

OTP를 이용한
무인 택배 보관함

Made by 강통코딩

Back-end개발 (2)

수원대학교 정보통신학과 졸업작품 보고서 - 5~6주차

목차

I. Node.js Server

II. Web Application(HTML)

III. 소켓통신(Server – Web)

IV. Mosquitto Server(MQTT)

5월 2주차

V. RaspberryPi

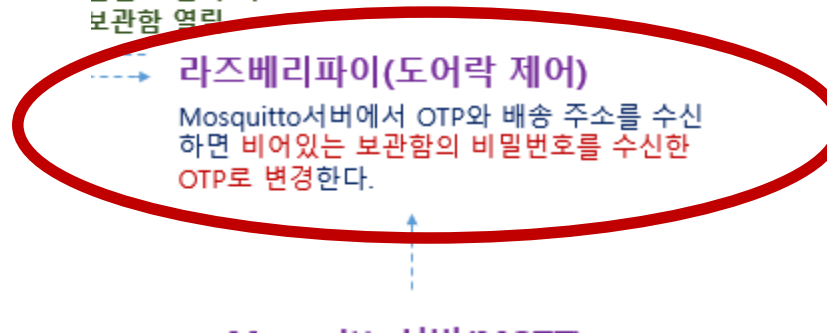
5월 3주차

VI. Android

Editor : PyCharm, cmd

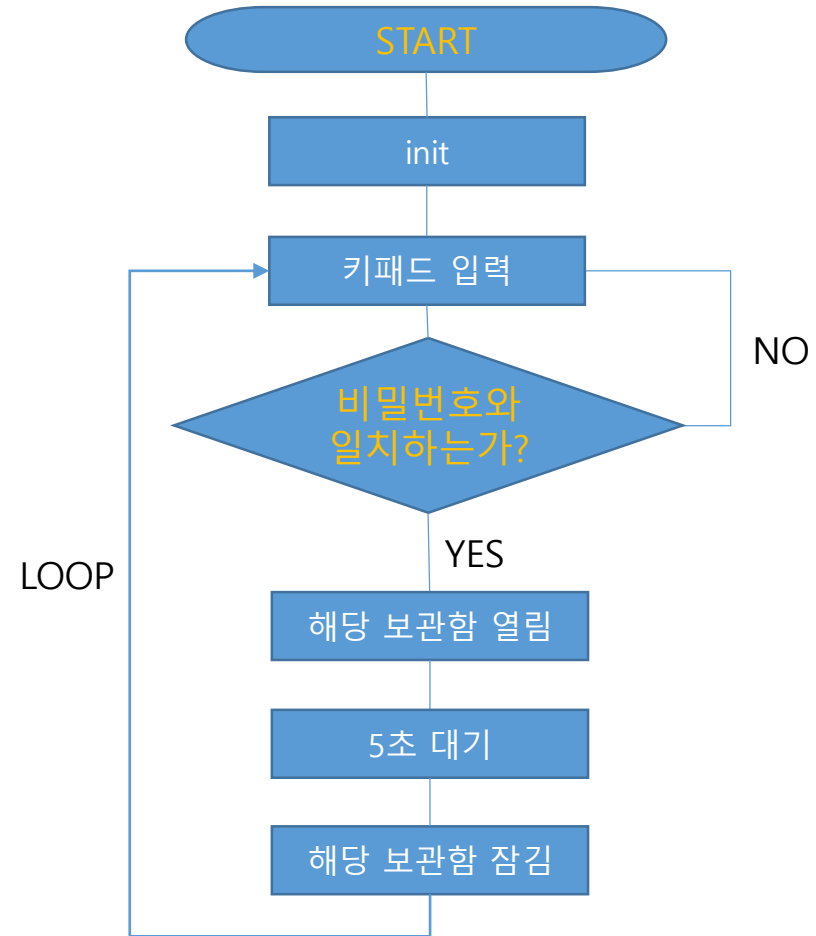
5. RaspberryPi

길번호 입력 시
보관함 열리



RaspberryPi

라즈베리파이 순서도



RaspberryPi

전역변수

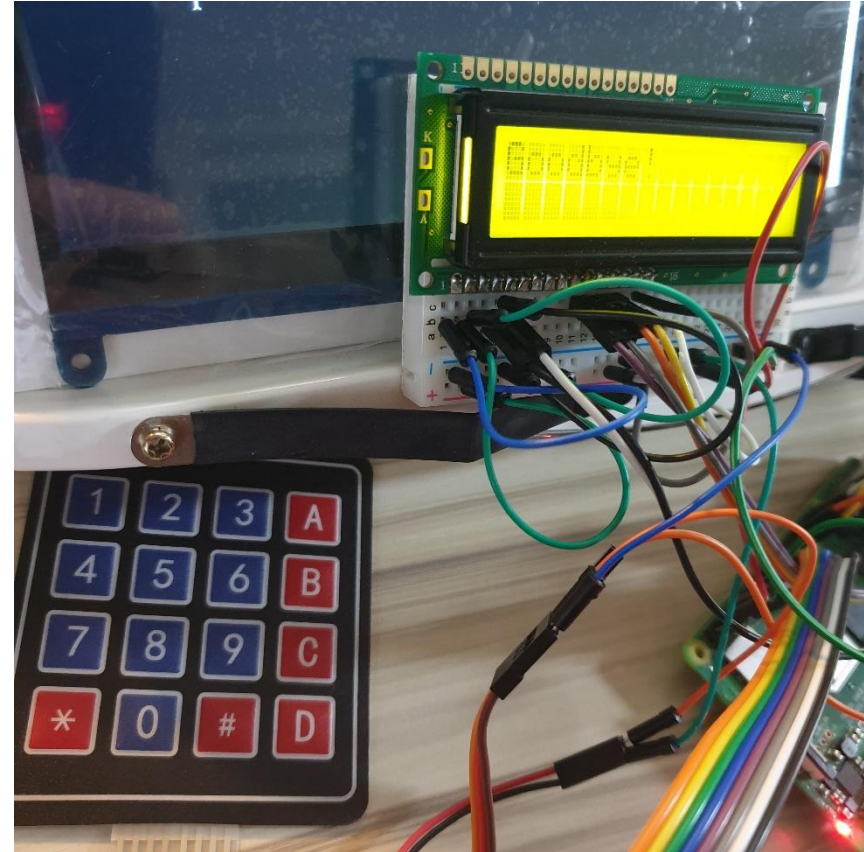
- 변수 sub_pw : 수신된 비밀번호 저장
- 변수 box1~3_pw : 보관함 1~3의 비밀번호
- 변수 box1~3_val : 보관함 1~3의 상태
("usable" : 비어있음 , "unusable" : 사용중)

```
104 #전역변수 선언
105 #수신한 otp저장
106 sub_pw = None
107
108 #보관함1~3의 비밀번호
109 box1_pw = None
110 box2_pw = None
111 box3_pw = None
112
113 #보관함1~3의 상태
114 box1_val = "usable"
115 box2_val = "usable"
116 box3_val = "usable"
```

RaspberryPi

키패드 모듈 + LCD 드라이버

- 도어락의 키패드입력 구현



RaspberryPi

키패드 모듈 코드

- 변수 press : 키패드 입력 번호 저장
- 변수 count : 키패드 입력 개수 저장
- 키패드 매핑 및 핀 넘버링

```
132 #키패드 모듈 초기화부분
133 press='' #입력 비밀번호 초기화
134 count=0 #입력 비밀번호 수
135
136 #키패드 모듈 매핑
137 KEYPAD = [
138     ["1", "2", "3", "A"],
139     ["4", "5", "6", "B"],
140     ["7", "8", "9", "C"],
141     ["*", "0", "#", "D"]
142 ]
143
144 #핀 넘버링
145 COL_PINS = [17, 15, 14, 4]
146 ROW_PINS = [24, 22, 27, 18]
147
148 #GPIO 매핑
149 factory = rpi_gpio.KeypadFactory()
150 keypad = factory.create_keypad(keypad=KEYPAD, row_pins=ROW_PINS, col_pins=COL_PINS)
151
152 # 키패드 버튼을 누를때마다 printKey 함수 호출
153 keypad.registerKeyPressHandler(printKey)
```

RaspberryPi

키패드 모듈 코드

- 키패드 '#'을 누르면 비밀번호 비교
- 키패드 '*'을 누르면 초기화
- 키패드 '0'~'9'를 누르면 press에 숫자를 담고 count의 값 증가

```
155 def printKey(key): #키패드 입력시 실행함수
156     lcd_byte(ord(key),LCD_CHR) #1602 LCD 하단 수 출력
157     global press #입력 비밀번호
158     global count #입력 비밀번호 수
159
160     if key=='#': # '#'은 비밀번호 입력 완료시
161         if box1_pw==press: # 1번 박스 비밀번호부터 비교
162             open_box1()
163         elif box2_pw==press:
164             open_box2()
165         elif box3_pw==press:
166             open_box3()
167         else:
168             fail() #모든 박스 비밀번호와 다를시 fail
169             press='' #초기화
170             count=0
171     elif key=='*': # '*'입력시 초기화
172         press=''
173         print(str(key))
174         count=0
175     else: #수 입력
176         count+=1
177         press=press+str(key)
178         print(press)
```


RaspberryPi

키패드 모듈 코드

- 서보모터를 제어하여 도어락 구현
- 변수 box1_val 을 사용중으로 변경

```
180      #1번 보관함 도어락 열림
181      def open_box1():
182          global box1_val
183          global p0
184          print("Box number 1 opened")
185          p0.ChangeDutyCycle(7.5) #개방
186          time.sleep(5)          #5초 대기
187          p0.ChangeDutyCycle(3)  #잠금
188          box1_val = "unusable" #1번 박스 상태 사용중으로 변경

205      def fail(): #실패
206          print("Password incorrect")
```

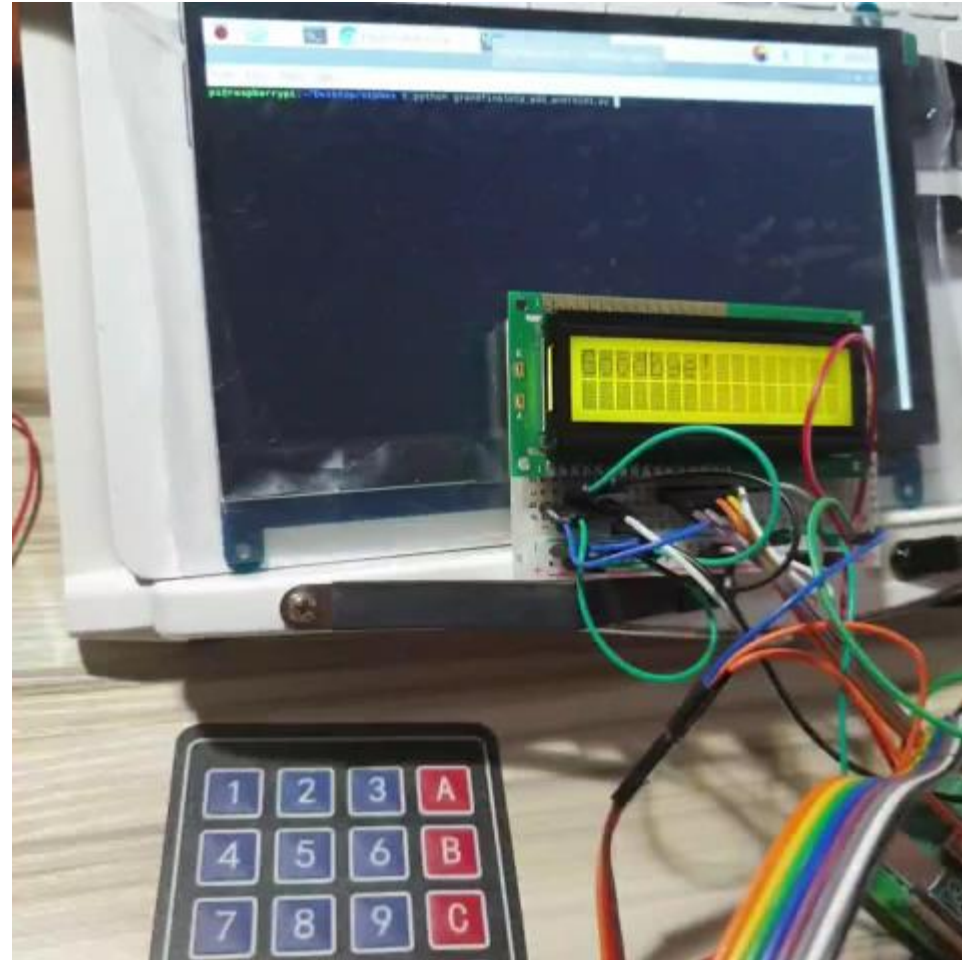
RaspberryPi

키패드 입력 사진

```
pi@raspberrypi: ~/Desktop
File Edit Tabs Help
pi@raspberrypi:~/Desktop/otpbox $ python grandfinalotp_add_android1.py
connected OK
Start subscribe : otpbox
Start subscribe : otpbox_android
Welcome
1
12
123
1234
Password incorrect
Subscribe from server
('Box1, Change password :', '8676', 'address :', '502')
8
86
867
8676
Box number 1 opened
```

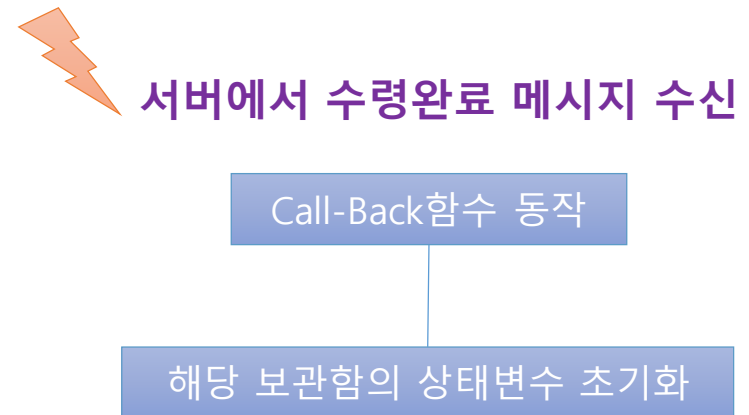
RaspberryPi

키패드 입력 시연영상



RaspberryPi

라즈베리파이 콜백함수 순서도



RaspberryPi

Mosquitto 서버 연결

- client 객체 생성
- Mosquitto 서버 ip의 1883번 포트 연결
- 콜백 함수 설정
- 비동기식 loop

```
86 # 새로운 클라이언트 생성
87 client = mqtt.Client()
88 # 콜백 함수 설정
89 client.on_connect = on_connect
90 client.on_disconnect = on_disconnect
91 client.on_subscribe = on_subscribe
92 client.on_message = on_message
93 # 집ip , port: 1883 에 연결
94 client.connect("211.197.225.191", 1883)
95
96 # 비동기식 loop
97 client.loop_start()
```

RaspberryPi

MQTT 수신 메시지 콜백함수 등록

- Mosquitto서버에서 subscribe하면 동작하는 함수
- Topic에 따라서 서버에서 온 메시지와 안드로이드에서 온 메시지를 구분함
- Topic : "otpbox" : 비밀번호 & 주소(JSON)
- Topic : "otpbox_android" : 수령한 사물함 번호

```
29 def on_message(client, userdata, msg):
30     topic = msg.topic
31     global box1_pw, box1_val, box2_pw, box2_
32     # 서버에서받은메시지
33     if (topic == "otpbox"):
34         print ("Subscribe from server")
35         # 박스선택 알고리즘
```

```
70     # 안드로이드에서받은메시지
71     elif (topic == "otpbox_android"):
72         print ("message from android")
73         data = msg.payload
74         if(data == "1"):
```

RaspberryPi

서버 메시지 수신 콜백함수

- "otpbox" 토픽 메시지가 들어올 경우 콜백함수 실행
- 보관함 상태 변수 box1~3_val이 모두 사용중일 경우 메시지 출력
- 사용 가능한 보관함이 있을 경우 수신한 JSON데이터 추출 (비밀번호, 주소)

```
# 서버에서받은메시지
if (topic == "otpbox"):
    print ("Subscribe from server")
    # 박스선택 알고리즘
    if (box1_val == "unusable" and box2_val == "unusable" and box3_val == "unusable"):
        print ("All box unusable")
    else:
        global sub_pw
        # JSON 데이터 추출
        data = msg.payload
        json_data = json.loads(data)
        sub_pw = str(json_data['password'])
        add = str(json_data['address'])
```

RaspberryPi

서버 메시지 수신 콜백함수

- 비어있는 박스가 있다면 비밀번호를 변경하고 상태변수를 사용중으로 변경
- 비밀번호, 주소, 해당박스번호를 JSON으로 파싱하여 publish (안드로이드에 전송)

```
elif (box3_val == "usable"):
    print ("Box3, Change password :", sub_pw, "address :", add)
    box3_pw = sub_pw
    temp = 3
    box3_val = "unusable"
else:
    print ("Error : select box")
    temp = 4

if (temp != 4):
    dataJson = {"password": sub_pw, "address": add, "boxnumber": str(temp)}
    parseJson = json.dumps(dataJson)
    client.publish("otpbox_rasp", parseJson, 1)
```


RaspberryPi

서버 메시지 수신 콜백함수 사진

- password : 3687
- address : 501

```
pi@raspberrypi:~/Desktop/otpbox $ python grandfinalotp_add_android1.py
connected OK
Start subscribe : otpbox
Start subscribe : otpbox_android
Welcome
Subscribe from server
('Box1, Change password :', '3687', 'address :', '501')
```

RaspberryPi

안드로이드 메시지 수신 콜백함수

- "otpbox_android" 토픽 메시지가 들어올 경우 콜백함수 실행
- 보관함 번호에 따라서 해당 보관함의 상태변수를 사용 가능으로 변경

```
# 안드로이드에서받은메시지
elif (topic == "otpbox_android"):
    print ("message from android")
    data = msg.payload
    if(data == "1"):
        box1_val = "usable"
        print("Box 1 changed status : usable")
    elif (data == "2"):
        box2_val = "usable"
        print("Box 2 changed status : usable")
    elif (data == "3"):
        box3_val = "usable"
        print("Box 3 changed status : usable")
    else:
        print("Error : subscribe android")
```

RaspberryPi

안드로이드 메시지 콜백함수 사진

- 1번박스의 상태변수 초기화

```
pi@raspberrypi:~/Desktop/otpbox $ python grandfinalotp_add_android1.py
connected OK
Start subscribe : otpbox
Start subscribe : otpbox_android
Welcome
Subscribe from server
('Box1, Change password :', '4208', 'address :', '501')
message from android
Box 1 changed status : usable
```

Editor : Android Studio

6. Android

---> 모바일 애플리케이션

Mosquitto서버에서 OTP와 배송 주소를 수신하면 해당 주소 택배 수령자의 모바일 애플리케이션 화면에 OTP를 띄운다.

고객이 택배를 수령한 후 '수령 완료'버튼을 누르면 도어락이 비었다는 Message를 Mosquitto서버로 보낸다.

Android

라이브러리 추가

- Org.eclipse.paho.. 추가

Modules	Declared Dependencies	
+ -	+ -	
<All Modules>	Dependency	Scope
app	appcompat:1.0.2	implementation
	constraintlayout:1.1.3	implementation
	espresso-core:3.1.1	androidTestImplementation
	junit:1.1.0	androidTestImplementation
	junit:4.12	testImplementation
	org.eclipse.paho.android.service:1.1.1	implementation
	org.eclipse.paho.client.mqttv3:1.2.0	implementation
	libs	implementation

Android

Mosquitto 서버 연결

- Mosquitto서버 ip의 1883번 포트로 클라이언트 생성
- 토큰 생성 및 연결

```
//MQTT클라이언트 객체 생성
mqttAndroidClient = new MqttAndroidClient( context: this, serverURI: "tcp://" + "211.197.225.191" + ":1883", MqttClient.generateClientId());

try {
    //mqttoken 생성
    IMqttToken token = mqttAndroidClient.connect(getMqttConnectionOption());
    token.setActionCallback(new IMqttActionListener() {
```

Android

안드로이드 콜백함수 등록

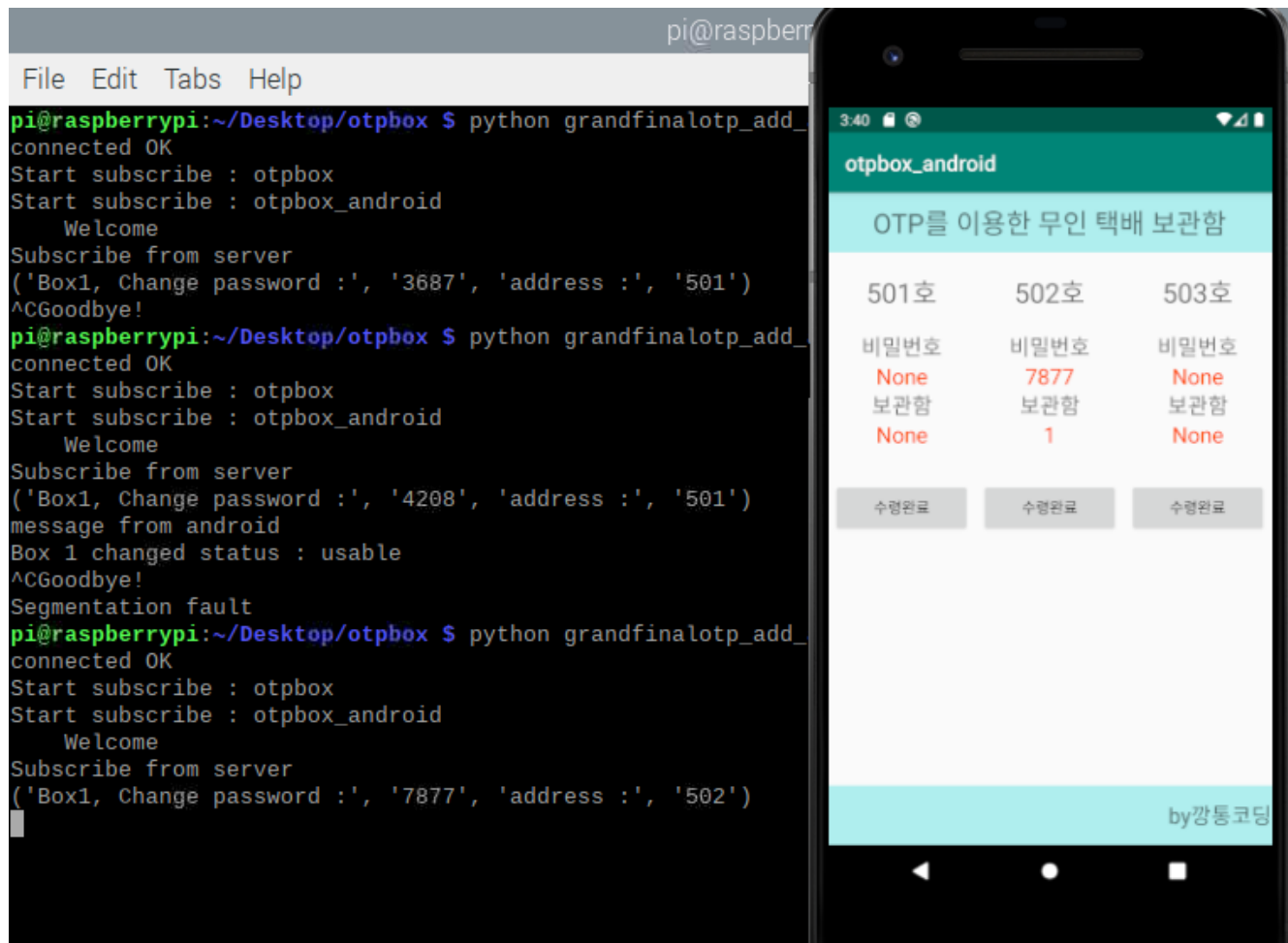
- 라즈베리에서 pub한 메시지 수신 시 작동하는 콜백함수
- JSON 추출(비밀번호, 보관함번호, 주소)
- 해당 주소에 비밀번호와 보관함번호 출력

```
//연결에 성공한경우
public void onSuccess(IMqttToken asyncActionToken) {
    mqttAndroidClient.setBufferOpts(getDisconnectedBufferOptions());
    Log.e( tag: "Connect_success", msg: "Success");
    // 익명함수 이용 구독 콜백함수
    // 라즈베리파이에서 publish한 topic : otpbox_rasp 수신
    try {
        mqttAndroidClient.subscribe( topicFilter: "otpbox_rasp", qos: 0, new IMqttMessageListener() {
            @Override
            public void messageArrived(String topic, MqttMessage message) throws Exception {
                String msg = new String(message.getPayload());
                Log.e( tag: "arrive message : ", msg);
                JSONObject jsonObject = new JSONObject(msg);
                password = jsonObject.getString( name: "password");
                boxnumber = jsonObject.getString( name: "boxnumber");
                address = jsonObject.getString( name: "address");
                if(address.equals("501")){
                    tv_pw_501.setText(password);
                    tv_box_501.setText(boxnumber);
                }
            }
        });
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

Android

안드로이드 콜백함수 사진

- 해당 주소에 비밀번호와 보관함번호 출력



Android

안드로이드 수령완료 버튼리스너 등록

- 수령완료 버튼 클릭시 해당 택배보관함 번호 publish

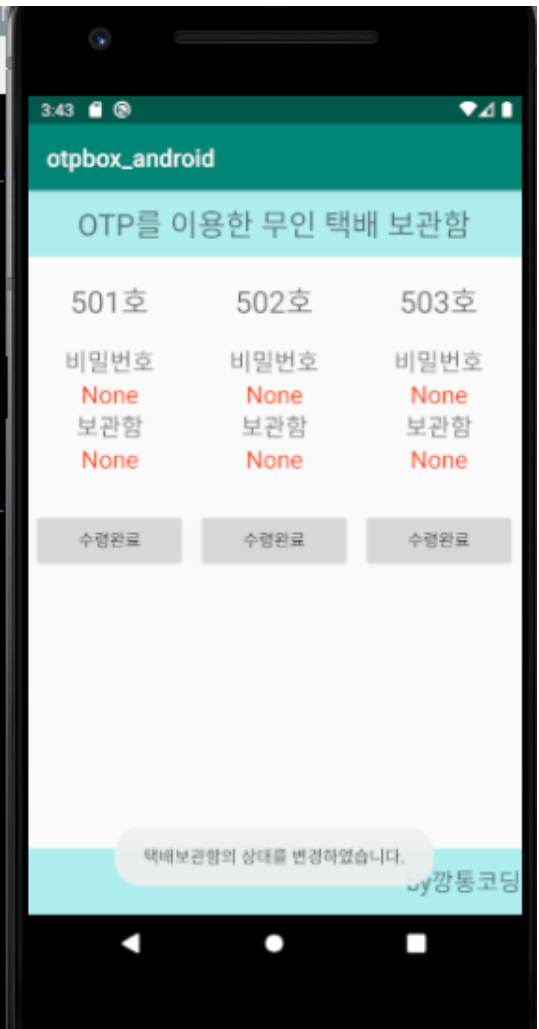
```
//익명함수 이용 버튼리스너
btn_sendrasp_501.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        try {
            String temp_boxnumber = String.valueOf(tv_box_501.getText());
            if(temp_boxnumber.equals("None")){
                Toast.makeText(getApplicationContext(), text: "받은 택배가 없습니다.", Toast.LENGTH_SHORT).show();
            }
            else{
                //otpbox_android 토픽으로 수령한 box number publish
                mqttAndroidClient.publish( topic: "otpbox_android", temp_boxnumber.getBytes(), qos: 0 , retained: false );
                tv_pw_501.setText("None");
                tv_box_501.setText("None");
                Toast.makeText(getApplicationContext(), text: "택배보관함의 상태를 변경하였습니다.", Toast.LENGTH_SHORT).show();
            }
        }
    }
});
```

Node.js Server

안드로이드 수령완료 버튼

- 토스트 메시지 출력
- 해당 택배보관함 상태변수 변경

```
File Edit Tabs Help
('Box1, Change password :', '3687', 'address :', '501')
^CGoodbye!
pi@raspberrypi:~/Desktop/otpbox $ python grandfinalotp_add_
connected OK
Start subscribe : otpbox
Start subscribe : otpbox_android
Welcome
Subscribe from server
('Box1, Change password :', '4208', 'address :', '501')
message from android
Box 1 changed status : usable
^CGoodbye!
Segmentation fault
pi@raspberrypi:~/Desktop/otpbox $ python grandfinalotp_add_
connected OK
Start subscribe : otpbox
Start subscribe : otpbox_android
Welcome
Subscribe from server
('Box1, Change password :', '7877', 'address :', '502')
message from android
Box 1 changed status : usable
Subscribe from server
('Box1, Change password :', '6628', 'address :', '502')
Subscribe from server
('Box2, Change password :', '4948', 'address :', '503')
message from android
Box 1 changed status : usable
message from android
Box 2 changed status : usable
```



Node.js Server

라즈베리파이 + 안드로이드 시연영상



OTP를 이용한 무인 택배 보관함

team광통코딩

