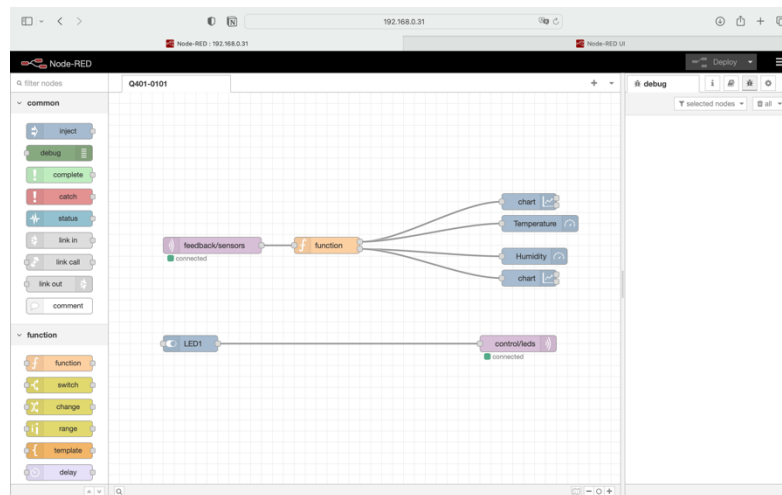| |
|---|
| **การสร้าง MQTT Server บน Raspberry Pi เพื่อใช้งาน Chatbot LINE ในฟาร์มอัจฉริยะ**
Chatbot LINE from Raspberry Pi MQTT Server for Smart Farming |
| **ขื่อ-สกุล : B6304577 นายภานพงศ์ แคนอินทร์** |
| **6/6 - คำถามท้ายบทเพื่อทดสอบความเข้าใจ** |

Quiz_401 – RPi Smart Farm

- แสดงรูป โปรแกรม ของผลการทำงานตามหัวข้อ การสร้าง UI ด้วย Node-RED สำหรับฟาร์มอัจฉริยะ

Capture Node-RED Flow

Node-RED Code

```
[
    {
        "id": "8982e95ddaea1943",
        "type": "tab",
        "label": "Q401-0101",
        "disabled": false,
        "info": ""
    },
    {
        "id": "287869c0be568bc8",
        "type": "function",
        "z": "8982e95ddaea1943",
        "name": "",
        "func": "//   Tempp=38.20,Humid=73.52 \n\nvar msg1   = {};\nvar msg2 = {};\n\nvar
output =  msg.payload.split(\",\");\n\nvar sTempp = output[0].split(\"=\");\nmsg1.payload =
sTempp[1];\nmsg1.topic = 'Temperature';\n\nvar   sHumid =
```

```
output[1].split(\"=\");\nmsg2.payload = sHumid[1];\nmsg2.topic = 'Humidity';\n\nreturn
[msg1,msg2];",
        "outputs": 2,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 400,
        "y": 280,
        "wires": [
            [
                "314e6fc73de0b3b2",
                "ffdc2f39a719cb9f"
            ],
            [
                "922989963c892994",
                "c87d555924e87fbc"
            ]
        ]
    },
    {
        "id": "922989963c892994",
        "type": "ui_gauge",
        "z": "8982e95ddaea1943",
        "tab": "5ab59fbbd41d64e9",
        "name": "Humidity",
        "group": "Farm",
        "order": "3",
        "format": "{{value}}",
        "min": 0,
        "max": "100",
        "x": 780,
        "y": 300,
        "wires": []
    },
    {
        "id": "1ee84dc4716b94ae",
        "type": "ui_switch",
        "z": "8982e95ddaea1943",
```

```
        "tab": "5ab59fbbd41d64e9",
        "name": "LED1",
        "topic": "topic",
        "group": "Farm",
        "order": "0",
        "onvalue": "ON1",
        "offvalue": "OFF1",
        "x": 150,
        "y": 460,
        "wires": [
            [
                "172f630b25f47c66"
            ]
        ]
    },
    {
        "id": "271f9555b85e5f30",
        "type": "mqtt in",
        "z": "8982e95ddaea1943",
        "name": "",
        "topic": "feedback/sensors",
        "qos": "0",
        "datatype": "auto-detect",
        "broker": "d9b6929cc83977a5",
        "nl": false,
        "rap": true,
        "rh": 0,
        "inputs": 0,
        "x": 190,
        "y": 280,
        "wires": [
            [
                "287869c0be568bc8"
            ]
        ]
    },
    {
        "id": "172f630b25f47c66",
        "type": "mqtt out",
```

```
        "z": "8982e95ddaea1943",
        "name": "",
        "topic": "control/leds",
        "qos": "0",
        "retain": "false",
        "respTopic": "",
        "contentType": "",
        "userProps": "",
        "correl": "",
        "expiry": "",
        "broker": "d9b6929cc83977a5",
        "x": 750,
        "y": 460,
        "wires": []
    },
    {
        "id": "c87d555924e87fbc",
        "type": "ui_chart",
        "z": "8982e95ddaea1943",
        "tab": "5ab59fbbd41d64e9",
        "name": "",
        "group": "Farm",
        "order": "4",
        "interpolate": "linear",
        "nodata": "No Data",
        "removeOlder": 1,
        "removeOlderUnit": "86400",
        "x": 770,
        "y": 340,
        "wires": [
            [],
            []
        ]
    },
    {
        "id": "314e6fc73de0b3b2",
        "type": "ui_chart",
        "z": "8982e95ddaea1943",
        "tab": "5ab59fbbd41d64e9",
```

```
        "name": "",
        "group": "Farm",
        "order": "2",
        "interpolate": "linear",
        "nodata": "No Data",
        "removeOlder": 1,
        "removeOlderUnit": "86400",
        "x": 770,
        "y": 200,
        "wires": [
            [],
            []
        ]
    },
    {
        "id": "ffdc2f39a719cb9f",
        "type": "ui_gauge",
        "z": "8982e95ddaea1943",
        "tab": "5ab59fbbd41d64e9",
        "name": "Temperature",
        "group": "Farm",
        "order": "1",
        "format": "{{value}}",
        "min": 0,
        "max": "100",
        "x": 790,
        "y": 240,
        "wires": []
    },
    {
        "id": "5ab59fbbd41d64e9",
        "type": "ui_tab",
        "name": "Farm",
        "icon": "dashboard",
        "order": "1"
    },
    {
        "id": "d9b6929cc83977a5",
        "type": "mqtt-broker",
```

```
        "name": "",
        "broker": "192.168.0.31",
        "port": "1883",
        "clientid": "",
        "autoConnect": true,
        "usetls": false,
        "protocolVersion": "4",
        "keepalive": "60",
        "cleansession": true,
        "birthTopic": "",
        "birthQos": "0",
        "birthRetain": "false",
        "birthPayload": "",
        "birthMsg": {},
        "closeTopic": "",
        "closeQos": "0",
        "closeRetain": "false",
        "closePayload": "",
        "closeMsg": {},
        "willTopic": "",
        "willQos": "0",
        "willRetain": "false",
        "willPayload": "",
        "willMsg": {},
        "userProps": "",
        "sessionExpiry": ""
    }
]
```

Code Arduino

```
#include <WiFi.h>

#include <PubSubClient.h>

#include <DHT.h> //https://www.arduinolibraries.info/libraries/dht-sensor-library

#define DHT_SENSOR_PIN  4 // ESP32 pin GIOP4 connected to DHT22 sensor

#define DHT_SENSOR_TYPE DHT22

#include <ArduinoJson.h>
```

```
#define LED0_Pin 2

#define SW0_Pin  15


int SW0_Status = 99;

unsigned long lastMsg = 0;

char msg[50];


DHT dht_sensor(DHT_SENSOR_PIN, DHT_SENSOR_TYPE);


const char* ssid = "KOi";

const char* password = "13062544";

const char *mqtt_broker = "192.168.0.31";

const char *mqtt_username = "mymqtt";

const char *mqtt_password = "mymqtt";

const int mqtt_port = 1883;




WiFiClient espClient;

PubSubClient client(espClient);


void connectToWiFi() {
  byte mac[6];
  delay(10);
  // We start by connecting to a WiFi network
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.mode(WIFI_STA);
```

```
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {

    Serial.print(".");

  }


  WiFi.macAddress(mac);

  Serial.println("");

  Serial.println("WiFi connected");

  Serial.println("IP address: ");

  Serial.println(WiFi.localIP());

}


void connectToMQTTBroker() {

 byte mqttFailCount = 0;

 byte tooManyFailures = 10;

 // Loop until we're reconnected

 while (!client.connected()) {

   if (mqttFailCount <= tooManyFailures) {

     String client_id = "esp32-client-";

     client_id += String(WiFi.macAddress());

     Serial.printf("The client %s connects to the public mqtt broker\n", client_id.c_str());

     if (client.connect("9z5srmKS2WAoFGYQaam1Py", mqtt_username, mqtt_password)) {

       Serial.println("Public Homezassistant mqtt broker connected");

       // Subscribe

       client.subscribe("feedback/status");

       client.subscribe("control/leds");

       delay(100);

     } else {

       mqttFailCount ++;
```

```
      Serial.print("Failed. Count = ");

      Serial.println(mqttFailCount);

      Serial.println("...trying again in 5 seconds");

      // Wait 5 seconds before retrying

      delay(5000);

    }

   }

 }

}


void MqttReceiverCallback(char* topic, byte * payload, unsigned int length) {

  char myPayLoad[50];

  Serial.print("Message arrived [");

  Serial.print(topic);

  Serial.print("] ");

  for (int i = 0; i < length; i++)

  { Serial.print((char)payload[i]);

    myPayLoad[i]     = payload[i];

    myPayLoad[i + 1]  = '\0'; // End of String

  }

  Serial.print("\n ---> ");    Serial.println(myPayLoad);

  myPayLoad[4] = '\0';  // String lessthan 4 Charector

  if ((String)myPayLoad == "ON1")    digitalWrite(LED0_Pin, HIGH);

  if ((String)myPayLoad == "OFF1")   digitalWrite(LED0_Pin, LOW);

}




void setup() {
```

```
  Serial.begin(115200);

  pinMode(LED0_Pin, OUTPUT);

  connectToWiFi();

  client.setServer(mqtt_broker, mqtt_port);

  client.setCallback(MqttReceiverCallback);

  dht_sensor.begin(); // initialize the DHT sensor

}


void loop() {


  if (!client.connected()) {

    connectToMQTTBroker();

  }

  client.loop();


  if (digitalRead(SW0_Pin) != SW0_Status) {

    SW0_Status = digitalRead(SW0_Pin);

    Serial.println(SW0_Status == HIGH ? "Status Switch = OFF" : "Status Switch = ON");

    client.publish("feedback/switch", (SW0_Status == HIGH ? "SW_OFF" : "SW_ON"));

    delay(100);

  }


  unsigned long now = millis();

  if (now - lastMsg > 1000) {


    // read humidity

    float humi  = dht_sensor.readHumidity();

    // read temperature in Celsius

    float tempC = dht_sensor.readTemperature();
```

```
  // read temperature in Fahrenheit

  float tempF = dht_sensor.readTemperature(true);


  // check whether the reading is successful or not

  if ( isnan(tempC) || isnan(tempF) || isnan(humi)) {

    Serial.println("Failed to read from DHT sensor!");

  } else {

    snprintf (msg, 50, "Tempp=%0.2f,Humid=%0.2f", tempC, humi);



    if (client.connected()) {

      client.publish("feedback/sensors", msg);

      Serial.println(msg);

      Serial.println("MQTT: Send Data!!!");

      Serial.println();

    }

  }

  lastMsg = millis();

 }

}
```
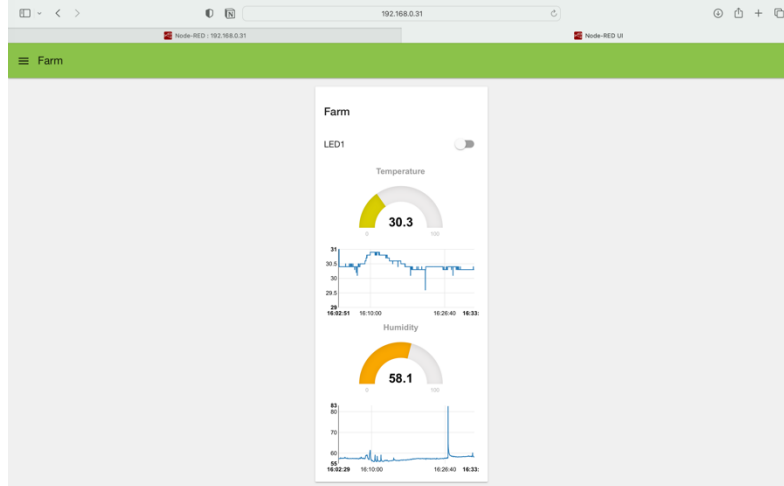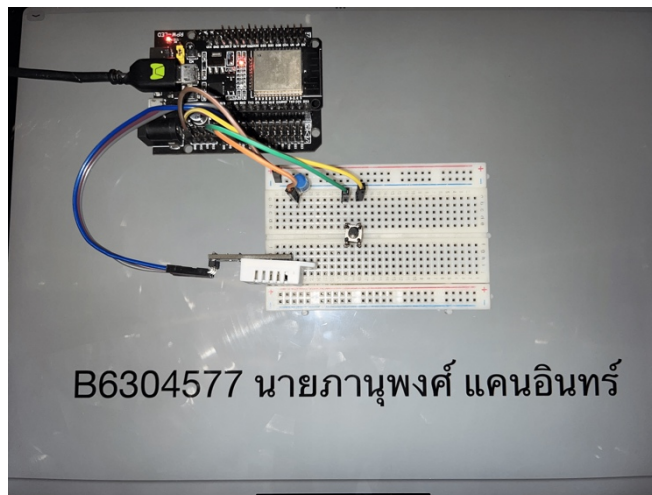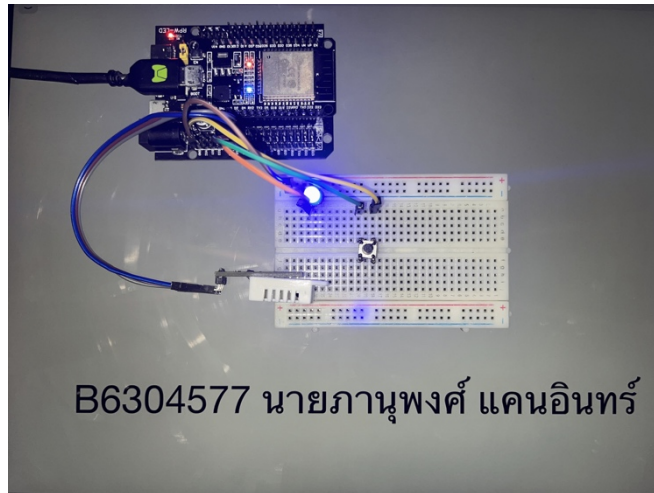
รูปการทดสอบ 1: UI Result
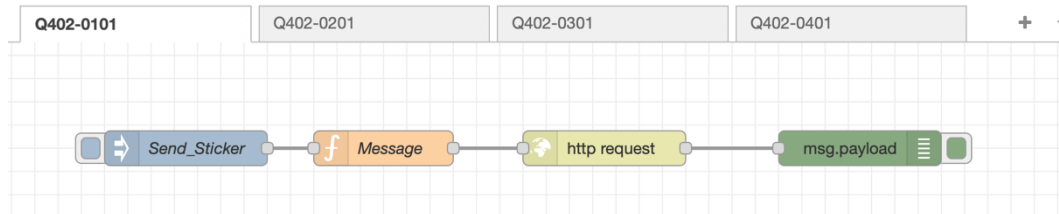
รูปการทดสอบ 1

รูปการทดสอบ 2
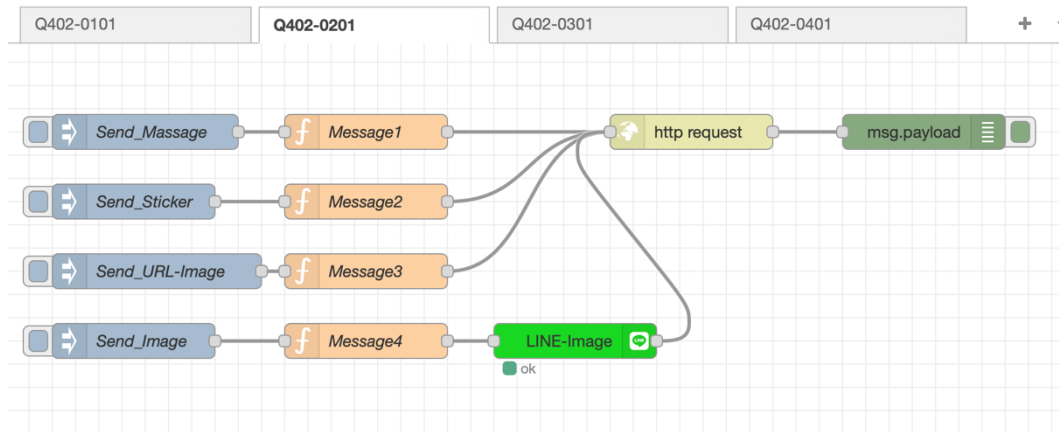
รูปการทดสอบ 3

Quiz_402 – LINE Notify

- **แสดงรูป โปรแกรม ของผลการทำงานตามหัวข้อ การส่งข้อความด้วย Node-RED สำหรับฟาร์มอัจฉริยะ**

Capture Node-RED Flow

mission1/4

mission 2/4

mission 3/4
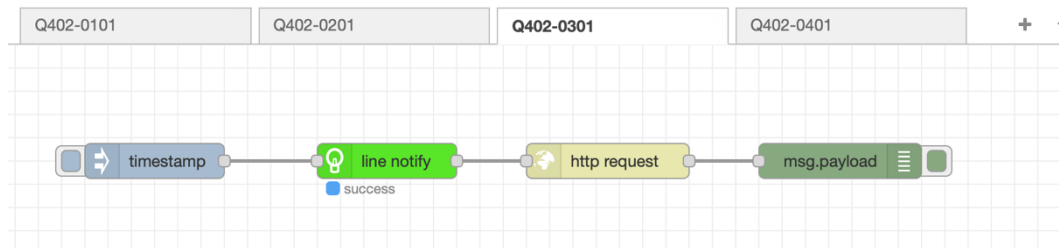
mission 4/4
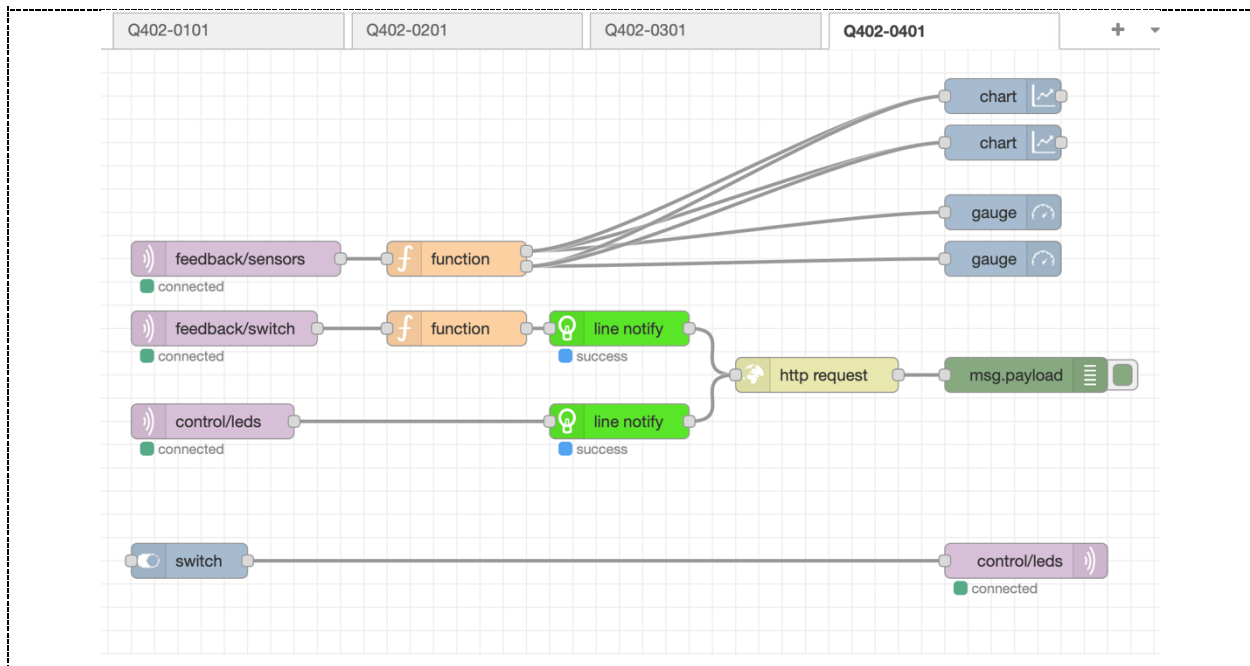
Node-RED Code

mission1/4

```
[
    {
        "id": "18068883f6581588",
        "type": "tab",
        "label": "Q402-0101",
        "disabled": false,
        "info": ""
    },
    {
        "id": "a809186e12e39a1f",
        "type": "debug",
        "z": "18068883f6581588",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "statusVal": "",
        "statusType": "auto",
        "x": 750,
```

```
        "y": 100,
        "wires": []
    },
    {
        "id": "209a69d8f2ab744b",
        "type": "function",
        "z": "18068883f6581588",
        "name": "Message",
        "func": "msg.token = 'FYfx1woSF3SYOZexMC7ia5jX6IaOQlO2EHL7KDeCsut';\nmsg.message=
'Hello';\nmsg.stickerPackageId = 6359;\nmsg.stickerId = 11069852;\n\nmsg.headers = {\n
'content-type':'application/x-www-form-urlencoded',\n 'Authorization':'Bearer ' +
msg.token\n};\n\nmsg.payload ={\n 'message': msg.message,\n
'stickerPackageId':msg.stickerPackageId,\n'stickerId':msg.stickerId\n};\nreturn msg;\n",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 340,
        "y": 100,
        "wires": [
            [
                "d06fca3f9d1231cc"
            ]
        ]
    },
    {
        "id": "6a5e1d098fbfd185",
        "type": "inject",
        "z": "18068883f6581588",
        "name": "Send_Sticker",
        "props": [
            {
                "p": "payload"
            },
            {
                "p": "topic",
                "vt": "str"
            }
        ],
```

```json
        "repeat": "",
        "crontab": "",
        "once": false,
        "onceDelay": 0.1,
        "topic": "",
        "payload": "",
        "payloadType": "date",
        "x": 170,
        "y": 100,
        "wires": [
            [
                "209a69d8f2ab744b"
            ]
        ]
    },
    {
        "id": "d06fca3f9d1231cc",
        "type": "http request",
        "z": "18068883f6581588",
        "name": "",
        "method": "POST",
        "ret": "txt",
        "paytoqs": "ignore",
        "url": "https://notify-api.line.me/api/notify",
        "tls": "",
        "persist": false,
        "proxy": "",
        "insecureHTTPParser": false,
        "authType": "",
        "senderr": false,
        "headers": [],
        "x": 530,
        "y": 100,
        "wires": [
            [
                "a809186e12e39a1f"
            ]
        ]
    }
```

```
]

mission2/4

[
    {
        "id": "bfdfb28d613836ee",
        "type": "tab",
        "label": "Q402-0201",
        "disabled": false,
        "info": ""
    },
    {
        "id": "99081251955e2969",
        "type": "debug",
        "z": "bfdfb28d613836ee",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "targetType": "msg",
        "statusVal": "",
        "statusType": "auto",
        "x": 850,
        "y": 80,
        "wires": []
    },
    {
        "id": "ca51dabcc70e764e",
        "type": "function",
        "z": "bfdfb28d613836ee",
        "name": "Message1",
        "func": "var token ='FYfx1woSF3SYOZexMC7ia5jX6IaOQlO2EHL7KDeCsut';\nvar message =
'LINE-Massage From Node-RED';\n\nmsg.headers = {\n 'content-type':'application/x-www-form-
urlencoded',\n'Authorization':'Bearer ' + token\n};\n\nmsg.payload = {\n 'message':
message\n};\nreturn msg;\n",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
```

```json
        "finalize": "",
        "libs": [],
        "x": 370,
        "y": 80,
        "wires": [
            [
                "1d6974daa5523c41"
            ]
        ]
    },
    {
        "id": "45c5828c24614fc5",
        "type": "inject",
        "z": "bfdfb28d613836ee",
        "name": "Send_Massage",
        "props": [
            {
                "p": "payload"
            },
            {
                "p": "topic",
                "vt": "str"
            }
        ],
        "repeat": "",
        "crontab": "",
        "once": false,
        "onceDelay": 0.1,
        "topic": "",
        "payload": "",
        "payloadType": "date",
        "x": 180,
        "y": 80,
        "wires": [
            [
                "ca51dabcc70e764e"
            ]
        ]
    },
```

```json
    {
        "id": "42b9851aad6c539c",
        "type": "inject",
        "z": "bfdfb28d613836ee",
        "name": "Send_Sticker",
        "props": [
            {
                "p": "payload"
            },
            {
                "p": "topic",
                "vt": "str"
            }
        ],
        "repeat": "",
        "crontab": "",
        "once": false,
        "onceDelay": 0.1,
        "topic": "",
        "payload": "",
        "payloadType": "date",
        "x": 170,
        "y": 140,
        "wires": [
            [
                "7db1785add669cde"
            ]
        ]
    },
    {
        "id": "e8a63beb9ce948c9",
        "type": "function",
        "z": "bfdfb28d613836ee",
        "name": "Message3",
        "func": "\nvar url_img = 'https://bear.org/wp-content/uploads/2008/01/Griz-family-danger-ahead.jpg';\nvar token = 'FYfx1woSF3SYOZexMC7ia5jX6IaOQlO2EHL7KDeCsut';\nvar message = 'LINE-URL Image';\n\nmsg.headers = {\n 'content-type':'application/x-www-form-urlencoded',\n'Authorization':'Bearer ' + token\n};\n\nmsg.payload = {\n'message':message,\n\t'imageThumbnail':url_img,\n\t'imageFullsize':url_img\n };\n\nreturn msg;\n",
```

```
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 370,
        "y": 200,
        "wires": [
            [
                "1d6974daa5523c41"
            ]
        ]
    },
    {
        "id": "c6cc0446edd6c42f",
        "type": "inject",
        "z": "bfdfb28d613836ee",
        "name": "Send_URL-Image",
        "props": [
            {
                "p": "payload"
            },
            {
                "p": "topic",
                "vt": "str"
            }
        ],
        "repeat": "",
        "crontab": "",
        "once": false,
        "onceDelay": 0.1,
        "topic": "",
        "payload": "",
        "payloadType": "date",
        "x": 190,
        "y": 200,
        "wires": [
            [
                "e8a63beb9ce948c9"
```

```
                ]
            ]
        },
        {
            "id": "1f1e97eda7559188",
            "type": "inject",
            "z": "bfdfb28d613836ee",
            "name": "Send_Image",
            "props": [
                {
                    "p": "payload"
                },
                {
                    "p": "topic",
                    "vt": "str"
                }
            ],
            "repeat": "",
            "crontab": "",
            "once": false,
            "onceDelay": 0.1,
            "topic": "",
            "payload": "Hallo ImageTest",
            "payloadType": "str",
            "x": 170,
            "y": 260,
            "wires": [
                [
                    "d35f0cee69c584d2"
                ]
            ]
        },
        {
            "id": "7db1785add669cde",
            "type": "function",
            "z": "bfdfb28d613836ee",
            "name": "Message2",
            "func": "var token ='FYfx1woSF3SYOZexMC7ia5jX6IaOQlO2EHL7KDeCsut';\nvar message =
'LINE-Sticker';\nvar stickerPackageId =8522;\nvar stickerId = 16581266;\n\nmsg.headers = {\n
'content-type':'application/x-www-form-urlencoded',\n 'Authorization':'Bearer ' +
```

```
token\n};\n\nmsg.payload = {\n 'message': message,\n'stickerPackageId':stickerPackageId,\n
'stickerId':stickerId\n};\n\nreturn msg;\n",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 370,
        "y": 140,
        "wires": [
            [
                "1d6974daa5523c41"
            ]
        ]
    },
    {
        "id": "d831d3f0ce4f7c5e",
        "type": "NotifyImage",
        "z": "bfdfb28d613836ee",
        "name": "LINE-Image",
        "tmsg": "",
        "imgfile": "https://nypost.com/wp-content/uploads/sites/2/2021/12/cat-
mad_13.jpg?quality=75&strip=all",
        "AccToken": "FYfx1woSF3SYOZexMC7ia5jX6IaOQlO2EHL7KDeCsut",
        "x": 550,
        "y": 260,
        "wires": [
            [
                "1d6974daa5523c41"
            ]
        ]
    },
    {
        "id": "d35f0cee69c584d2",
        "type": "function",
        "z": "bfdfb28d613836ee",
        "name": "Message4",
        "func": "var message = 'Send Image Data';\nmsg.payload =message;\nreturn msg;",
        "outputs": 1,
```

```
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 370,
        "y": 260,
        "wires": [
            [
                "d831d3f0ce4f7c5e"
            ]
        ]
    },
    {
        "id": "1d6974daa5523c41",
        "type": "http request",
        "z": "bfdfb28d613836ee",
        "name": "",
        "method": "POST",
        "ret": "txt",
        "paytoqs": "ignore",
        "url": "https://notify-api.line.me/api/notify",
        "tls": "",
        "persist": false,
        "proxy": "",
        "insecureHTTPParser": false,
        "authType": "",
        "senderr": false,
        "headers": [],
        "x": 650,
        "y": 80,
        "wires": [
            [
                "99081251955e2969"
            ]
        ]
    }
]
```

mission3/4

```
[
    {
        "id": "6790d352.2280ac",
        "type": "tab",
        "label": "Q402-0301",
        "disabled": false,
        "info": ""
    },
    {
        "id": "ca2e8cd2.5bd98",
        "type": "line-notify",
        "z": "6790d352.2280ac",
        "name": "",
        "message": "Send \"Test Massage\" from RaspberryPi",
        "contentType": "message",
        "imageThumbnail": "",
        "imageUrl": "",
        "sticker": "default",
        "stickerPackageId": "11",
        "stickerId": "1",
        "silent": false,
        "creds": "591415dad7908dc3",
        "x": 340,
        "y": 100,
        "wires": [
            [
                "aecf6b014cdb95b7"
            ]
        ]
    },
    {
        "id": "b23ae221.44489",
        "type": "debug",
        "z": "6790d352.2280ac",
        "name": "",
        "active": true,
        "tosidebar": true,
```

```
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "targetType": "msg",
        "statusVal": "",
        "statusType": "auto",
        "x": 730,
        "y": 100,
        "wires": []
    },
    {
        "id": "88d3a65e.a258c8",
        "type": "inject",
        "z": "6790d352.2280ac",
        "name": "",
        "props": [
            {
                "p": "payload"
            },
            {
                "p": "topic",
                "vt": "str"
            }
        ],
        "repeat": "",
        "crontab": "",
        "once": false,
        "onceDelay": 0.1,
        "topic": "",
        "payload": "",
        "payloadType": "date",
        "x": 140,
        "y": 100,
        "wires": [
            [
                "ca2e8cd2.5bd98"
            ]
        ]
    },
```

```
    {
        "id": "aecf6b014cdb95b7",
        "type": "http request",
        "z": "6790d352.2280ac",
        "name": "",
        "method": "POST",
        "ret": "txt",
        "paytoqs": "ignore",
        "url": "https://notify-api.line.me/api/notify",
        "tls": "",
        "persist": false,
        "proxy": "",
        "insecureHTTPParser": false,
        "authType": "",
        "senderr": false,
        "headers": [],
        "x": 530,
        "y": 100,
        "wires": [
            [
                "b23ae221.44489"
            ]
        ]
    },
    {
        "id": "591415dad7908dc3",
        "type": "linetoken",
        "name": "M4Q402"
    }
]
mission1/4
[
    {
        "id": "0f1e807e520403ff",
        "type": "tab",
        "label": "Q402-0401",
        "disabled": false,
        "info": ""
```

```
    },
    {
        "id": "26213b1ceab8d687",
        "type": "line-notify",
        "z": "0f1e807e520403ff",
        "name": "",
        "message": "Sensor ActiveXXXXXXXXXXXXXXXX",
        "contentType": "message",
        "imageThumbnail": "",
        "imageUrl": "",
        "sticker": "default",
        "stickerPackageId": "11",
        "stickerId": "1",
        "silent": false,
        "creds": "591415dad7908dc3",
        "x": 460,
        "y": 240,
        "wires": [
            [
                "e759f324160a52de"
            ]
        ]
    },
    {
        "id": "55c7da970b5006a2",
        "type": "debug",
        "z": "0f1e807e520403ff",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "targetType": "msg",
        "statusVal": "",
        "statusType": "auto",
        "x": 810,
        "y": 280,
        "wires": []
```

```
        },
        {
            "id": "51ea9635d139d70e",
            "type": "ui_switch",
            "z": "0f1e807e520403ff",
            "tab": "4410af667a3d8692",
            "name": "",
            "topic": "topic",
            "group": "Q402",
            "order": 2,
            "onvalue": "ON1",
            "offvalue": "OFF1",
            "x": 90,
            "y": 440,
            "wires": [
                [
                    "52607273423ad64e"
                ]
            ]
        },
        {
            "id": "e7793cb2f893f284",
            "type": "function",
            "z": "0f1e807e520403ff",
            "name": "",
            "func": "//Tempp=38.20,Humid=73.52 \n\nvar msg1 = {};\nvar msg2 = {};\n\nvar output
=msg.payload.split(\",\");\n\nvar sTempp = output[0].split(\"=\");\nmsg1.payload =
sTempp[1];\nmsg1.topic= 'Temperature';\n\nvar sHumid = output[1].split(\"=\");\nmsg2.payload
= sHumid[1];\nmsg2.topic ='Humidity';\n\nreturn[msg1,msg2];",
            "outputs": 2,
            "noerr": 0,
            "initialize": "",
            "finalize": "",
            "libs": [],
            "x": 320,
            "y": 180,
            "wires": [
                [
                    "6ac6b44784427ffc",
                    "976fac59a1d4967d",
```

```
            "376f66c48cfa5f5a"
        ],
        [
            "107df563935a0217",
            "376f66c48cfa5f5a",
            "976fac59a1d4967d"
        ]
    ]
},
{
    "id": "6ac6b44784427ffc",
    "type": "ui_gauge",
    "z": "0f1e807e520403ff",
    "tab": "4410af667a3d8692",
    "name": "",
    "group": "Q402",
    "order": 2,
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "x": 790,
    "y": 140,
    "wires": []
},
{
    "id": "976fac59a1d4967d",
    "type": "ui_chart",
    "z": "0f1e807e520403ff",
    "tab": "4410af667a3d8692",
    "name": "",
    "group": "Q402",
    "order": 1,
    "interpolate": "linear",
    "nodata": "",
    "removeOlder": "5",
    "removeOlderUnit": "60",
    "x": 790,
    "y": 80,
    "wires": [
```

```
                []
        ]
    },
    {
        "id": "376f66c48cfa5f5a",
        "type": "ui_chart",
        "z": "0f1e807e520403ff",
        "tab": "4410af667a3d8692",
        "name": "",
        "group": "Q402",
        "order": 0,
        "interpolate": "linear",
        "nodata": "",
        "removeOlder": 1,
        "removeOlderUnit": "86400",
        "x": 790,
        "y": 40,
        "wires": [
            []
        ]
    },
    {
        "id": "107df563935a0217",
        "type": "ui_gauge",
        "z": "0f1e807e520403ff",
        "tab": "4410af667a3d8692",
        "name": "",
        "group": "Q402",
        "order": 4,
        "format": "{{value}}",
        "min": 0,
        "max": "100",
        "x": 790,
        "y": 180,
        "wires": []
    },
    {
        "id": "e759f324160a52de",
        "type": "http request",
```

```
        "z": "0f1e807e520403ff",
        "name": "",
        "method": "POST",
        "ret": "txt",
        "paytoqs": "ignore",
        "url": "https://notify-api.line.me/api/notify",
        "tls": "",
        "persist": false,
        "proxy": "",
        "insecureHTTPParser": false,
        "authType": "",
        "senderr": false,
        "headers": [],
        "x": 630,
        "y": 280,
        "wires": [
            [
                "55c7da970b5006a2"
            ]
        ]
    },
    {
        "id": "bf7aa3046b21a120",
        "type": "mqtt in",
        "z": "0f1e807e520403ff",
        "name": "",
        "topic": "feedback/sensors",
        "qos": "0",
        "datatype": "auto-detect",
        "broker": "d9b6929cc83977a5",
        "nl": false,
        "rap": true,
        "rh": 0,
        "inputs": 0,
        "x": 130,
        "y": 180,
        "wires": [
            [
                "e7793cb2f893f284"
```

```
            ]
        ]
    },
    {
        "id": "52607273423ad64e",
        "type": "mqtt out",
        "z": "0f1e807e520403ff",
        "name": "",
        "topic": "control/leds",
        "qos": "0",
        "retain": "false",
        "respTopic": "",
        "contentType": "",
        "userProps": "",
        "correl": "",
        "expiry": "",
        "broker": "d9b6929cc83977a5",
        "x": 810,
        "y": 440,
        "wires": []
    },
    {
        "id": "6c5bd7c7f79e6205",
        "type": "mqtt in",
        "z": "0f1e807e520403ff",
        "name": "",
        "topic": "feedback/switch",
        "qos": "0",
        "datatype": "auto-detect",
        "broker": "d9b6929cc83977a5",
        "nl": false,
        "rap": true,
        "rh": 0,
        "inputs": 0,
        "x": 120,
        "y": 240,
        "wires": [
            [
                "6e39010d22ea1619"
```

```
            ]
        ]
    },
    {
        "id": "b8a769625e1c9190",
        "type": "mqtt in",
        "z": "0f1e807e520403ff",
        "name": "",
        "topic": "control/leds",
        "qos": "0",
        "datatype": "auto-detect",
        "broker": "d9b6929cc83977a5",
        "nl": false,
        "rap": true,
        "rh": 0,
        "inputs": 0,
        "x": 110,
        "y": 320,
        "wires": [
            [
                "c5633dedfbbd146a"
            ]
        ]
    },
    {
        "id": "c5633dedfbbd146a",
        "type": "line-notify",
        "z": "0f1e807e520403ff",
        "name": "",
        "message": "LED ActiveYYYYYYYYYYYYYYY",
        "contentType": "message",
        "imageThumbnail": "",
        "imageUrl": "",
        "sticker": "default",
        "stickerPackageId": "11",
        "stickerId": "1",
        "silent": false,
        "creds": "591415dad7908dc3",
        "x": 460,
```

```
        "y": 320,
        "wires": [
            [
                "e759f324160a52de"
            ]
        ]
    },
    {
        "id": "6e39010d22ea1619",
        "type": "function",
        "z": "0f1e807e520403ff",
        "name": "",
        "func": "\nvar stsNow =context.global.sts;\nif (msg.payload == 'SW_ON') {\n stsNow =
true;\n}\nif (msg.payload == 'SW_OFF'){\n stsNow = false;\n}\ncontext.global.sts =
stsNow;\nmsg.payload = stsNow;\n\nreturn msg;",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 320,
        "y": 240,
        "wires": [
            [
                "26213b1ceab8d687"
            ]
        ]
    },
    {
        "id": "591415dad7908dc3",
        "type": "linetoken",
        "name": "M4Q402"
    },
    {
        "id": "4410af667a3d8692",
        "type": "ui_tab",
        "name": "Q402-0401",
        "icon": "dashboard",
        "order": "4"
```

```
        },
        {
            "id": "d9b6929cc83977a5",
            "type": "mqtt-broker",
            "name": "",
            "broker": "192.168.0.31",
            "port": "1883",
            "clientid": "",
            "autoConnect": true,
            "usetls": false,
            "protocolVersion": "4",
            "keepalive": "60",
            "cleansession": true,
            "birthTopic": "",
            "birthQos": "0",
            "birthRetain": "false",
            "birthPayload": "",
            "birthMsg": {},
            "closeTopic": "",
            "closeQos": "0",
            "closeRetain": "false",
            "closePayload": "",
            "closeMsg": {},
            "willTopic": "",
            "willQos": "0",
            "willRetain": "false",
            "willPayload": "",
            "willMsg": {},
            "userProps": "",
            "sessionExpiry": ""
        }
]
```

รูปการทดสอบ 1

🔔 Quiz_402 – LINE Noti: Hello

18.07 น.

รูปการทดสอบ 2:

🔔 Quiz_402 – LINE Noti: LINE-Massage From Node-RED

Quiz_402 – LINE Noti: LINE-Sticker

รูปการทดสอบ 3

🔔 Quiz_402 – LINE Noti: Send Image Data
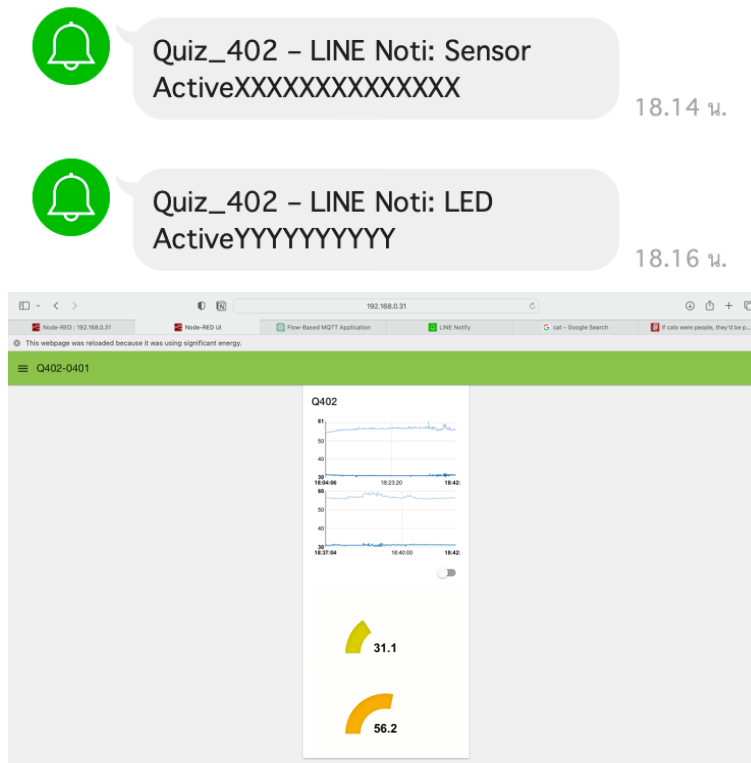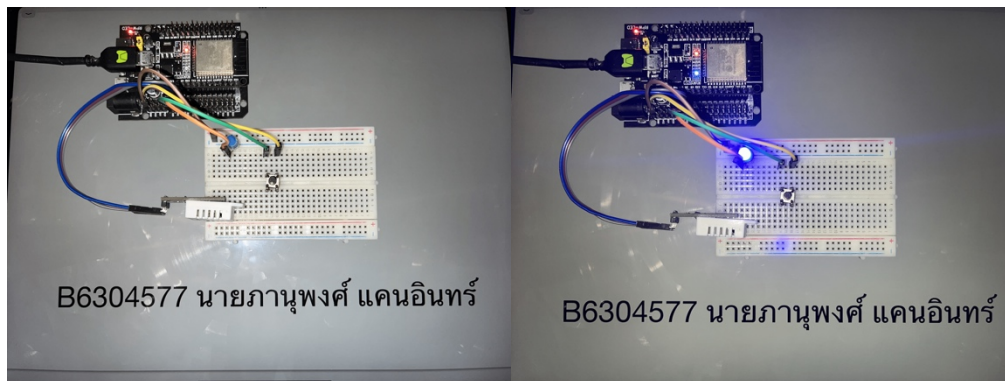
18.11 น.

🔔

18.11 ↓

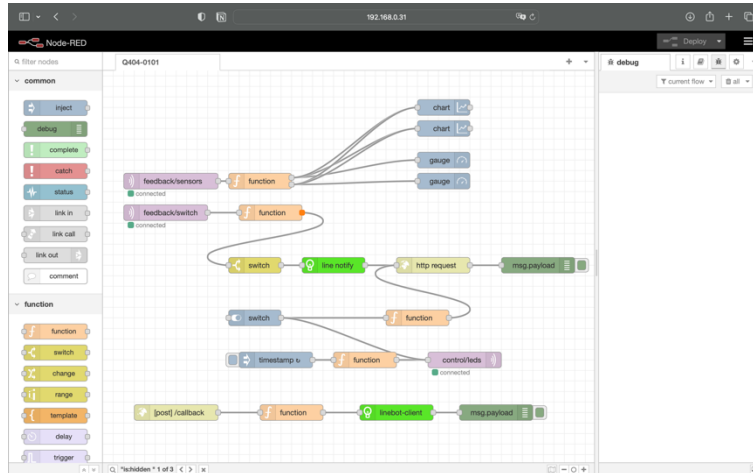บันทึก | บันทึกเป็น... | แชร์ | Keep

รูปการทดสอบ 4

รูปการทดสอบ 5

Quiz_403 – LINE Chatbot

- **แสดงรูป โปรแกรม ของผลการทำงานตามหัวข้อ การโต้ตอบด้วยข้อความสำหรับฟาร์มอัจฉริยะ**

Capture Node-RED Flow

Node-RED Code

```
[
    {
        "id": "ca81a9d4127e494b",
        "type": "tab",
        "label": "Q404-0101",
        "disabled": false,
        "info": ""
    },
    {
        "id": "e7622ccb4bf17cea",
        "type": "function",
        "z": "ca81a9d4127e494b",
        "name": "",
        "func": "//   Tempp=38.20,Humid=73.52 \n\nvar msg1 = {};\nvar msg2 = {};\n\nvar
output =   msg.payload.split(\",\");\n\nvar sTempp =   output[0].split(\"=\");\nmsg1.payload
= sTempp[1];\nmsg1.topic =   'Temperature';\n\nvar sHumid =
output[1].split(\"=\");\nmsg2.payload = sHumid[1];\nmsg2.topic =
'Humidity';\n\ncontext.global.tempp = sTempp[1];\ncontext.global.humid =
sHumid[1];\n\nreturn   [msg1,msg2];",
        "outputs": 2,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
```

```
    "libs": [],
    "x": 300,
    "y": 320,
    "wires": [
        [
            "8d02bda2ce6a4e83",
            "056ca60eaa1e5871",
            "ae97a9fe27d15673"
        ],
        [
            "04bf88bdb94b1023",
            "ae97a9fe27d15673",
            "056ca60eaa1e5871"
        ]
    ]
},
{
    "id": "8d02bda2ce6a4e83",
    "type": "ui_gauge",
    "z": "ca81a9d4127e494b",
    "tab": "1a56f108c6e37004",
    "name": "",
    "group": "Q403-0101",
    "order": 2,
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "x": 650,
    "y": 280,
    "wires": []
},
{
    "id": "056ca60eaa1e5871",
    "type": "ui_chart",
    "z": "ca81a9d4127e494b",
    "tab": "1a56f108c6e37004",
    "name": "",
    "group": "Q403-0101",
    "order": 1,
```

```
        "interpolate": "linear",
        "nodata": "",
        "removeOlder": "5",
        "removeOlderUnit": "60",
        "x": 650,
        "y": 220,
        "wires": [
            []
        ]
    },
    {
        "id": "ae97a9fe27d15673",
        "type": "ui_chart",
        "z": "ca81a9d4127e494b",
        "tab": "1a56f108c6e37004",
        "name": "",
        "group": "Q403-0101",
        "order": 0,
        "interpolate": "linear",
        "nodata": "",
        "removeOlder": 1,
        "removeOlderUnit": "86400",
        "x": 650,
        "y": 180,
        "wires": [
            []
        ]
    },
    {
        "id": "04bf88bdb94b1023",
        "type": "ui_gauge",
        "z": "ca81a9d4127e494b",
        "tab": "1a56f108c6e37004",
        "name": "",
        "group": "Q403-0101",
        "order": 4,
        "format": "{{value}}",
        "min": 0,
        "max": "100",
```

```
            "x": 650,
            "y": 320,
            "wires": []
        },
        {
            "id": "bb483091a924395e",
            "type": "ui_switch",
            "z": "ca81a9d4127e494b",
            "tab": "1a56f108c6e37004",
            "name": "",
            "topic": "topic",
            "group": "Q403-0101",
            "order": 2,
            "onvalue": "ON1",
            "offvalue": "OFF1",
            "x": 290,
            "y": 580,
            "wires": [
                [
                    "7feab23724ead6ef",
                    "a0ce9594ad51bc5d"
                ]
            ]
        },
        {
            "id": "9197b1d96b2bdcfe",
            "type": "function",
            "z": "ca81a9d4127e494b",
            "name": "",
            "func": "\nvar    stsNow = context.global.sts;\nif (msg.payload == 'SW_ON') {\n    stsNow = true;\n}\nif (msg.payload ==   'SW_OFF') {\n    stsNow = false;\n}\ncontext.global.sts = stsNow;\nmsg.payload = stsNow;\n\nreturn    msg;",
            "outputs": 1,
            "noerr": 0,
            "initialize": "",
            "finalize": "",
            "libs": [],
            "x": 320,
            "y": 380,
```

```
        "wires": [
            [
                "8d924afac0269860"
            ]
        ]
    },
    {
        "id": "8428c3554b4735a6",
        "type": "line-notify",
        "z": "ca81a9d4127e494b",
        "name": "",
        "message": "Sensor   Active",
        "contentType": "message",
        "imageThumbnail": "",
        "imageUrl": "",
        "sticker": "default",
        "stickerPackageId": "11",
        "stickerId": "1",
        "silent": false,
        "creds": "591415dad7908dc3",
        "x": 440,
        "y": 480,
        "wires": [
            [
                "652ee4d0e5aac573"
            ]
        ]
    },
    {
        "id": "9afd7fab7f9cb4f5",
        "type": "debug",
        "z": "ca81a9d4127e494b",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "targetType": "msg",
```

```
            "statusVal": "",
            "statusType": "auto",
            "x": 830,
            "y": 480,
            "wires": []
        },
        {
            "id": "8d924afac0269860",
            "type": "switch",
            "z": "ca81a9d4127e494b",
            "name": "",
            "property": "payload",
            "propertyType": "msg",
            "rules": [
                {
                    "t": "eq",
                    "v": "1",
                    "vt": "str"
                }
            ],
            "checkall": "true",
            "repair": false,
            "outputs": 1,
            "x": 290,
            "y": 480,
            "wires": [
                [
                    "8428c3554b4735a6"
                ]
            ]
        },
        {
            "id": "a868e94c90eb9647",
            "type": "function",
            "z": "ca81a9d4127e494b",
            "name": "",
            "func": "var receive_msg = msg.payload;\nvar reply
;\n\nif(receive_msg.events[0].message.type==\"text\")\n      {       \n
if(receive_msg.events[0].message.text==\"Hello\")\n        {\n          reply = {type:'text',
text:\"Hello\"};\n        }\n\n      else if(receive_msg.events[0].message.text==\"Tempp\")\n
```

```
{\n         reply = {type:'text', text:    context.global.tempp};\n         }\n\n        else
if(receive_msg.events[0].message.text==\"Humid\")\n          {\n          reply = {type:'text',
text:    context.global.humid};\n         }\n\n        else
if(receive_msg.events[0].message.text==\"Sensor\")\n         {   \n          var xSensor =
'Sensor Off'\n         if(context.global.sts){xSensor = 'Sensor On'}\n         reply =
{type:'text', text:    xSensor};\n         }\n\n        else
if(receive_msg.events[0].message.text==\"On\")\n        {\n          context.global.ctrl =
'ON1'\n         reply = {type:'text', text:\"Ok turn on\"};\n        }\n\n        else
if(receive_msg.events[0].message.text==\"Off\")\n        {   \n          context.global.ctrl =
'OFF1'\n          reply = {type:'text', text:\"Ok turn off\"};\n        }\n\n        else\n      {
\n         reply = {type:'text', text:\"i don't know\"}\n         }\n     }\n\nvar value =
[receive_msg,reply];\nmsg.payload = value;\nreturn   msg;\n",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 360,
        "y": 760,
        "wires": [
            [
                "22a80689dd4116ce"
            ]
        ]
    },
    {
        "id": "f6699a8a4e33ecc6",
        "type": "debug",
        "z": "ca81a9d4127e494b",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "targetType": "msg",
        "statusVal": "",
        "statusType": "auto",
        "x": 750,
        "y": 760,
        "wires": []
    },
    {
```

```
    "id": "7feab23724ead6ef",
    "type": "function",
    "z": "ca81a9d4127e494b",
    "name": "",
    "func": "context.global.ctrl   = msg.payload\nreturn   msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 600,
    "y": 580,
    "wires": [
        [
            "652ee4d0e5aac573"
        ]
    ]
},
{
    "id": "326938dc1ca536d2",
    "type": "inject",
    "z": "ca81a9d4127e494b",
    "name": "",
    "props": [
        {
            "p": "payload"
        },
        {
            "p": "topic",
            "vt": "str"
        }
    ],
    "repeat": "10",
    "crontab": "",
    "once": false,
    "onceDelay": 0.1,
    "topic": "",
    "payload": "",
    "payloadType": "date",
```

```
        "x": 330,
        "y": 660,
        "wires": [
            [
                "dd26eb7aebe35092"
            ]
        ]
    },
    {
        "id": "dd26eb7aebe35092",
        "type": "function",
        "z": "ca81a9d4127e494b",
        "name": "",
        "func": "msg.payload    = context.global.ctrl \nreturn    msg;",
        "outputs": 1,
        "noerr": 0,
        "initialize": "",
        "finalize": "",
        "libs": [],
        "x": 500,
        "y": 660,
        "wires": [
            [
                "a0ce9594ad51bc5d"
            ]
        ]
    },
    {
        "id": "652ee4d0e5aac573",
        "type": "http request",
        "z": "ca81a9d4127e494b",
        "name": "",
        "method": "POST",
        "ret": "txt",
        "paytoqs": "ignore",
        "url": "https://notify-api.line.me/api/notify",
        "tls": "",
        "persist": false,
        "proxy": "",
```

```
        "insecureHTTPParser": false,
        "authType": "",
        "senderr": false,
        "headers": [],
        "x": 630,
        "y": 480,
        "wires": [
            [
                "9afd7fab7f9cb4f5"
            ]
        ]
    },
    {
        "id": "eca20c5d08fd540d",
        "type": "mqtt in",
        "z": "ca81a9d4127e494b",
        "name": "",
        "topic": "feedback/sensors",
        "qos": "0",
        "datatype": "auto-detect",
        "broker": "d9b6929cc83977a5",
        "nl": false,
        "rap": true,
        "rh": 0,
        "inputs": 0,
        "x": 130,
        "y": 320,
        "wires": [
            [
                "e7622ccb4bf17cea"
            ]
        ]
    },
    {
        "id": "8bcf5d4ba1886bbb",
        "type": "mqtt in",
        "z": "ca81a9d4127e494b",
        "name": "",
        "topic": "feedback/switch",
```

```
        "qos": "0",

        "datatype": "auto-detect",

        "broker": "d9b6929cc83977a5",

        "nl": false,

        "rap": true,

        "rh": 0,

        "inputs": 0,

        "x": 120,

        "y": 380,

        "wires": [

            [

                "9197b1d96b2bdcfe"

            ]

        ]

    },

    {

        "id": "a0ce9594ad51bc5d",

        "type": "mqtt out",

        "z": "ca81a9d4127e494b",

        "name": "",

        "topic": "control/leds",

        "qos": "0",

        "retain": "false",

        "respTopic": "",

        "contentType": "",

        "userProps": "",

        "correl": "",

        "expiry": "",

        "broker": "d9b6929cc83977a5",

        "x": 690,

        "y": 660,

        "wires": []

    },

    {

        "id": "6cec19fea0d1b42a",

        "type": "http in",

        "z": "ca81a9d4127e494b",

        "name": "",

        "url": "/callback",
```

```
        "method": "post",

        "upload": false,

        "swaggerDoc": "",

        "x": 140,

        "y": 760,

        "wires": [

            [

                "a868e94c90eb9647"

            ]

        ]

    },

    {

        "id": "22a80689dd4116ce",

        "type": "linebot-client",

        "z": "ca81a9d4127e494b",

        "name": "",

        "channelAccessToken":
"uHe5G5d9b76TMY+4V/4ps7QcXPx+mC10MunVwNqBknT+0P9/zHCmrFFjeGfkKn+v22UUHSUudDJRcKhiBUo+fmKPx9z
uDb6GmKSXw/J3jVGUsTEQEZpZVZbJJAQNHgdkTVg83niOP0RerH3Tmt7qPwdB04t89/1O/w1cDnyilFU=",

        "channelSecret": "bfa47238da49f53f7afc046db8263a9b",

        "x": 560,

        "y": 760,

        "wires": [

            [

                "f6699a8a4e33ecc6"

            ]

        ]

    },

    {

        "id": "1a56f108c6e37004",

        "type": "ui_tab",

        "name": "Q403-0101",

        "icon": "dashboard",

        "order": "1"

    },

    {

        "id": "591415dad7908dc3",

        "type": "linetoken",

        "name": "M4Q402"
```

```
    },
    {
        "id": "d9b6929cc83977a5",
        "type": "mqtt-broker",
        "name": "",
        "broker": "192.168.0.31",
        "port": "1883",
        "clientid": "",
        "autoConnect": true,
        "usetls": false,
        "protocolVersion": "4",
        "keepalive": "60",
        "cleansession": true,
        "birthTopic": "",
        "birthQos": "0",
        "birthRetain": "false",
        "birthPayload": "",
        "birthMsg": {},
        "closeTopic": "",
        "closeQos": "0",
        "closeRetain": "false",
        "closePayload": "",
        "closeMsg": {},
        "willTopic": "",
        "willQos": "0",
        "willRetain": "false",
        "willPayload": "",
        "willMsg": {},
        "userProps": "",
        "sessionExpiry": ""
    }
]
```

รูปการทดสอบ 1

รูปการทดสอบ 2:

รูปการทดสอบ 3

รูปการทดสอบ 4

รูปการทดสอบ 5

B6304577 นายภานุพงศ์ แคนอินทร์

รูปการทดสอบ 6

B6304577 นายภานุพงศ์ แคนอินทร์