

# 눈치보이조



# 8조 눈치보이조

지도교수  
한재일 교수님

팀장  
김도훈

팀원  
곽지훈  
김상원  
명석현  
홍령기  
소가위



# 목차

1.

1차 중간 자문평가 피드백

2.

피드백 반영 사항

3.

진행 사항

4.

시연 영상



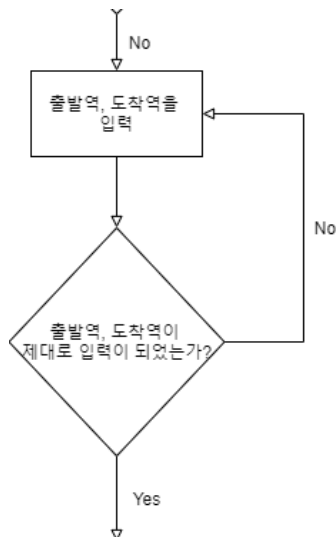
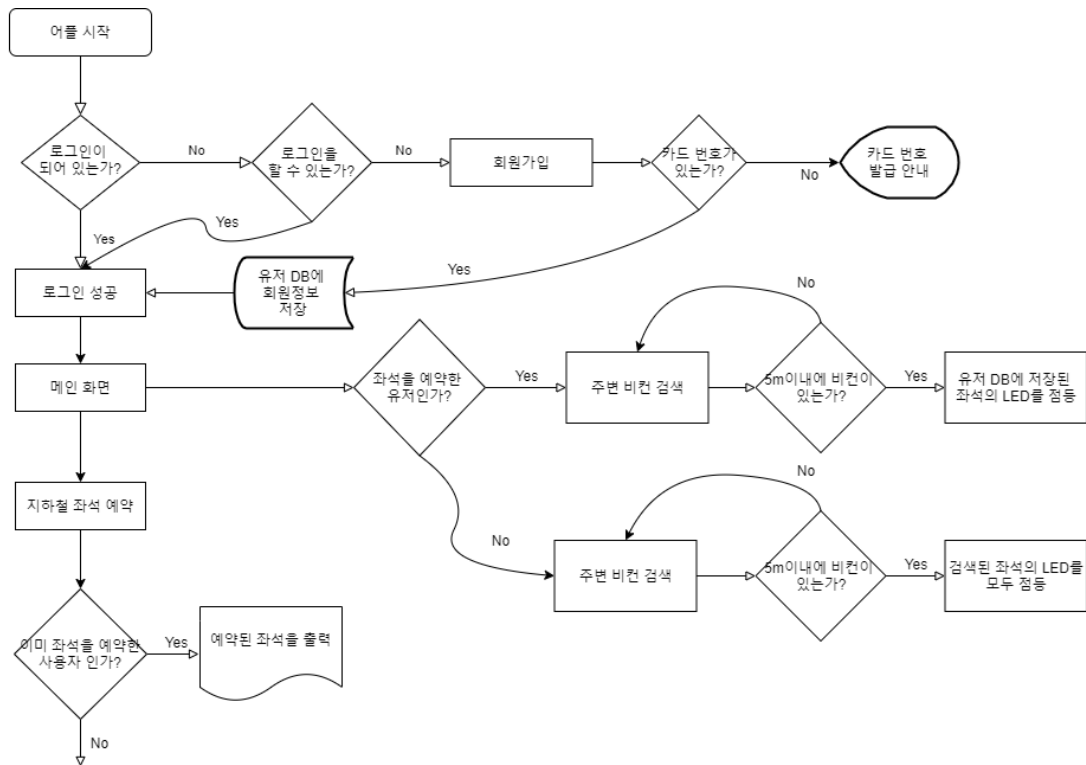
# 1. 1차 중간 자문평가 피드백

1. 프로젝트가 완료되었을 때 예상 되는 결과를 더욱 명확히 정의할 필요가 있다.
2. Test Plan을 만들어 보는 것이 과제의 목표를 명확히 하는데 도움이 될 것이다.
3. 열차 연결 예약 알고리즘에 대해 머신러닝을 추가하면 좋을 것 같다.

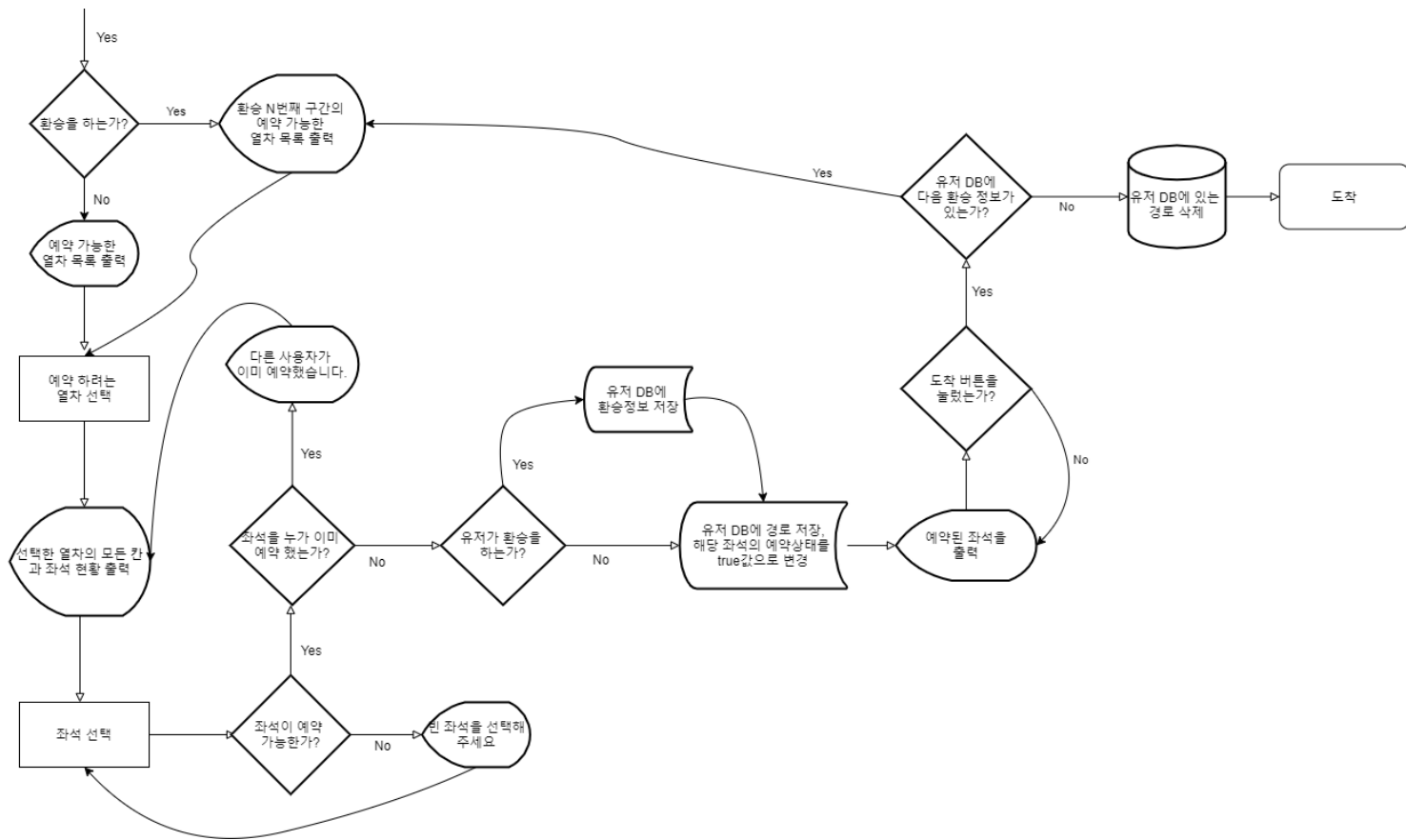


## 2. 피드백 반영 사항

- 1, 2번 피드백의 경우 Test Plan을 보다 명확히 했습니다.

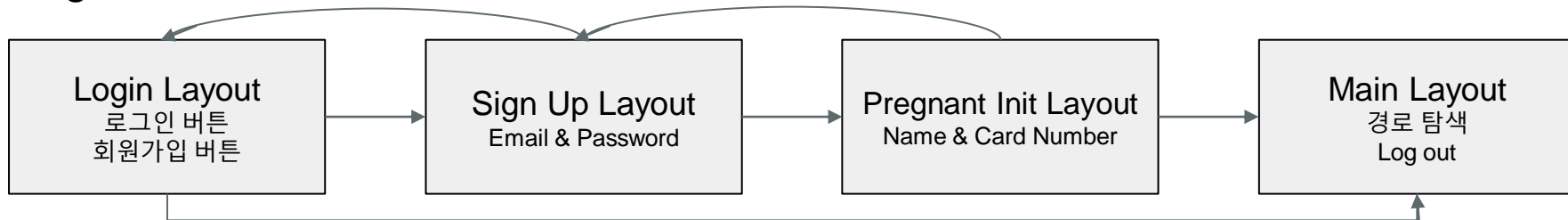


## 2. 피드백 반영 사항

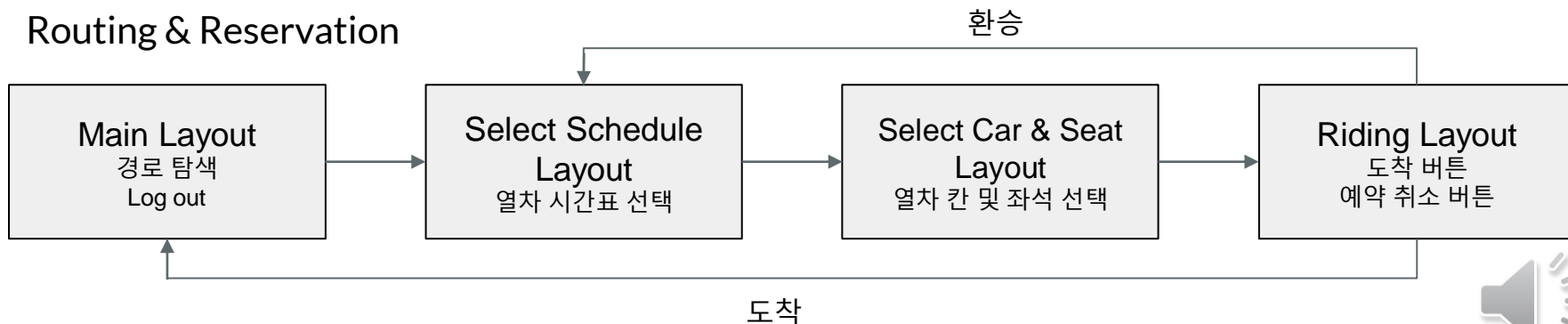


### 3. 진행 사항 (Application)

#### Login & authentication



#### Routing & Reservation



### 3. 진행 사항 (Application)

#### BLE Filtering & Packet Transmission

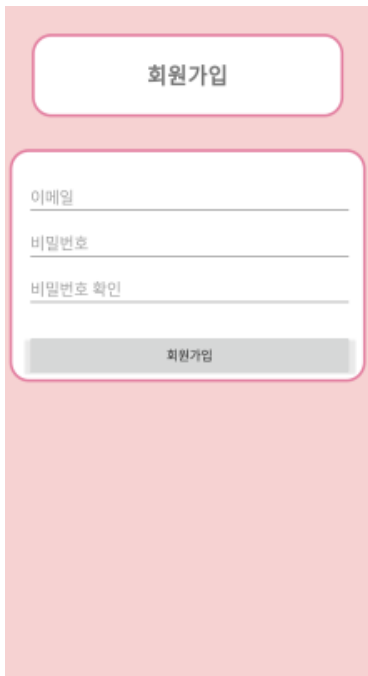




### 3. 진행 사항 (Application)



Mobile app home screen mockup. The screen has a green status bar at the top with the time 1:02. Below the status bar is a white header with the text "눈치보이조". The main content area has a large orange circle containing a white square with the text "눈치보이조?" and a stylized orange car icon. Below the circle are two input fields labeled "이메일" and "비밀번호". At the bottom are two buttons: "로그인" and "회원가입".



Mobile app membership sign-up screen mockup. The screen has a white header with the text "회원가입". Below the header are three input fields labeled "이메일", "비밀번호", and "비밀번호 확인". At the bottom is a button labeled "회원가입".

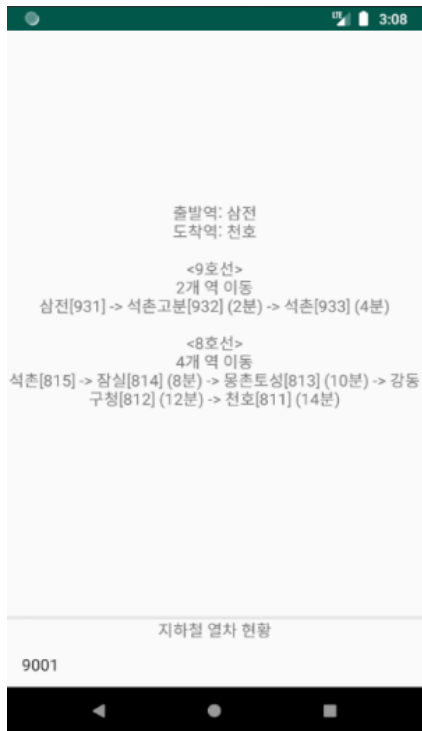


Mobile app membership verification screen mockup. The screen has a white header with the text "임산부 인증". Below the header are two input fields labeled "이름" and "카드번호(- 떼고 작성)". At the bottom is a button labeled "등록".

- 회원가입, 로그인, 산모 인증 구현 완료.
- 현재는 이메일과 비밀번호를 이용한 회원가입만을 지원.
- 임산부 인증의 경우, 카드번호는 각 유저마다 DB내에 저장되어 있는 카드번호와 비교하여 인증 여부를 결정



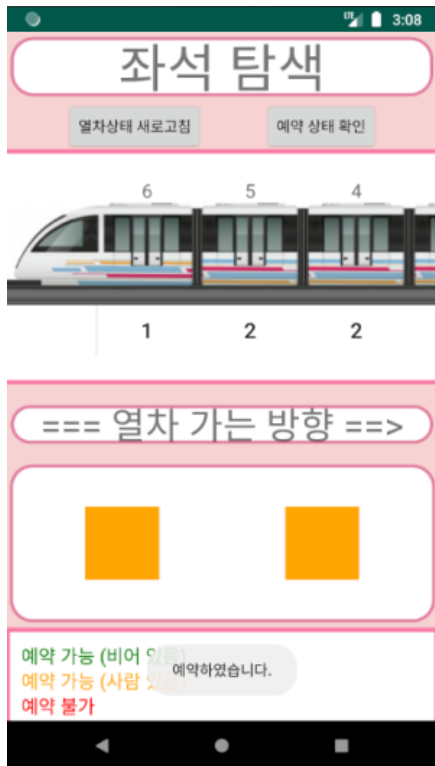
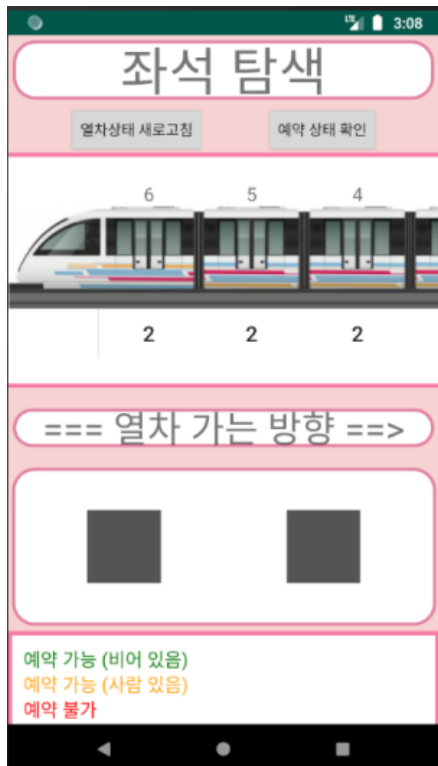
### 3. 진행 사항 (Application)



- 지하철 경로 찾기 구현 완료.
- Open source인 ODSay API의 지하철 경로검색 조회 레퍼런스 사용.
- 출발역 도착역 입력 후 경로 출력 (한글 또는 영어).
- 환승이 필요한 구간일 때, 첫 번째 구간 예약.
- 환승 후 구간은 첫 번째 구간이 끝난 후 예약.
- 하단의 열차 목록을 드래그 하여 열차 선택.



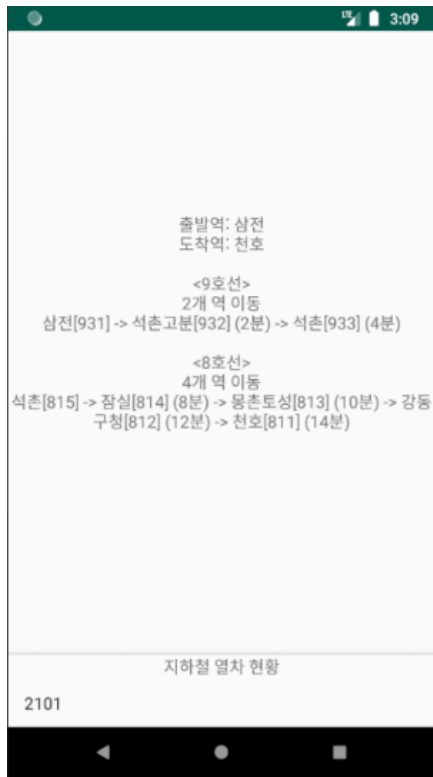
### 3. 진행 사항 (Application)



- 선택한 열차의 좌석 현황.
- 원하는 열차 칸 선택 후 좌석 상태를 색상으로 표현.
- 좌석 선택 시 해당 좌석의 상태 재확인 후 예약 시도.
- 상단의 예약 상태 확인 버튼 클릭 후 예약 현황 화면으로 이동.



### 3. 진행 사항 (Application)



- 현재 예약 되어있는 좌석 현황.
- 목적지에 도착하여 좌석 이용이 끝났을 시 도착 버튼 클릭.
- 환승이 남아있으면 열차 선택 화면으로 이동 및 다음 구간 예약.
- 최종 목적지에 도착하거나 예약취소 버튼 클릭 시 모든 예약 취소 및 처음 화면으로 이동.



### 3. 진행 사항 (Application)



```
try {
    Log.d(TAG2, msg: "try문 진입");
    socket = IO.socket( url: "http://0c358090.ngrok.io");
    Log.d(TAG2, msg: "소켓 생성");
    socket.on(Socket.EVENT_CONNECT, onConnect);
    Log.d(TAG2, msg: "연결");
    socket.connect();
    socket.on( event: "myMsg", onNewMessage);
    Log.d(TAG2, msg: "마이너 보냄");
} catch (URISyntaxException e){
    throw new RuntimeException(e);
}
//textView.append("Beacon TXPOWER : "+txpower+"\n");
//myStartActivity(NodeTestActivity.class, minor);
```

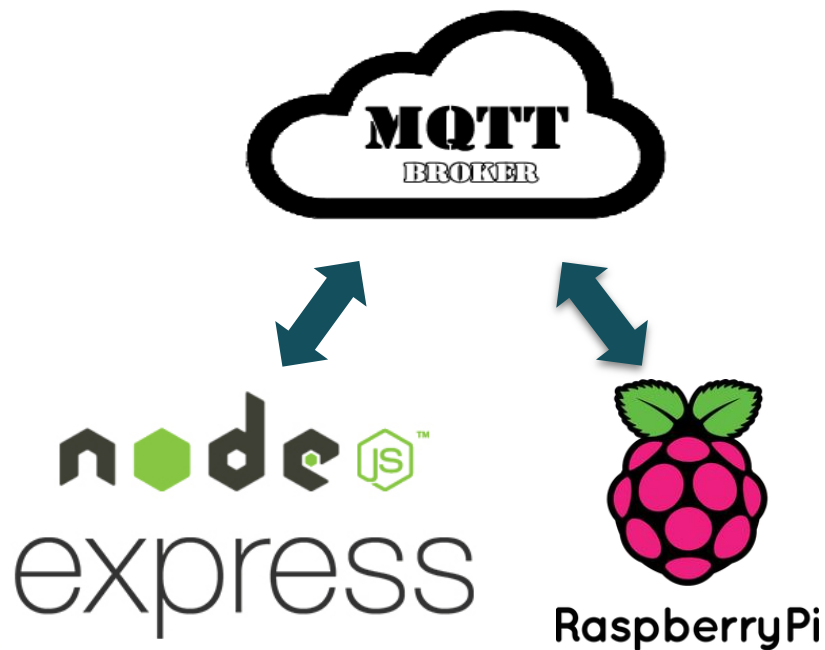
```
static Emitter.Listener onConnect = (args) => {
    socket.emit( event: "say", ...args: "hi");
    Log.d(TAG2, msg: "say 이벤트");
};

static Emitter.Listener onNewMessage = (args) => {
    socket.emit( event: "login", minor_to);
    Log.d(TAG2, msg: "login 이벤트");
};
```

- App에서 라즈베리 파이의 BLE 패킷을 100ms마다 수신하여, 패킷의 데이터를 분석
- 패킷의 TXpower값이 3이내로(3m이내) 들어올 경우, 해당 Major, Minor값을 설정된 Express server로 전송
- Express server와 App간 통신은 1대1이며, Socket.io를 이용하여 순차적으로 Connection, Say, Login이벤트를 발생



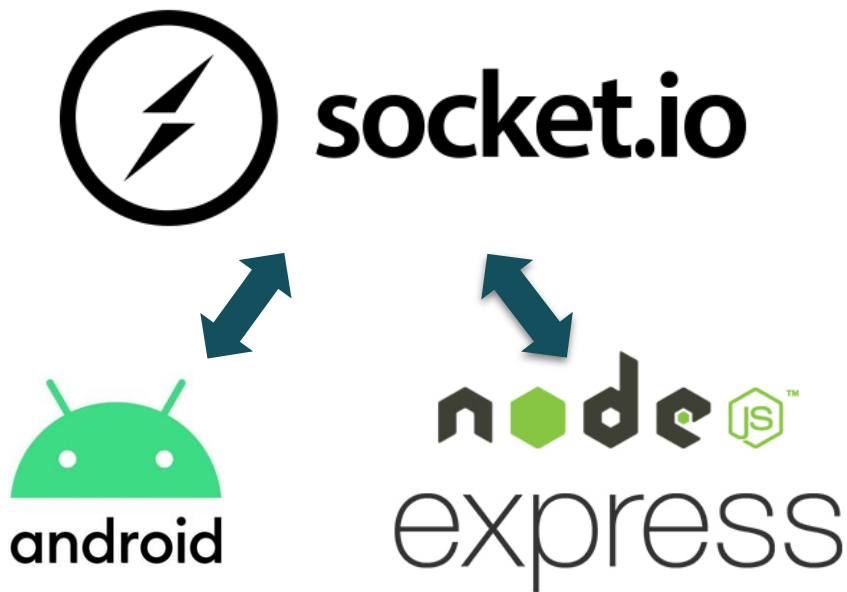
### 3. 진행 사항 (Server)



- Google Firebase를 사용하기 때문에 기존 계획에는 없었지만 라즈베리 파이와 어플리케이션 간의 통신에 MQTT broker를 사용하기 위해서 Express 서버를 구현하였습니다.
- Express 서버는 라즈베리 파이로부터 MQTT Broker를 통해 로드셀 센서값을 받아와 Firestore의 값을 갱신합니다.
- 또한 안드로이드로부터 비컨의 Minor 값을 받아와 역시 MQTT Broker를 통해 라즈베리 파이에 전달합니다.



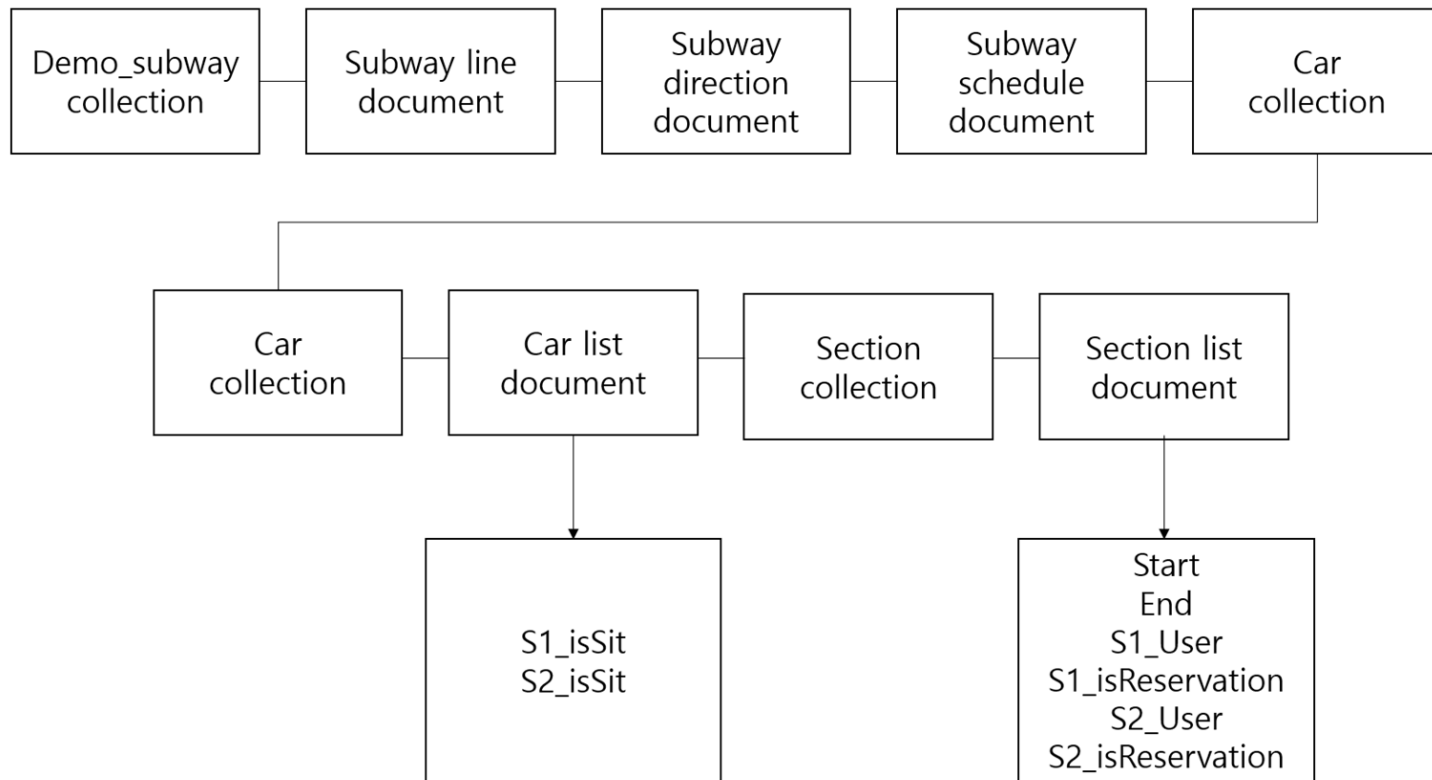
### 3. 진행 사항 (Server)



- Express 서버는 안드로이드와 라즈베리파이를 연결하기 위해서 socket.io room을 통해 멀티 통신합니다. 단 현재는 room이 구현되지 않아 1:1 통신만 가능합니다.
- socket.io를 통해서 비컨의 Minor 값을 express에 전송합니다.



### 3. 진행 사항 (DB)





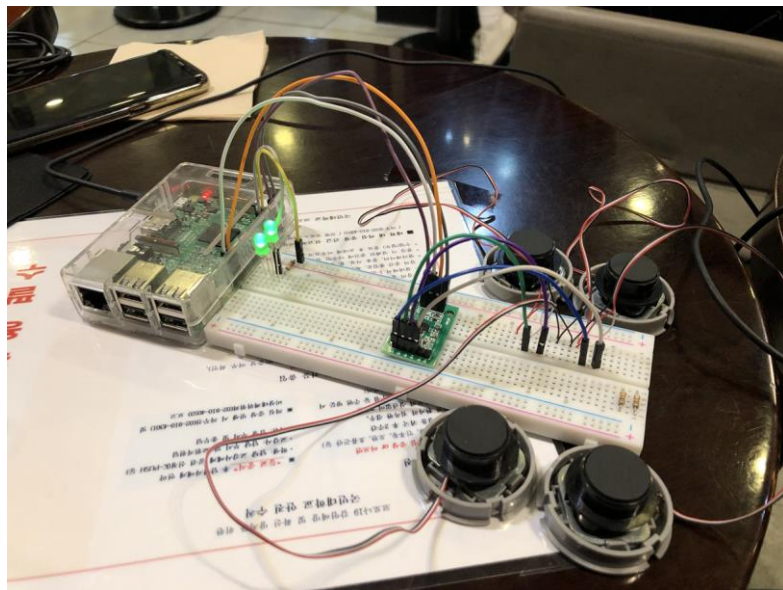
### 3. 진행 사항 (DB)

- Firestore에 사용자 정보를 저장할 user collection, 산모 인증을 위해 카드번호와 이름을 저장해둘 pregnant\_init collection, 예약 및 좌석 정보를 저장할 Demo\_subway collection을 만들었습니다.
- user collection의 각 document들은 id, password, name, cardNum과 인증 여부를 저장하는 isPregnant, 예약 정보와 환승 정보를 저장하는 reservation\_info, transfer\_info의 field를 갖습니다.
- pregnant\_init collection은 실제 카드를 사용할 수 없기 때문에 임의의 name, cardNum field를 갖습니다.
- Demo\_subway의 경우 시연을 위한 8호선과 9호선의 DB만을 갖습니다. 각 노선 document에는 열차의 리스트들이 저장되며 각각의 열차는 그들의 칸에 대하여 좌석의 상태를 저장할 수 있습니다. 또한 각각의 좌석에 대하여 노선의 구간 별로 예약 정보를 저장할 수 있습니다.
- 기존의 DB는 특정 경로를 예약해버리면 예약자가 하차하기 전에는 다른 사용자의 경로에 상관없이 예약을 할 수 없었던 문제를 해결하기 위해 각 칸마다 노선의 구간별로 좌석정보를 저장할 수 있게 했습니다.



### 3. 진행 사항 (Hardware)

- 아두이노에서 라즈베리 파이로 하드웨어 설계 변경  
=> 하나의 실행파일에서 모든 동작이 진행되는 아두이노의 성능상 한계로 인해 자체 OS에서 여러 프로그램을 한번에 실행시킬 수 있는 라즈베리 파이로 변경하였습니다.

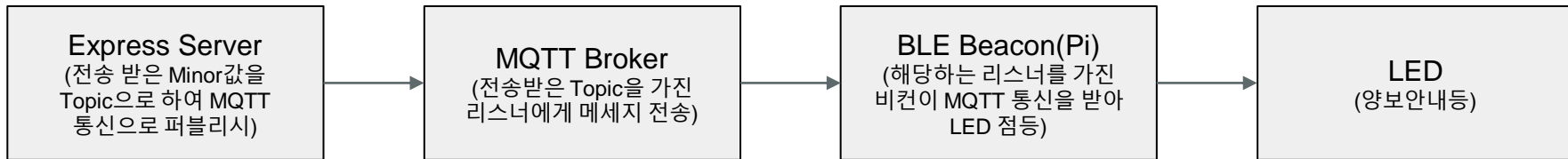


```
pi@raspberrypi: ~/Desktop/PinkLight
pi@raspberrypi:~/Desktop/PinkLight $ sudo python Load_Cell.py
^Ccleaning...
pi@raspberrypi:~/Desktop/PinkLight $ sudo python Load_Cell.py
1

pi@raspberrypi:~/Desktop/PinkLight
pi@raspberrypi:~/Desktop/PinkLight $ cd Desktop/PinkLight/
pi@raspberrypi:~/Desktop/PinkLight $ sudo python Led_Test.py
Connected with result code 0
1
^CTraceback (most recent call last):
  File "Led_Test.py", line 36, in <module>
    client.loop_forever()
  File "/usr/local/lib/python2.7/dist-packages/paho/mqtt/client.py", line 1782, in loop_forever
    rc = self.loop(timeout, max_packets)
  File "/usr/local/lib/python2.7/dist-packages/paho/mqtt/client.py", line 1163, in loop
    socklist = select.select(rlist, wlist, [], timeout)
KeyboardInterrupt
pi@raspberrypi:~/Desktop/PinkLight $ sudo python Led_Test.py
Led_Test.py:8: RuntimeWarning: This channel is already in use, continuing anyway.
  Use GPIO.setwarnings(False) to disable warnings.
  GPIO.setup(LED, GPIO.OUT, initial = GPIO.LOW)
Connected with result code 0
1
0
1
```

### 3. 진행 사항 (Hardware)

- 어플리케이션이 거리 내 BLE를 인식하여 해당 비컨의 LED를 점등하는 기능
  - App에서 BLE 비컨 인식 => 익스프레스 서버로 해당 비컨의 Minor값 전송

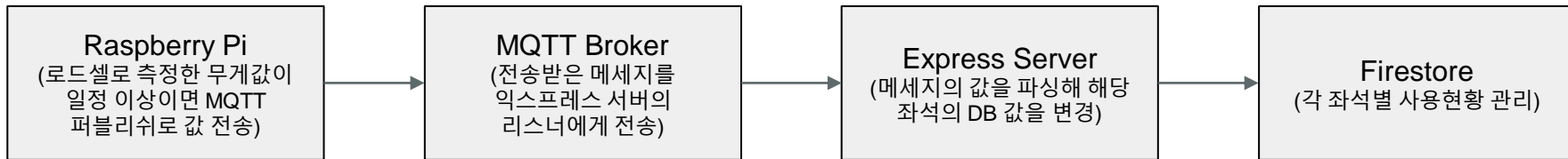


- 각 비컨(라즈베리 파이)는 Minor 값으로 구분되므로 다른 비컨의 LED는 켜지지 않습니다.
- 해당 좌석이 예약되었을 경우 예약자만이 다가갔을 때 양보안내등이 켜지는 기능구현 예정입니다.



### 3. 진행 사항 (Hardware)

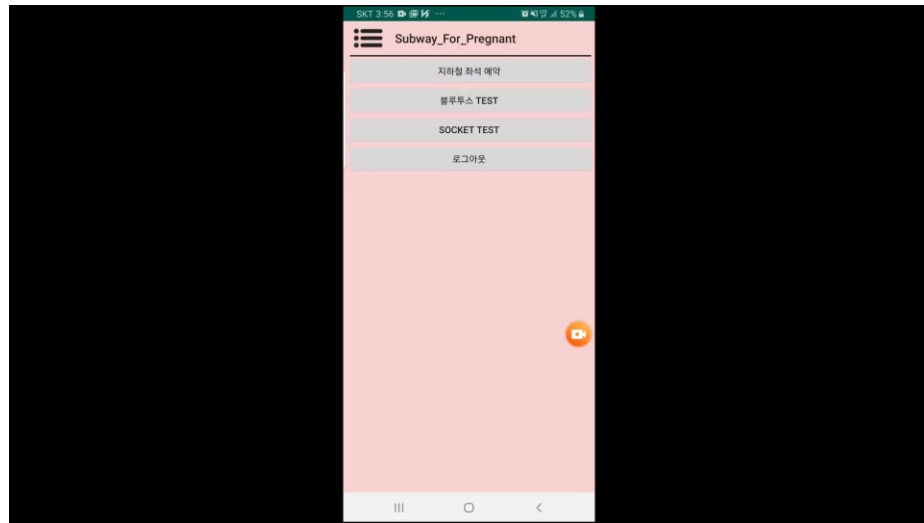
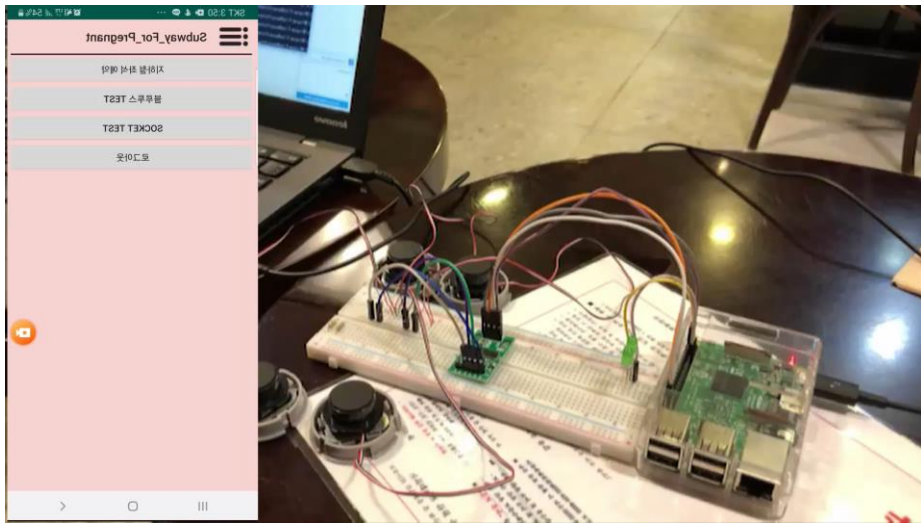
- 로드셀 센서를 라즈베리 파이와 연동하여 좌석 사용 여부 DB에 저장 기능
  - 로드셀 센서를 이용, 일정 이상의 무게를 감지하게 되면 사람이 앉아있다고 판단, 좌석이 사용중이라는 내용을 서버를 통해 DB로 전송하도록 기능을 구현하였습니다.



- 하드웨어 사진에서 보신 로드셀은 실제 제작되는 시제품의 의자 밑에 장착 될 예정입니다.
- 지금은 좌석이 사용중인지 아닌지만 알 수 있고, 해당 좌석에 앉아있는 사람이 산모인지 아닌지는 판별할 수 있는 방법이 없어 해당 내용을 확인 할 수 있는 방법을 고안중입니다.



## 4. 시연 영상



감사합니다.

