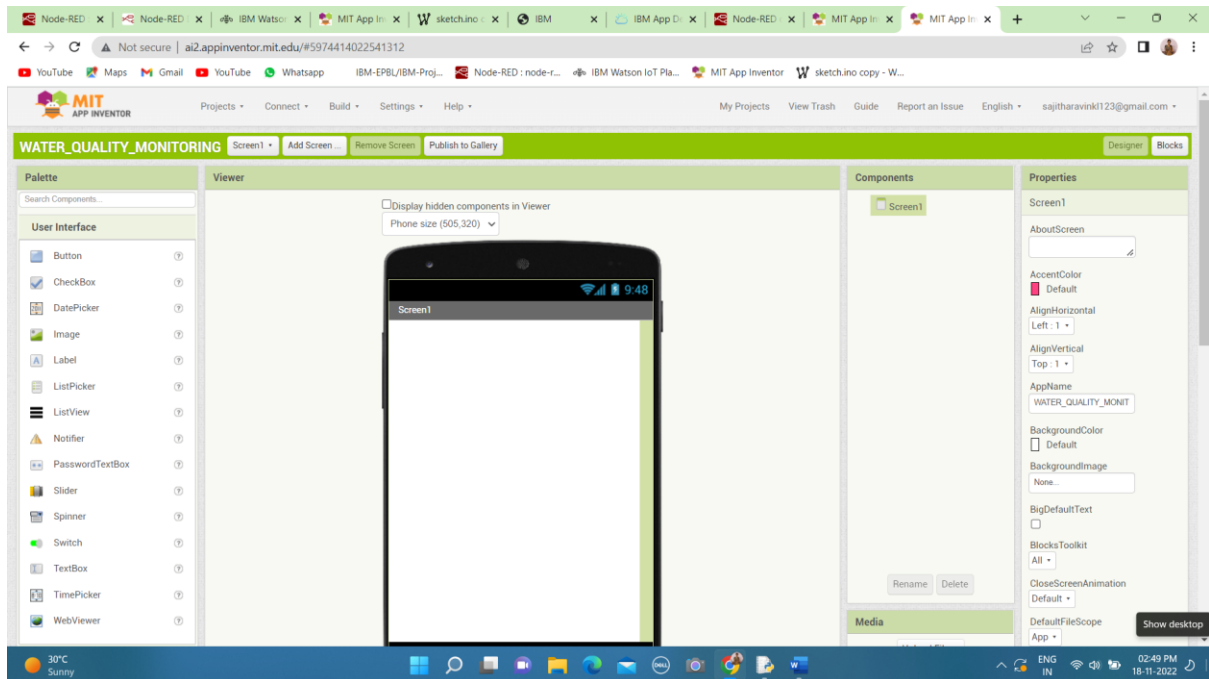
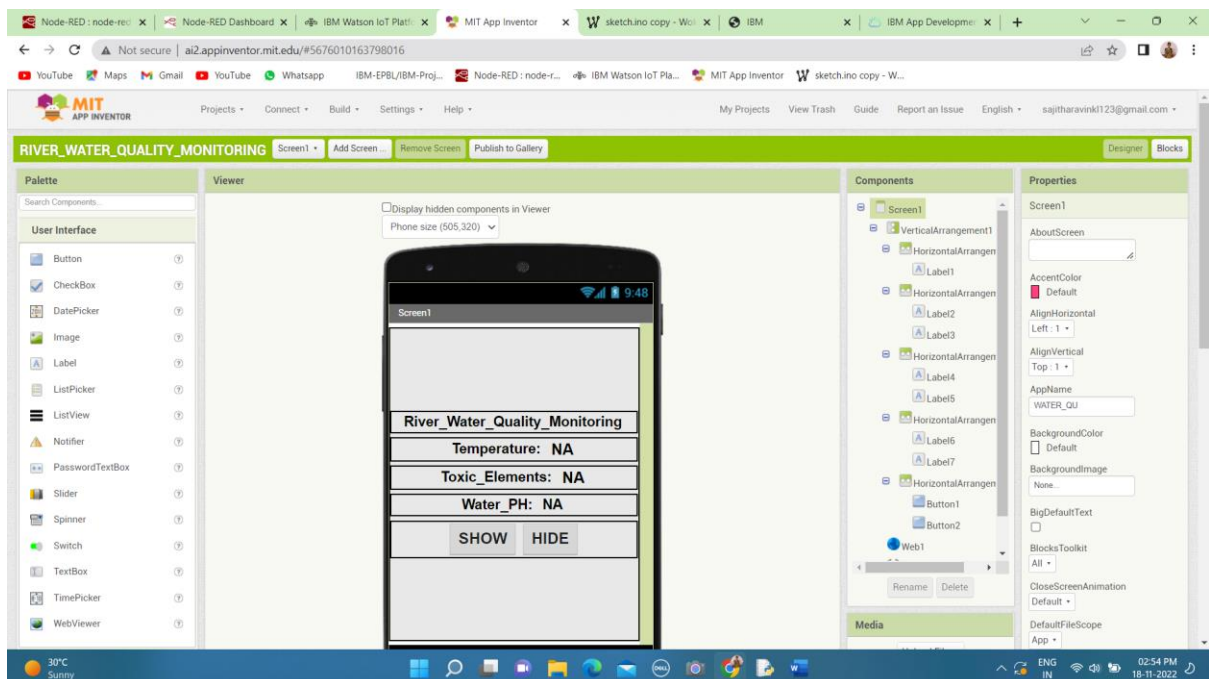


SPRINT-4

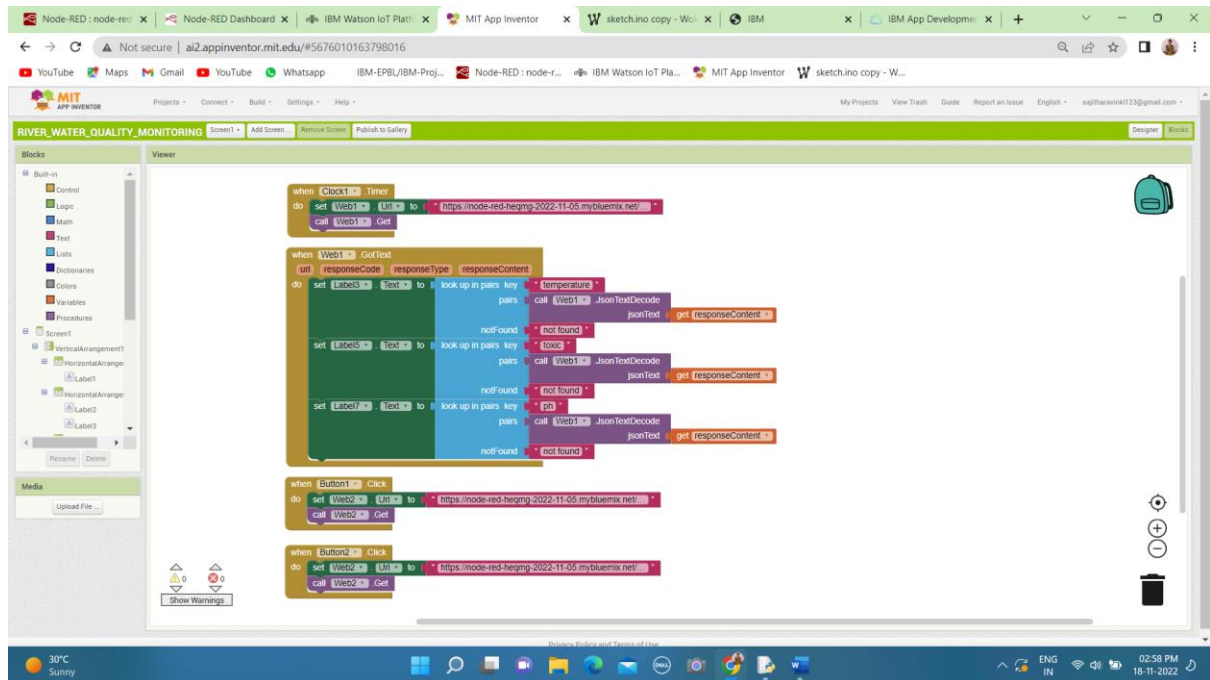
MIT APP LOGIN:



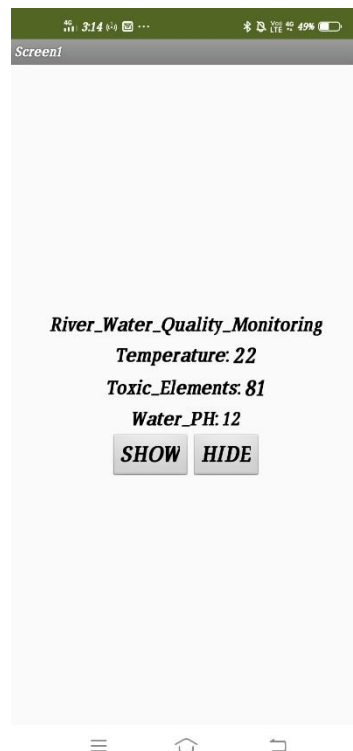
MIT DESIGN BLOCK:



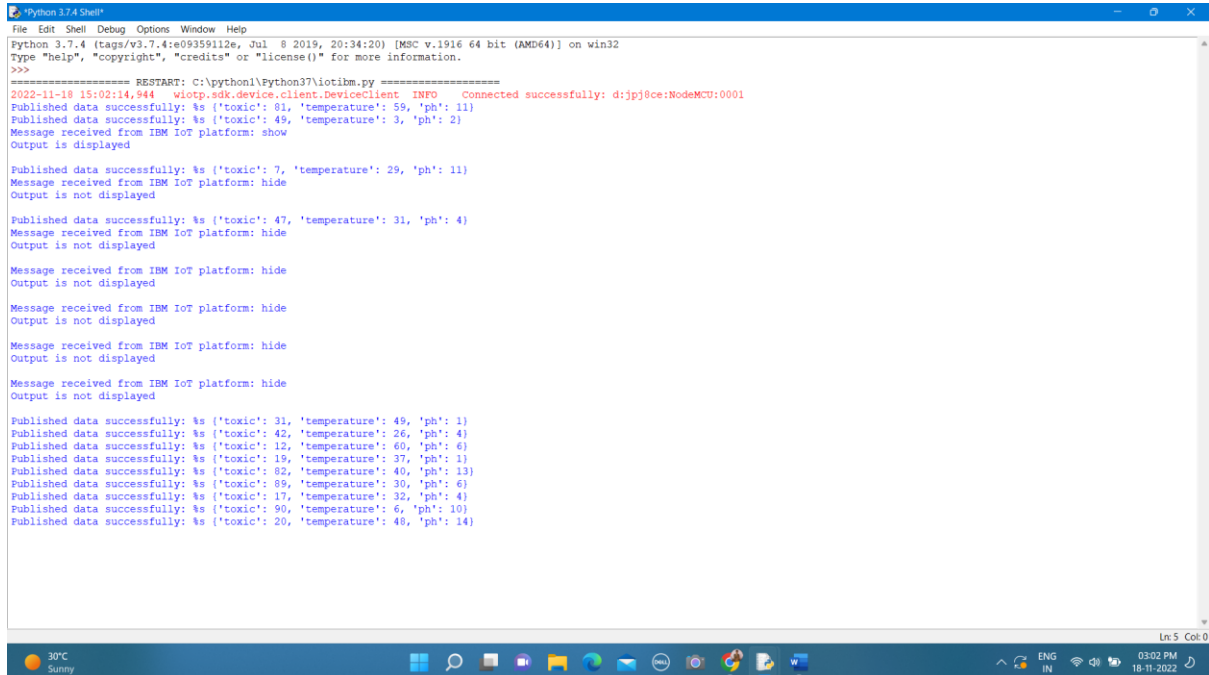
MIT BLOCK CODING:



MOBILE APPLICATION OUTPUT:



PYTHON OUTPUT WHILE CONNECTING WITH MIT INVENTOR , WATSON AND NODE-RED:



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\python1\Python37\iotibm.py =====
2022-11-19 15:02:14.944 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:jjp8ce:NodeMCU:0001
Published data successfully: ts {'toxic': 51, 'temperature': 55, 'ph': 11}
Published data successfully: ts {'toxic': 49, 'temperature': 3, 'ph': 2}
Message received from IBM IoT platform: show
Output is displayed

Published data successfully: ts {'toxic': 7, 'temperature': 29, 'ph': 11}
Message received from IBM IoT platform: hide
Output is not displayed

Published data successfully: ts {'toxic': 47, 'temperature': 31, 'ph': 4}
Message received from IBM IoT platform: hide
Output is not displayed

Message received from IBM IoT platform: hide
Output is not displayed

Message received from IBM IoT platform: hide
Output is not displayed

Message received from IBM IoT platform: hide
Output is not displayed

Message received from IBM IoT platform: hide
Output is not displayed

Published data successfully: ts {'toxic': 31, 'temperature': 49, 'ph': 1}
Published data successfully: ts {'toxic': 42, 'temperature': 26, 'ph': 4}
Published data successfully: ts {'toxic': 12, 'temperature': 60, 'ph': 6}
Published data successfully: ts {'toxic': 19, 'temperature': 37, 'ph': 11}
Published data successfully: ts {'toxic': 82, 'temperature': 40, 'ph': 13}
Published data successfully: ts {'toxic': 89, 'temperature': 30, 'ph': 6}
Published data successfully: ts {'toxic': 17, 'temperature': 32, 'ph': 4}
Published data successfully: ts {'toxic': 90, 'temperature': 6, 'ph': 10}
Published data successfully: ts {'toxic': 20, 'temperature': 48, 'ph': 14}
```

WOKWI CODE:

```
#include <WiFi.h>
```

```
#include <WiFiClient.h>
```

```
#include <PubSubClient.h>
```

```
#include "DHT.h"
```

```
const char* ssid = "Wokwi-GUEST";
```

```
const char* password = "";
```

```
#define DHTPIN 4
```

```
#define DHTTYPE DHT22
```

```
#define ORG_ID "jjp8ce"
```

```
#define DEVICE_TYPE "ESP32"
```

```
#define DEVICE_ID "1234"
```

```

#define TOKEN "0&a7Gug4flWi3(VFM_"

char server[] = ORG_ID ".messaging.internetofthings.ibmcloud.com";
char pubTopic1[] = "iot-2/evt/status1/fmt/json";
char pubTopic2[] = "iot-2/evt/status2/fmt/json";
char authMethod[] = "use-token-auth";
char* apiKey = "a-jpj8ce-yz5qyctnjs";
char* token = TOKEN;

// char clientId[] = "d:" ORG_ID ":" DEVICE_TYPE ":" DEVICE_ID;
char* clientId = "a:" ORG_ID ":" DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server, 1883, NULL, wifiClient);
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.begin(ssid, password, 6);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");

  Serial.print("WiFi connected, IP address: ");
  Serial.println(WiFi.localIP());

```

```

if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!client.connect(clientId, apiKey, token)) {
        Serial.print(".");
        delay(500);
    }
    Serial.println("Bluemix connected");
}
}

long lastMsg = 0;
void loop() {
    client.loop();
    long now = millis();
    if (now - lastMsg > 3000) {
        lastMsg = now;
        float humidity = dht.readHumidity();
        float temperature = dht.readTemperature();
        String payload = "{\"d\":{\"Name\":\"\" DEVICE_ID \"\"\"";
        payload += "\",\"temperature\":\":";
        payload += temperature;
        payload += "\"} }";

        Serial.print("Sending payload: ");
        Serial.println(payload);

        if (client.publish(pubTopic1, (char*) payload.c_str())) {
            Serial.println("Publish ok");
        } else {
            Serial.println("Publish failed");
        }
    }
}

```

```

String payload1 = "{\"d\":{\"Name\":\"\" DEVICE_ID \"\"";

payload1 += "\",\"humidity\"";

payload1 += humidity;

payload1 += "}"}";

if (client.publish(pubTopic2, (char*) payload1.c_str())) {

    Serial.println("Publish ok");

} else {

    Serial.println("Publish failed");

}

}

}

```

WOKWI OUTPUT:

