**SPRINT-1**

|  |  |
| --- | --- |
| Team ID | PNT2022TMID40719 |
| Project Name | Project – Industry Specific Intelligent Fire  Management system |

***CONFIGURING ESP32 USING WOKWI PROJECTS***

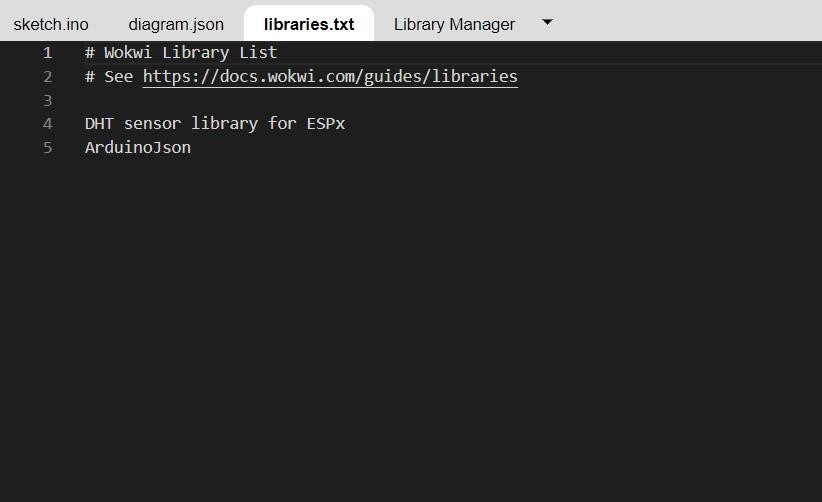
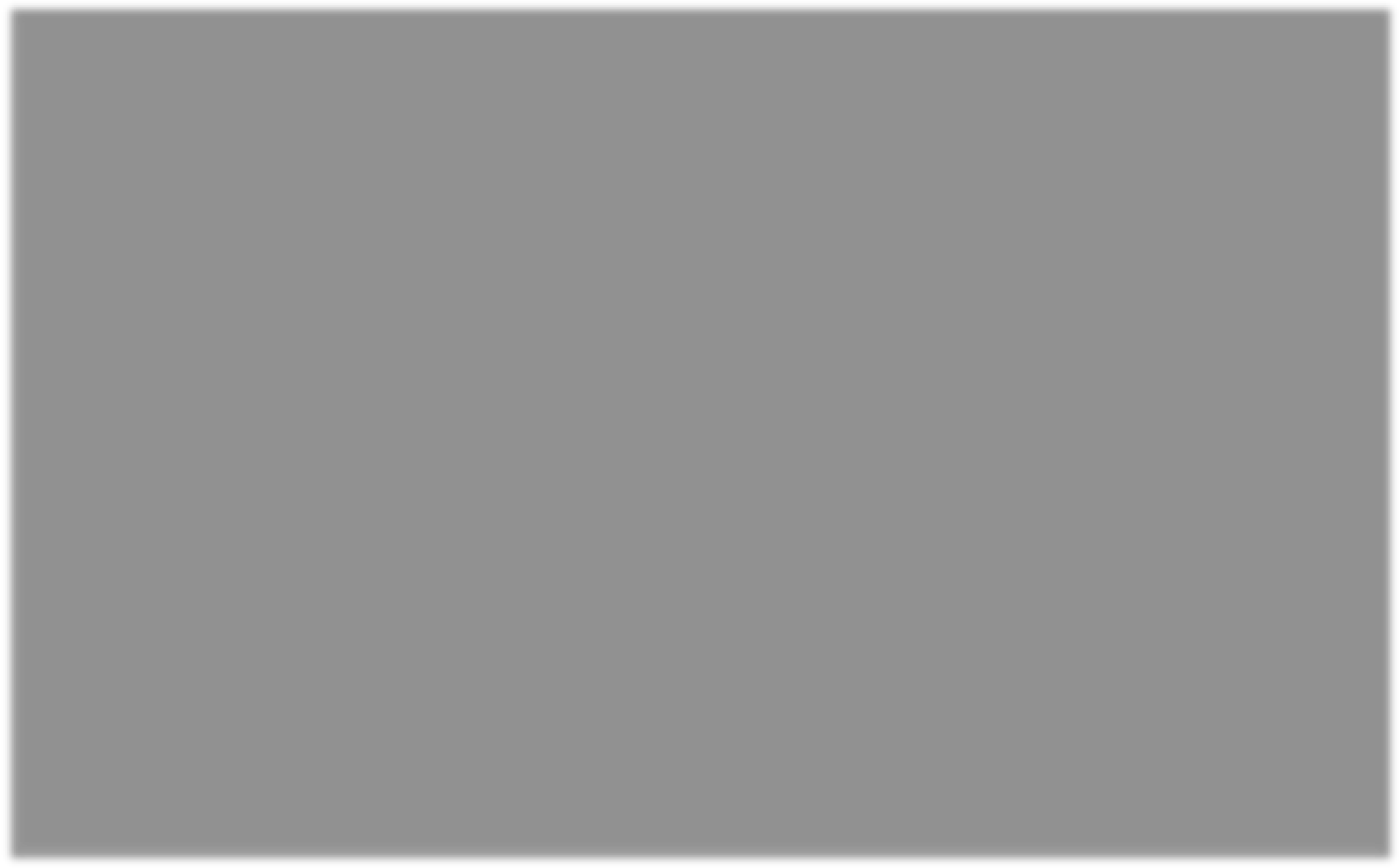
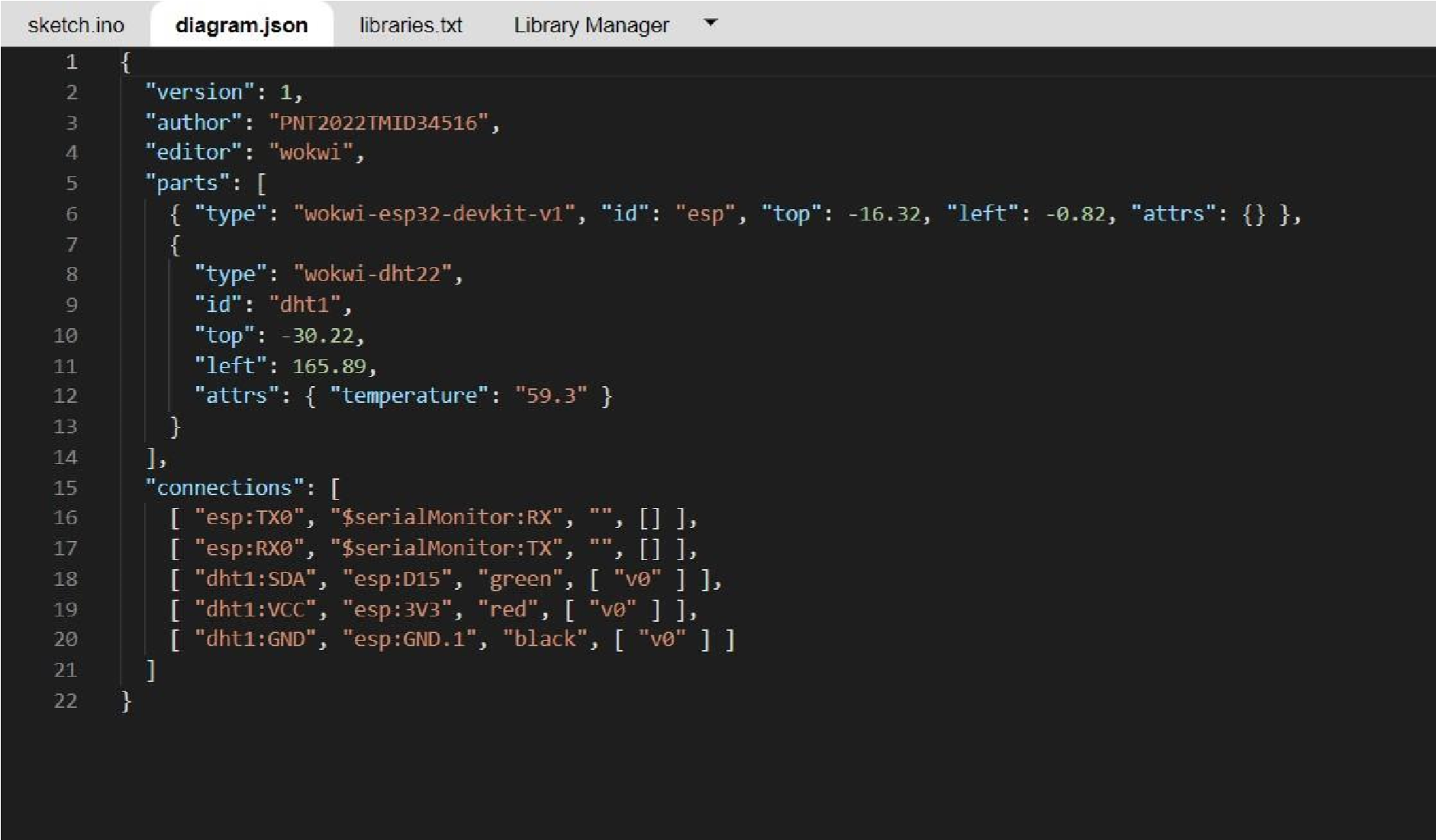
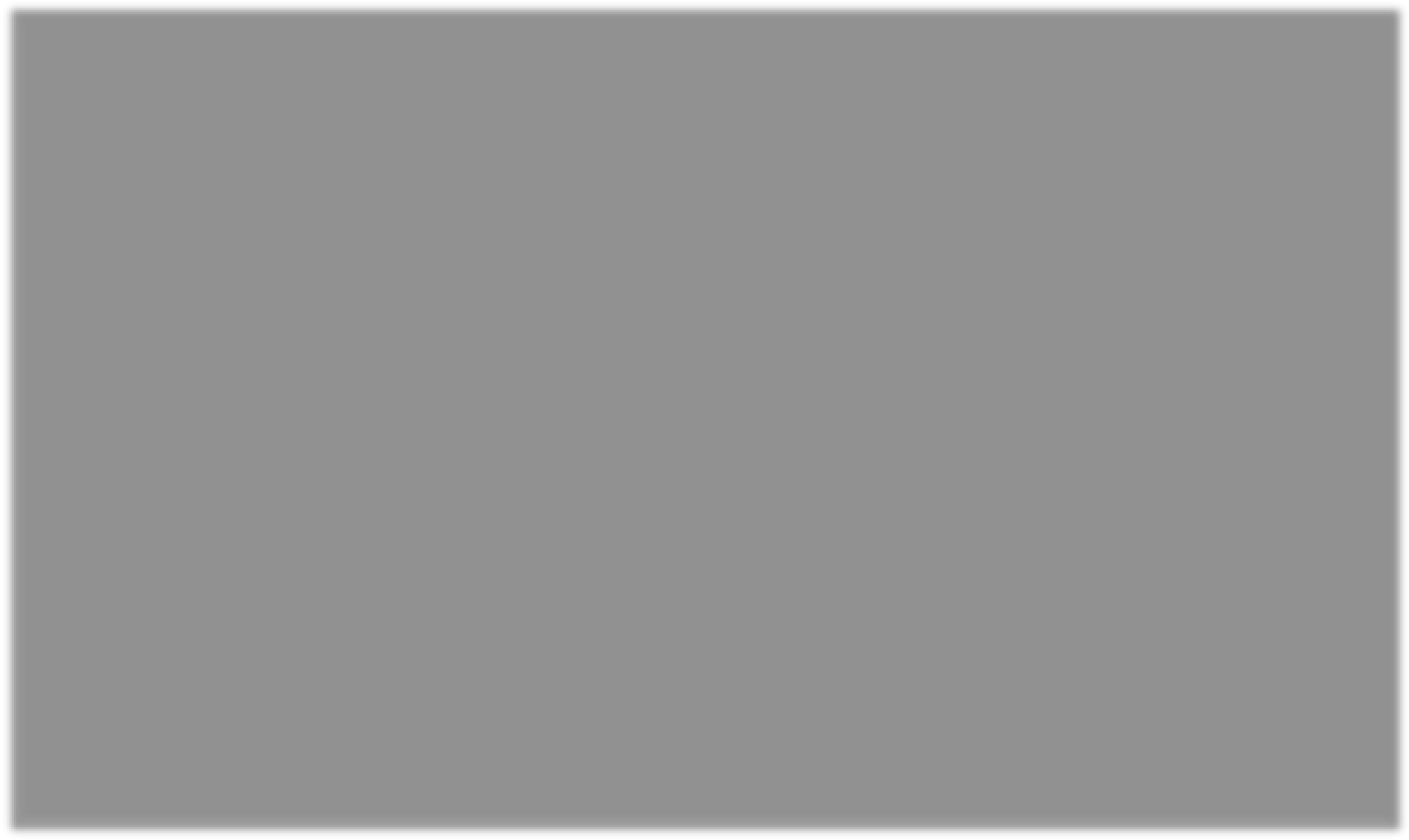
**PROGRAM:**

|  |
| --- |
| #include "DHTesp.h"  #include <cstdlib> #include <time.h> const int DHT\_PIN =  15; bool is\_exhaust\_fan\_on = false; bool is\_sprinkler\_on = false; float temperature =  0; int gas\_ppm = 0; int flame =  0; int flow =  0;    String flame\_status = "";  String accident\_status = "";  String sprinkler\_status = "";    DHTesp dhtSensor;    void setup() {  **Serial**.begin(99900);    /\*\*\*\* sensor pin setups \*\*\*\*/ dhtSensor.setup(DHT\_PIN, DHTesp::DHT22);  //if real gas sensor is used make sure the senor is heated up for acurate readings  /\*  - Here random values for readings and stdout were used to show the working of the devices as physical or simulated devices are not available.  \*/ } void loop() {    TempAndHumidity data = dhtSensor.getTempAndHumidity(); |
| //setting a random seed srand(time(0));    //initial variable activities like declaring , assigning temperature = data.temperature; gas\_ppm = rand()%1000;  int flamereading = rand()%1024; flame = map(flamereading,0,1024,0,1024); int flamerange = map(flamereading,0,1024,0,3); int flow =  ((rand()%100)>50?1:0);    //set a flame status based on how close it is..... switch (flamerange) { case 2: // A fire closer than 1.5 feet away.  flame\_status = "Close Fire"; break; case 1: // A fire between 1-3 feet away.  flame\_status = "Distant Fire"; break; case 0: // No fire detected.  flame\_status = "No Fire"; break;  }    //toggle the fan according to gas in ppm in the room if(gas\_ppm > 100){ is\_exhaust\_fan\_on = true;  } else{ is\_exhaust\_fan\_on = false;  }    //find the accident status 'cause fake alert may be caused by some mischief activities if(temperature < 40 && flamerange ==2){ accident\_status = "need auditing"; is\_sprinkler\_on = false;  } else if(temperature < 40 && flamerange ==0){ accident\_status = "not found"; is\_sprinkler\_on = false;  } else if(temperature > 50 && flamerange == 1){ is\_sprinkler\_on = true; accident\_status =  "moderate";  } else if(temperature > 55 && flamerange == 2){ |
| is\_sprinkler\_on = true; accident\_status = "severe"; }else{ is\_sprinkler\_on  = false; accident\_status  = "none";  }    //send the sprinkler status if(is\_sprinkler\_on){ if(flow){ sprinkler\_status = "working";  } else{ sprinkler\_status = "not working"; } } else if(is\_sprinkler\_on == false){ sprinkler\_status = "it should not!";  } else{ sprinkler\_status  = "Error!!";  }    //Obivously the output.It is like json format 'cause it will help us for future sprints  String out = "{\n\t\"senor\_values\":{"; out+="\n\t\t\"gas\_ppm\":"+String(gas\_ppm)+","; out+="\n\t\t\"temperature\":"+String(temperature,2)+","; out+="\n\t\t\"flame\":"+String(flame)+","; out+="\n\t\t\"flow\":"+String(flow)+",\n\t}"; out+="\n\t\"output\":{";    out+="\n\t\t\"is\_exhaust\_fan\_on\":"+String((is\_exhaust\_fan\_on)?"true":"false ")+",";    out+="\n\t\t\"is\_sprinkler\_on\":"+String((is\_sprinkler\_on)?"true":"false")+"  ,"; out+="\n\t}";  out+="\n\t\"messages\":{"; out+="\n\t\t\"fire\_status\":"+flame\_status+","; out+="\n\t\t\"flow\_status\":"+sprinkler\_status+","; out+="\n\t\t\"accident\_status\":"+accident\_status+","; out+="\n\t}"; out+="\n}";  **Serial**.println(out);    delay(2000);  } |

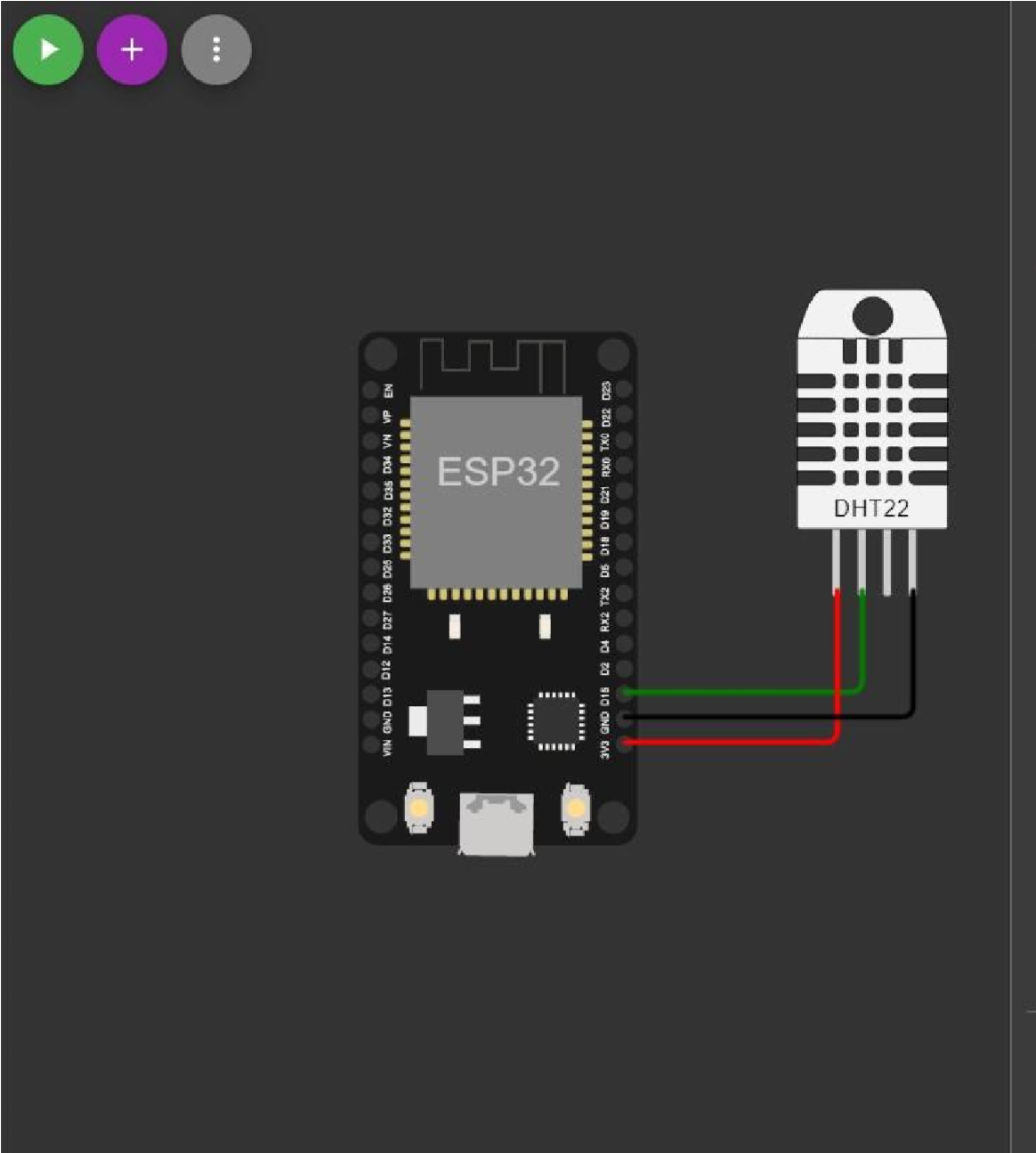
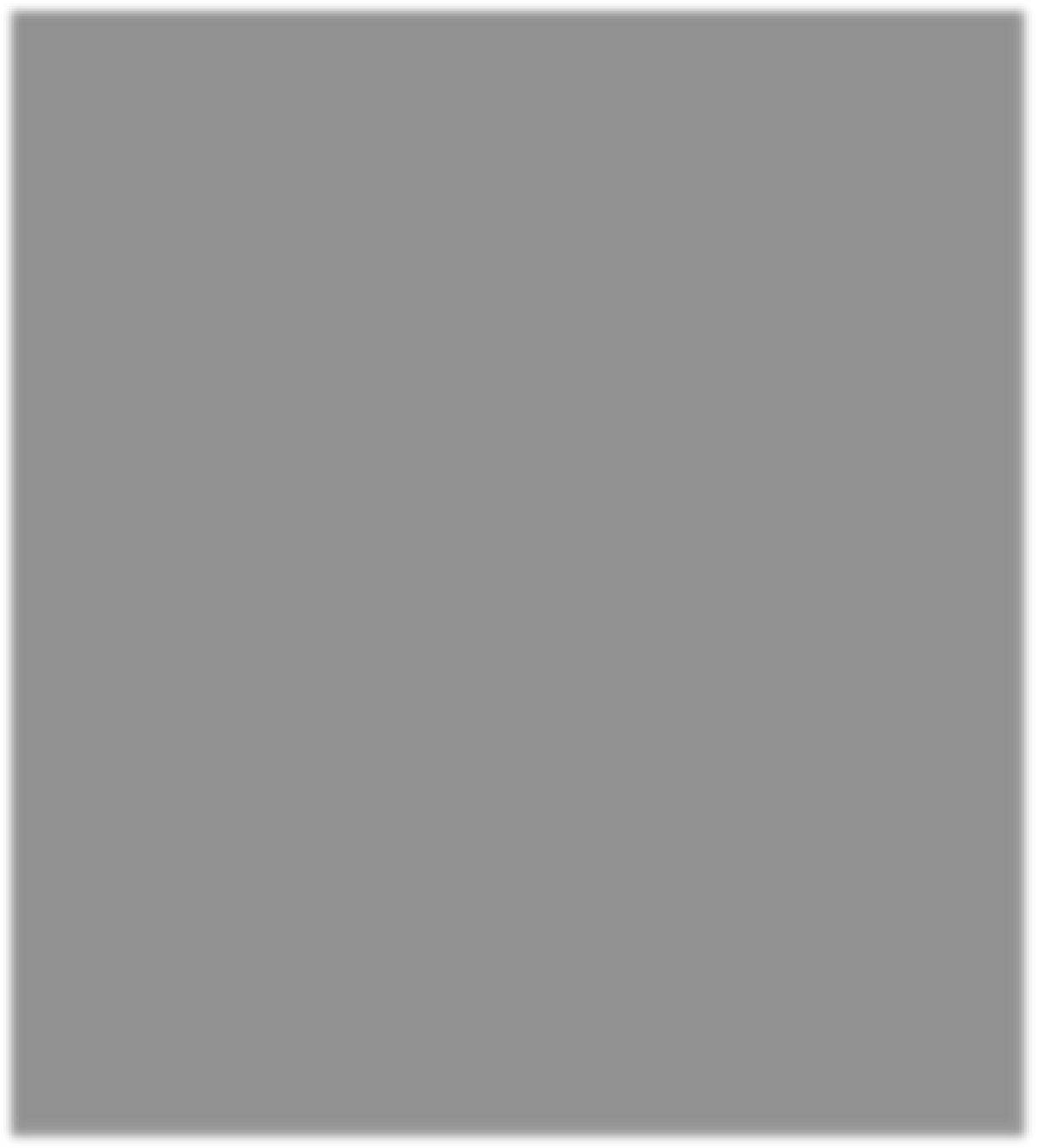
DIAGRAM.JSON



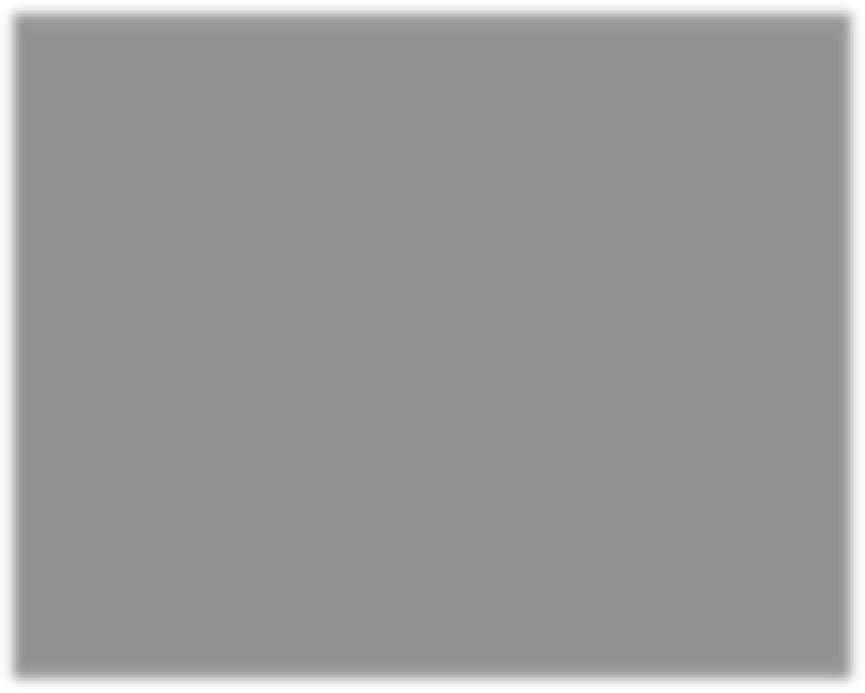
LIBRARIES



CIRCUIT



OUTPUT:



WOKWI LINK [https://wokwi.com/projects/34845600743371630](https://wokwi.com/projects/348456007433716308)

[8](https://wokwi.com/projects/348456007433716308)