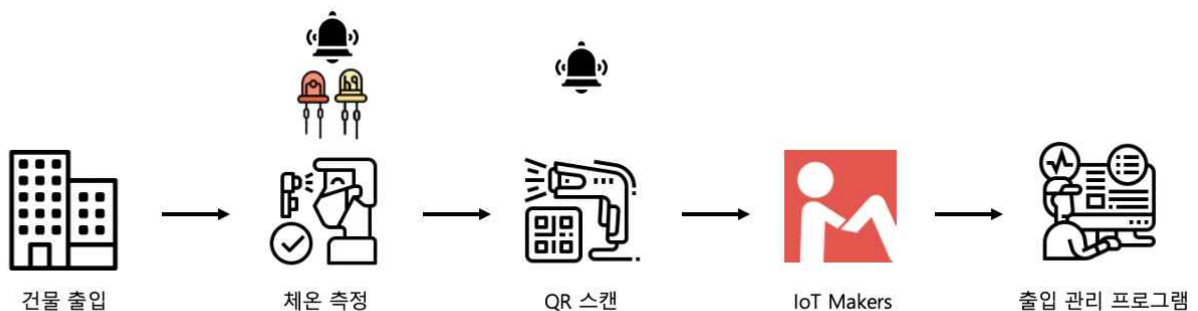


[그림 1] 센서 회로도

[그림 1] 센서 회로도입니다.

센서는 2개의 LED(RED, YELLOW), 피에조 부저, 비접촉식 온도센서로 구성되어 있습니다.

4. 시스템 구성



[그림 2] 시스템 구성도

[그림 2] 시스템 구성도입니다.

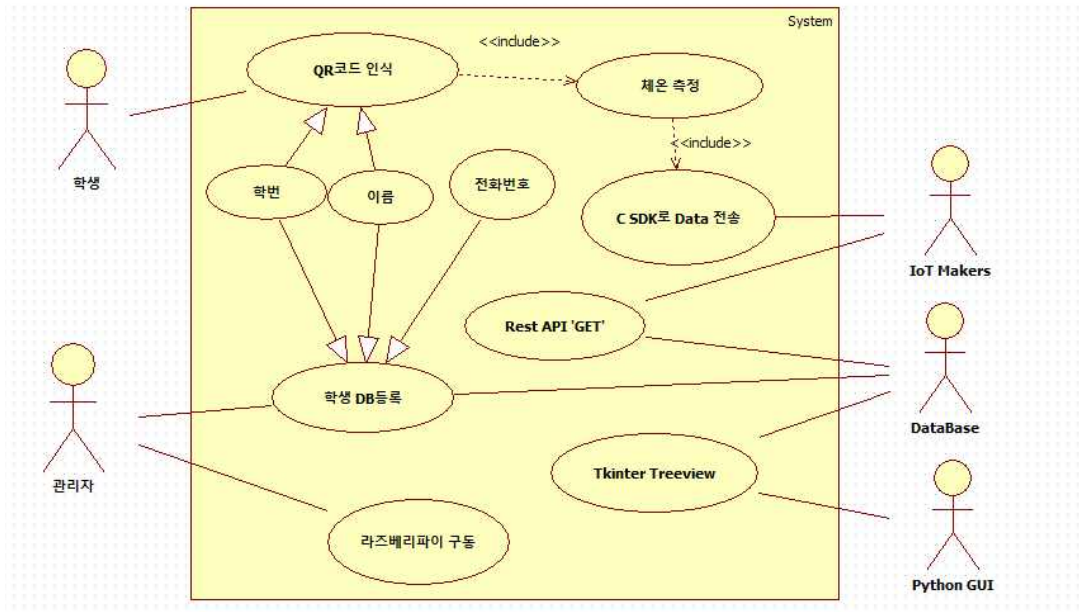
건물 출입 시 체온 측정을 합니다. 체온이 37.5도 이상이면 빨간색 LED가 켜지고 정상 체온일 경우 노란색 LED가 켜지며 부저가 울립니다.

그 후 출입자가 ID가 포함된 Mobile QR코드를 QR코드 스캐너를 통해 인식시키게 되면 부저가 울리며 IoT Makers에 건물 ID, 출입자의 ID, 체온 정보가 IoT Makers에 전송됩니다.

중앙관리자는 출입 관리 프로그램을 통해 IoT Makers에 입력되어 있는 각 건물에 출입기록을 조회할 수 있습니다.

5. 시스템 설계

5-1 Usecase Diagram



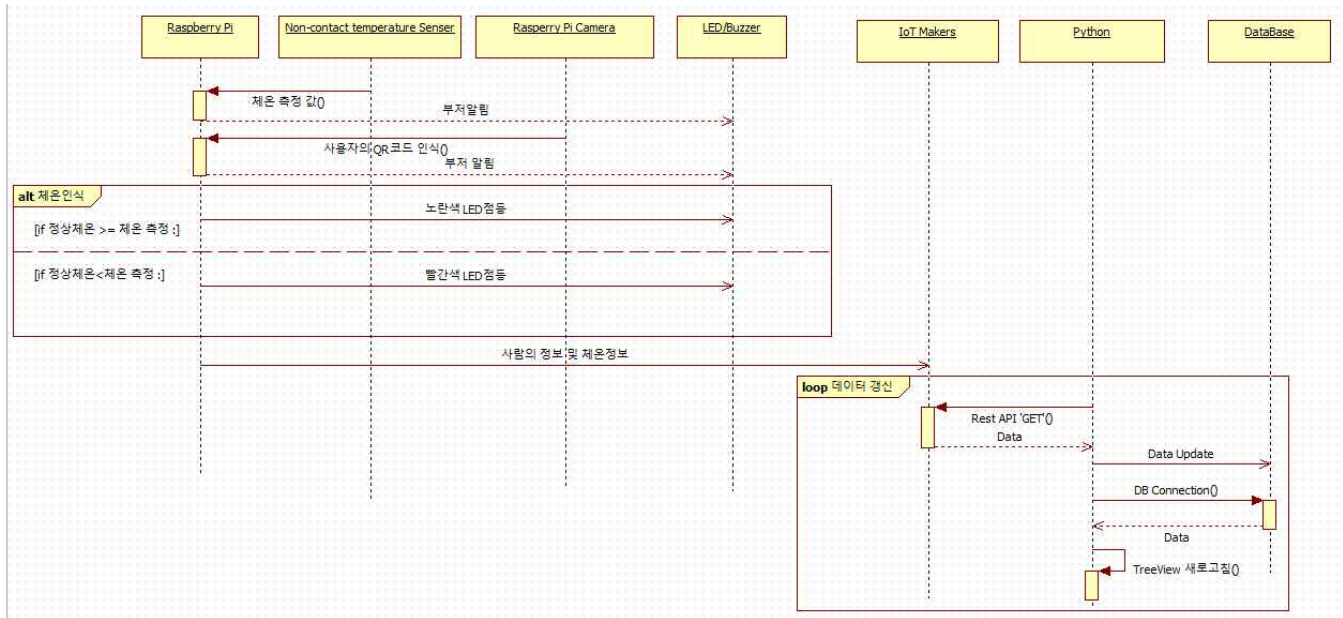
[그림 3] Usecase Diagram

[그림 3] Usecase Diagram 은 건물 출입 체온 기록 관리 시스템의 기능이나 서비스를 명세하고 있습니다. [표 5] Usecase Scenario에는 시스템에서 상호작용을 하는 과정을 구체적으로 묘사한 기술서입니다.

| | |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 유스케이스명 | 실시간 질병 관리 시스템 |
| 액터명 | 주 액터 : 학생, 출입관리하는 관리자 부 액터 : IoT Makers, DataBase, Python GUI |
| 개요 | 학교를 출입하는 사람들의 체온을 측정하고, 평균체온보다 높은 사람일 경우, 추적, 관리하여 코로나19 의심환자를 파악한다. |
| 사전조건 | 라즈베리파이의 네트워크 연결이 원활해야 한다. 라즈베리파이의 전원공급이 원활해야 한다. 비접촉 체온센서가 정상적으로 작동해야 한다. 카메라 모듈이 정상적으로 작동해야 한다. |
| 사후조건 | 체온이 측정된 경우, 부저가 울리며 QR코드를 인식한다. 체온 측정 후 정상체온보다 높은 경우 LED에 빨간불이 들어온다. 체온 측정 후 정상체온이면 LED에 노란불이 들어온다. 체온을 측정하지 않는다. QR코드가 인식되지 않는다. |
| 기본흐름 | 1. 관리자는 라즈베리파이를 설치한다. 2. 관리자는 학생 데이터를 저장한다. 3. 체온측정 유스케이스를 실행한다. 4. 체온측정 데이터와 사람의 정보가 IoT Makers에 전송이 된다. 5. DataBase에서 일정시간마다 Rest API의 'GET' 을 이용하여 Polling 방식으로 IoT Makers에서 데이터를 불러온다. 6. 라이브러리 Tkinter treeview를 이용하여 Python GUI를 실행한다. |
| 대체흐름 1 | 2a. 체온측정 후 QR코드 인식을 하지 않는 경우. 2a.1 라즈베리파이에 연결된 빨간색 LED가 깜박거림을 반복한다. |
| 대체흐름 2 | 3a. Python GUI에 인터넷이 연결되지 않은 경우 3a.1 Python GUI 하단에 'Connection Error' 이라고 뜨며 이제까지 저장된 데이터는 조회가 가능하게 함. |

[표 5] Usecase Scenario

5-2 Sequence Diagram



[그림 4] Sequence Diagram

[그림 4] Sequence Diagram 은 객차간의 상호작용을 보여주는 시퀀스 다이어그램입니다. 진행 순서는 다음과 같습니다

1. 비접촉식 온도센서에서 사람의 체온을 측정한다.
2. 체온측정 후 부저가 울리며 QR코드 인식준비가 된다.
3. QR코드 인식이 되면 부저가 울리며 측정한 체온이 정상일 경우 노란색 LED가 점등된다.
4. 측정한 체온이 정상이 아닌 경우, 빨간색 LED가 점등이 된다.
5. 수집된 데이터는 IoT Makers에 전송이 된다.
6. Python에서 일정 시간마다 IoT Makers의 데이터를 Rest API ‘GET’ 으로 가져온다.
7. 가져온 데이터를 데이터베이스에 저장할 하며 Python에서 데이터베이스에 저장된 데이터를 가지고 Treeview를 이용하여 시각적으로 표현을 한다.

6. 시스템 기능요소

6-1 라즈베리파이

라즈베리 파이는 각 건물 입구에 설치되어 출입자의 체온을 측정 및 QR 코드 스캔 등 센서를 통해 기능을 수행합니다. 또한, 출입 기록을 IoTMakers에 전송하는 기능을 수행합니다.

아래에서 각 기능에 대해서 세부적으로 설명합니다.

(1) 센서 제어

(a) LED

LED는 빨간색 LED와 노란색 LED가 설치되어 있습니다. 체온을 측정했을 때 체온이 37.5도 이상일 경우 빨간색 LED를 켜주고, 정상 체온일 경우에는 노란색 LED를 켜줍니다. 출입자는 LED 확인을 통해 자신의 체온이 정상인지 확인할 수 있습니다.

(b) 부저

체온 측정을 했을 때와 출입자가 QR코드를 스캔했을 때 부저가 울리게 구현했습니다. 출입자는 부저 소리를 통해 다음 절차를 진행해야 하는지 알 수 있습니다.

(c) 카메라 모듈

출입자의 ID가 포함된 모바일 ID인 QR코드를 인식하는 기능을 수행합니다. Open CV 라이브러리를 활용하여 QR코드를 인식하게 되면 QR코드에서 출입자의 ID를 추출합니다.

(2) 출입 기록 전송

IoT Makers에 등록되어 있는 디바이스의 태그 스트림을 추가한 후 해당 태그 스트림 및 디바이스 정보

를 Config File로 만든 후 C SDK를 활용하여 Config File을 통해 해당하는 디바이스 및 태그 스트림에 출입 기록을 전송합니다.

6-2 출입 기록 관리 프로그램

출입 기록 관리 프로그램은 라즈베리파이에서 IoT Makers에서 전송된 데이터를 가져옵니다. 가져오는 방식은 Python 기준으로 Polling으로 가져오게 되며, 가져온 데이터를 SQLite에 저장하고 SQLite에 저장된 데이터를 읽어와 Treeview로 사용자에게 보여주는 방식으로 진행이 됩니다.

(1)기본화면

| 출입 장소 | 학번 | 성명 | 제은 | 출입 시간 |
|--------|---------|-----|------|-------------------------|
| 간호학과 | 5416212 | 김영한 | 36.7 | 2020-06-20 00:41:38.359 |
| 간호학과 | 5360123 | 이강희 | 36.5 | 2020-06-20 00:41:25.657 |
| 본관 | 5294053 | 황재현 | 37.8 | 2020-06-20 00:41:03.952 |
| 사회과학대학 | 5293794 | 이성운 | 36.7 | 2020-06-20 00:40:45.772 |
| 대학원 | 5293715 | 윤주현 | 37.7 | 2020-06-20 00:40:33.626 |
| 인문대학 | 5293690 | 오승철 | 37.0 | 2020-06-20 00:40:15.805 |
| 사회대학 | 5293550 | 박찬섭 | 36.9 | 2020-06-20 00:39:58.620 |
| 사회대학 | 5293520 | 박수진 | 35.9 | 2020-06-20 00:39:45.116 |
| 공과대학 | 5293490 | 박성연 | 36.6 | 2020-06-20 00:39:25.565 |
| 공과대학 | 5293291 | 김동준 | 37.6 | 2020-06-20 00:39:13.549 |

마지막 수정 시간 2020-06-29 11:41:05
최근 로그 시간 2020-06-22 15:20:13

[그림 5] 기본화면

프로그램을 실행하게 되면 그림2와 같이 나오게 됩니다. 좌측 상단에는 날짜를 선택하여 조회할 수 있게 날짜선택창, 우측 상단에는 전체목록과 의심환자 목록을 선택할 수 있는 Combobox, 그리고 조회 버튼이 있습니다. 그리고 Treeview로 SQLite에 저장된 데이터들을 불러오며, 의심환자들은 빨간색으로 표현을 했습니다. 마지막으로 우하단에는 조회버튼을 누른 시간과 DB에 저장된 마지막 TimeStamp를 출력하게 됩니다.

[그림 5] 기본화면 에서는 보이지 않지만, 좌하단에 BackGround에서 IoT Makers에서 받아온 데이터를 SQLite로 업데이트하는 과정을 ProgressBar로 시각적으로 보여줍니다. 자세한 내용은 (3)에서 다루겠습니다.

| 출입 장소 | 학번 | 성명 | 제은 | 출입 시간 |
|-------|---------|-----|------|-------------------------|
| 본관 | 5294053 | 황재현 | 37.8 | 2020-06-20 00:41:03.952 |
| 대학원 | 5293715 | 윤주현 | 37.7 | 2020-06-20 00:40:33.626 |
| 공과대학 | 5293291 | 김동준 | 37.6 | 2020-06-20 00:39:13.549 |

마지막 수정 시간 2020-06-29 12:11:56
최근 로그 시간 2020-06-22 15:20:13

[그림 6] 날짜 선택창(좌), 의심환자 조회(우)

(2)Treeview Double Click

Treeview는 SQLite에 저장된 데이터들을 보여줍니다. 리스트형식으로 보여주는 Treeview에서 원하는 항목을 더블클릭할 시 해당 사람의 학번, 성명, 소속전공 전화번호의 학사정보가 보여지게 됩니다.

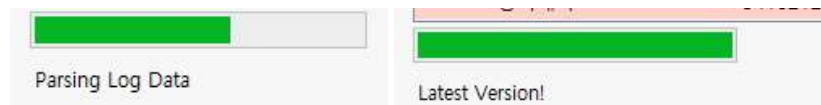
(3)Progress Bar

교내 출입 기록

| 학사 정보 | | 성명 | 체온 | 출입 시간 |
|--------|---------------|-----|------|-------------------------|
| 학번 | 5293715 | 김영란 | 36.7 | 2020-06-20 00:41:38.359 |
| 성명 | 윤주현 | 이강희 | 36.5 | 2020-06-20 00:41:25.657 |
| 소속전공 | 경영학과 | 황재현 | 37.8 | 2020-06-20 00:41:03.952 |
| 전화번호 | 010-4434-1345 | 이성운 | 36.7 | 2020-06-20 00:40:45.772 |
| 사회과학대학 | 5293794 | 윤주현 | 37.7 | 2020-06-20 00:40:33.626 |
| 대학원 | 5293715 | 오승철 | 37.0 | 2020-06-20 00:40:15.805 |
| 인문대학 | 5293690 | 박찬섭 | 36.9 | 2020-06-20 00:39:58.620 |
| 사회대학 | 5293550 | 박수진 | 35.9 | 2020-06-20 00:39:45.116 |
| 공과대학 | 5293520 | 박성연 | 36.6 | 2020-06-20 00:39:25.565 |
| 공과대학 | 5293490 | 김동준 | 37.6 | 2020-06-20 00:39:13.549 |
| 공과대학 | 5293291 | | | |

마지막 수정 시간 2020-06-29 11:48:19
최근 로그 시간 2020-06-22 15:20:13

[그림 7] Treeview Double Click



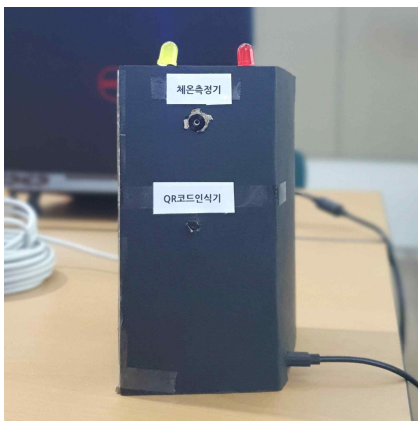
[그림 8] Data를 Parsing중 (좌), 업데이트가 최신일 때(우)

Progress Bar는 Background에서 진행되는 과정을 시각적으로 보여줍니다. IoT Makers에서 SQLite에 업데이트 되는 과정을 Progress Bar로 표현이 됩니다. 순서는 'Bring Log From IoT Makers', 'Parsing Log Data', 'Update DataBase', 'Done!'이며 인터넷이 연결이 되지 않거나 Access Token이 만료되는 등 Connection 에러가 뜨면 'Connection Fail' 이라고 뜨게 됩니다.

Progress Bar가 진행되는 과정에는 조회버튼이 비 활성화가 되며 Treeview의 DoubleClick도 비활성화가 됩니다. 완료가 된다면 다시 조회버튼과 Doubleclick이 활성화가 됩니다.

7. 결과물

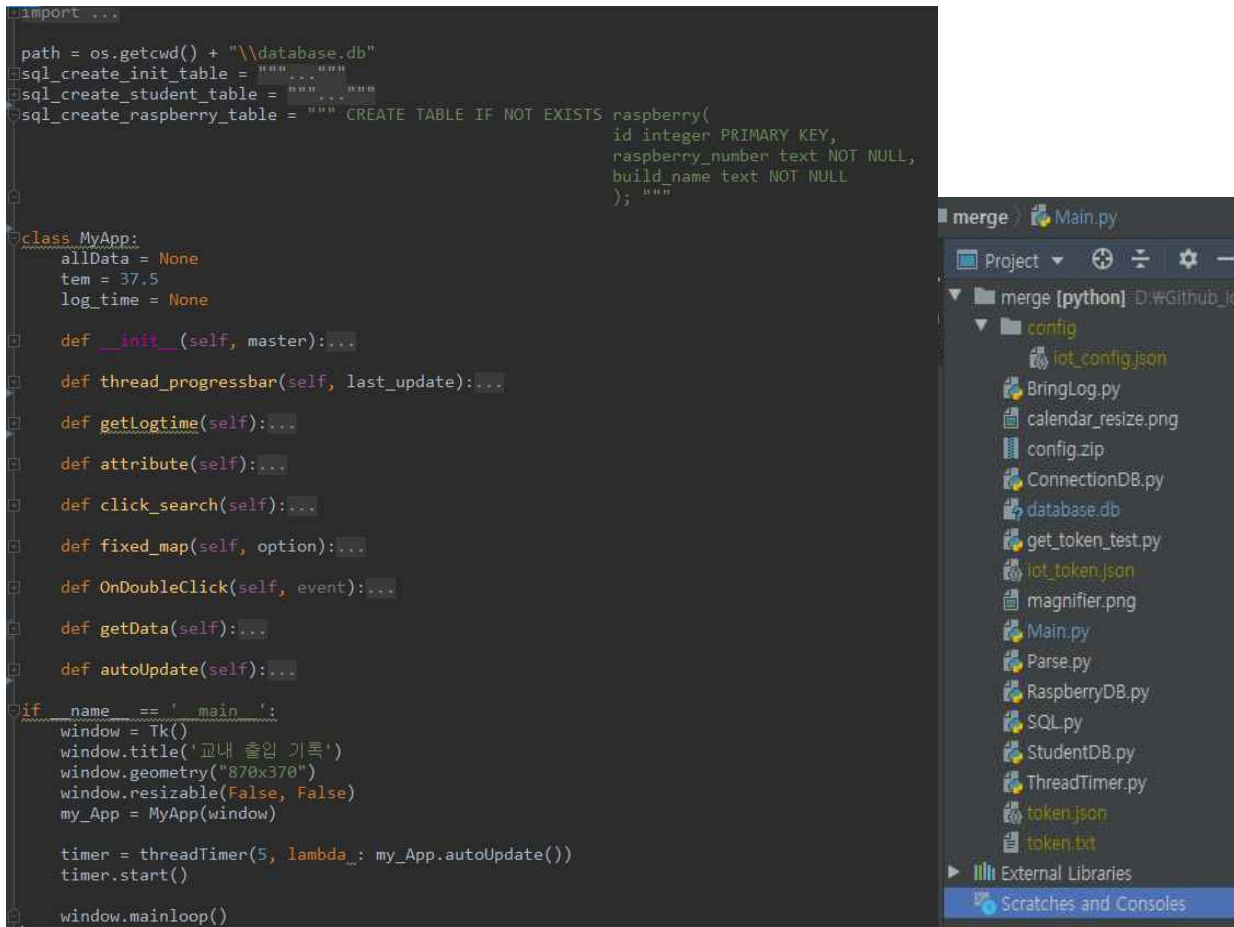
7.1 라즈베리파이



[그림 9] 라즈베리파이 외형

[그림 9] 은 건물 출입구에 설치되는 라즈베리파이의 외형입니다. LED 2개와 체온 측정기, QR코드 인식기로 구성되어 있습니다.

7.2 파이썬



[그림 10] Main 코드(좌), 어플리케이션 구성 파일들(우)

Main은 어플리케이션을 구성하는 Python Main 코드입니다. 아래 [표 6] Python Class별 기능 명세표는 Python을 구성하는 클래스들을 설명하는 표입니다.

| | |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Main | Tkinter를 이용하여 Python GUI를 실행합니다. Button combo box, calendar, label, progress bar와 Treeview를 정의합니다. Treeview에 OnDoubleClick을 이벤트로 등록합니다. Thread를 이용하여 자동으로 IoT Makers에서 데이터를 가져와 업데이트를 합니다. Thread의 진행상황을 progress bar로 보여줍니다. |
| get_token_test | Access Token이 만료될 경우 Token을 재발급 받습니다. 재발급받은 Token을 json형식으로 저장합니다. |
| BringLog | json파일에 저장된 Access Token을 불러옵니다. Rest API를 이용하여 'GET' 방식으로 데이터를 불러옵니다. |
| RaspberryDB | 라즈베리파이 테이블과 학생테이블을 정의합니다. |
| StudentDB | 라즈베리파이 시리얼 번호와 학생의 학사 정보를 저장합니다. |
| connectionDB | SQLite에 접근하기위한 DB Connection을 합니다. |
| SQL | Main에서 받은 위치(현재 파일의 위치)에 SQLite DB를 생성합니다. 각 정의된 Table을 생성합니다. Database에 있는 모든 데이터를 불러옵니다. Database의 마지막 타임스탬프를 불러옵니다. Database에 있는 Table들을 Join하여 데이터를 불러옵니다. |
| Parse | IoT Makers에서 받은 Json형식 데이터를 Database에 저장하기 위해 파싱합니다. 파싱한 데이터는 Database에 저장합니다. |
| ThreadTimer | Thread에 함수를 실행시키기 위해 정의합니다. Thread가 종료되지 않는 경우를 고려합니다. |

[표 6] Python Class별 기능 명세표

8. 기대효과 및 개선사항

(1) 기대효과

- (a) 건물 출입 정보를 활용하면 파악이 용이하기 때문에 역학조사 시 인적, 물적 자원에 대한 손실 감소를 될 것입니다.
- (b) 현재는 각 학교, 기관 및 회사에서 관리자가 데이터를 조회할 수 있으나 추후 모든 데이터들을 통합한다면 상위 기관(질병관리본부)에서도 활용할 수 있습니다.

(2) 개선사항

- (a) 임시로 외형 제작을 했기 때문에 3D 프린터 등을 이용한 외형 제작이 필요합니다.
- (b) 출입 기록 관리 프로그램(파이썬 프로그램)에서 IoTmakers에 있는 출입 로그를 RESTful API를 이용해 PULL방식으로 구현하였으나 IoTmakers에서 제공하는 MQTT SDK를 사용하여 PUSH방식으로 IoTmakers에 출입로그가 들어왔을 때 파이썬 프로그램에 로그를 전송해주는 방식으로 통신 방식의 개선이 필요합니다.