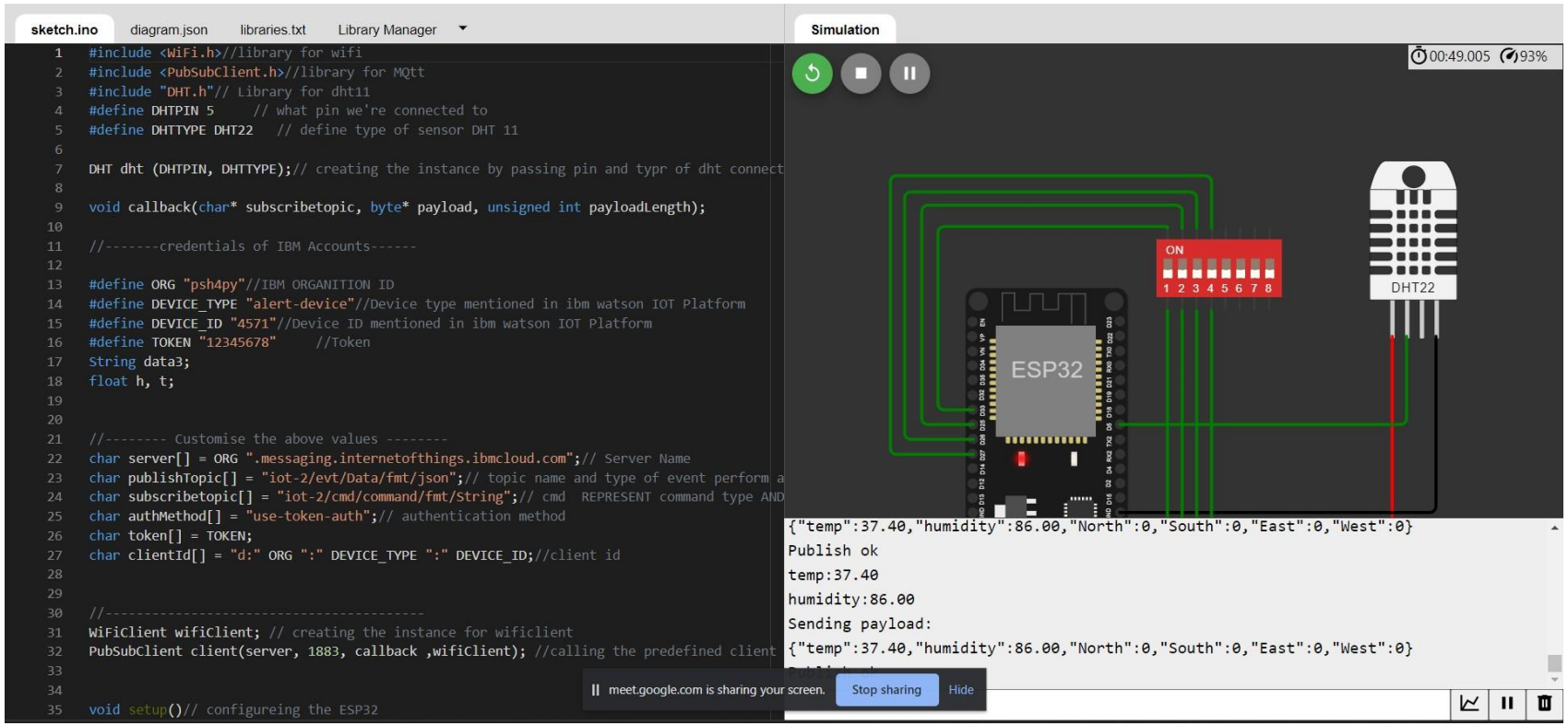
Project Development Phase
Sprint-3: MIT App Design and Testing

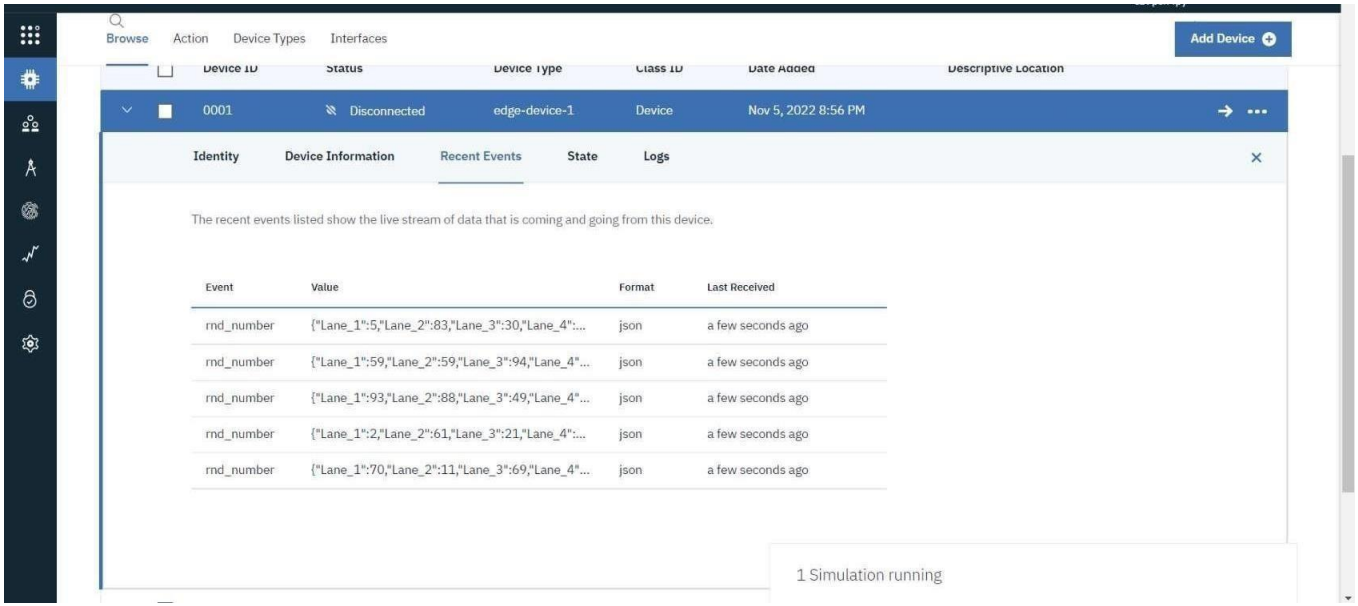
Date	16/11/2022
Team ID	PNT2022TMID14966
Project Name	Signs with Smart Connectivity for BetterRoad Safety
Maximum Marks	8 Marks

Wokwi Simulation:

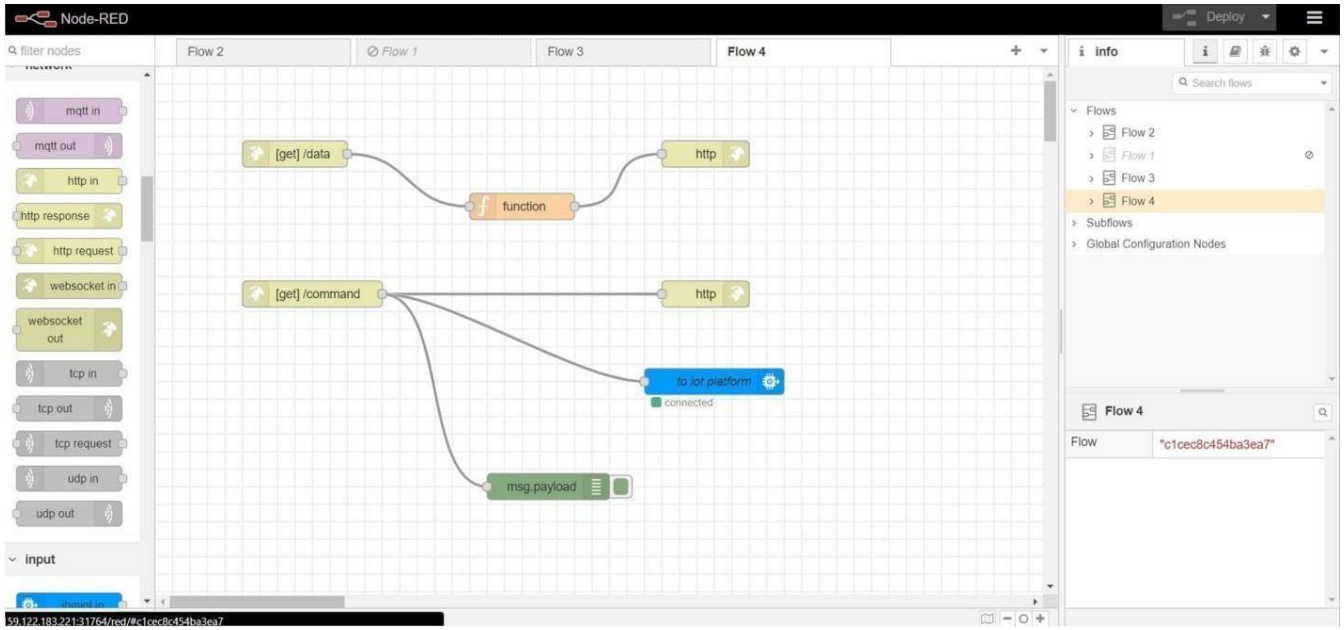



```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 5     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11

DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connect

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "psh4py"//IBM ORGANITION ID
#define DEVICE_TYPE "alert-device"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "4571"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"      //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event perform a
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//-------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client


void setup()// configureing the ESP32
```

IoT Device – IoT Platform



Node Red – Connect with MIT AppInventor



Edit function node

Delete Cancel

O Properties

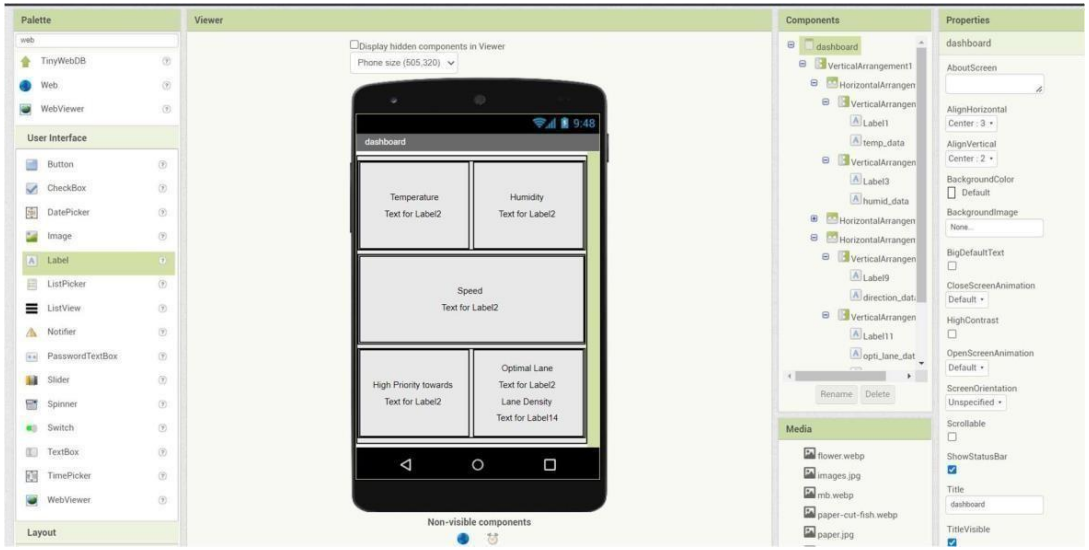
'g• Name Name

O Setup On Start On Message On Stop

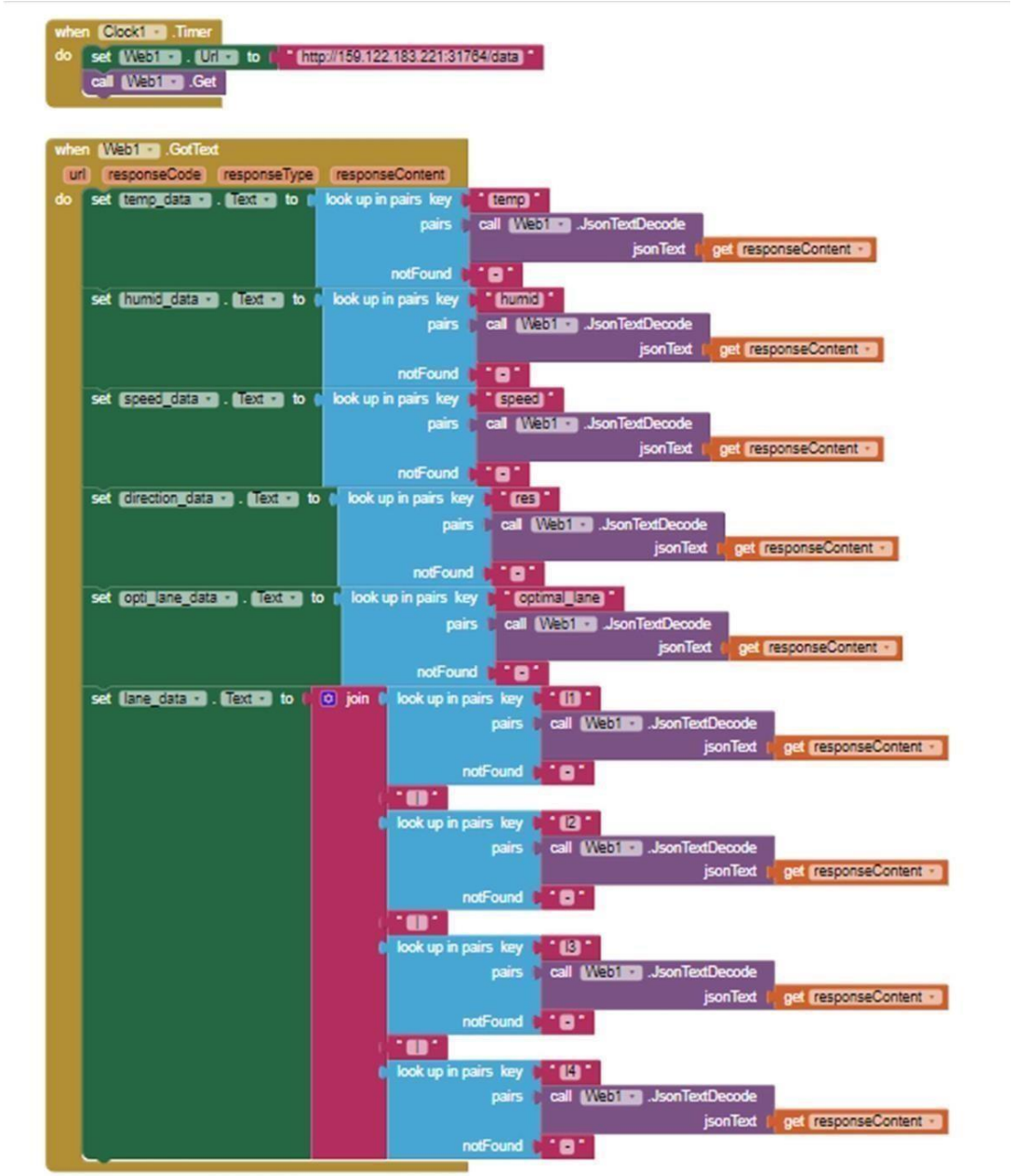
```
• msg. payload = {
2       "temp" :globa1. get ( "temp" ),
3       "humid":global.get(”humid"),
p   "speed”:global.get(”speed"),
s       "n" : global. get (”n”),
6       "s " : g1obal . get (”s"),
7       "e ” : global . get ( "e" ) ,
8       "w" : global . get ( "w") ,
g       ”res” : global. get (” res ” ) ,
16      ”11” : globa1. get ( ”l1 ” ) ,
tt      ”12 ” : global . get( ”12 ” )
12      "13 " :globa1. get( "13 "),
13      "14" : glOba1 . get ( "14" ) ,
t4      "opt imal  l ane" :global . get ( "opt inal  Jane" )
15
16’ };
17
ig  return msg;
```

Output from Node red:

MIT App Inventor UI design:



MIT App Inventor Backend design:



Sprint 3 delivery:

(OUTPUT) Display from MIT App: