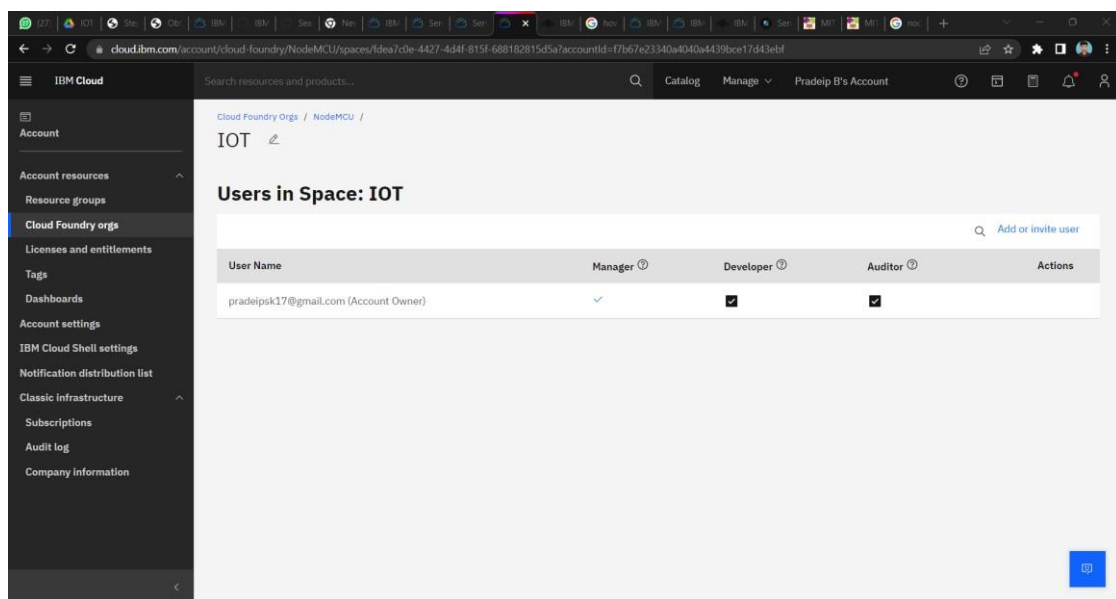


## SPRINT 1

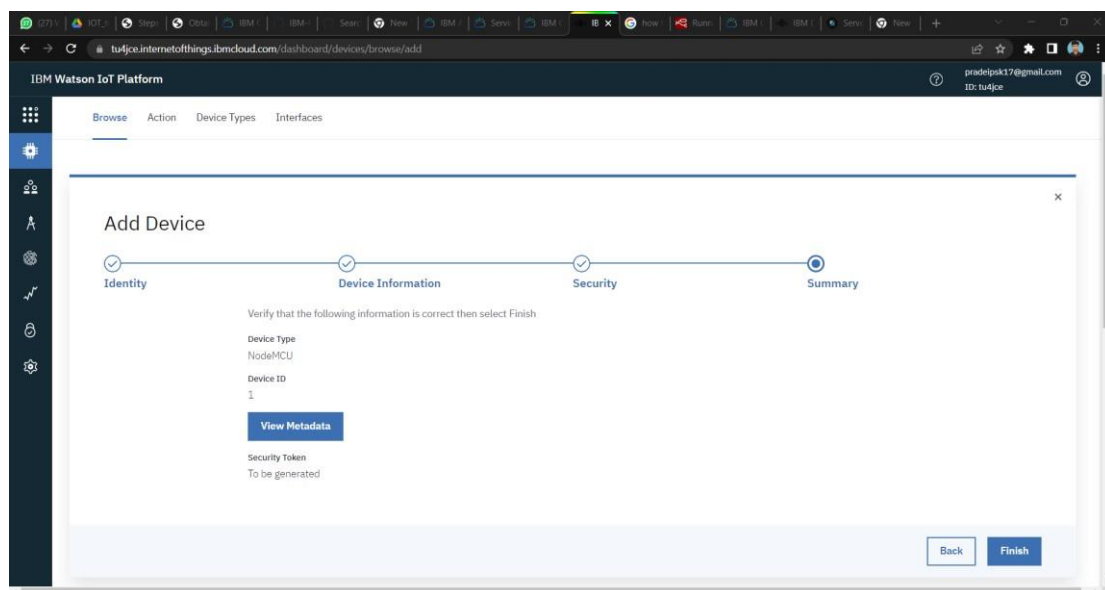
Date	29 October 2022
Team ID	PNT2022TMID26330
Project Name	Smart Farmer – IoT Enabled Farming Application
Maximum Marks	8 Marks

An account has been created on the respective platforms like IBM Cloud, IBM Watson, Node-Red, MIT App Inventor.

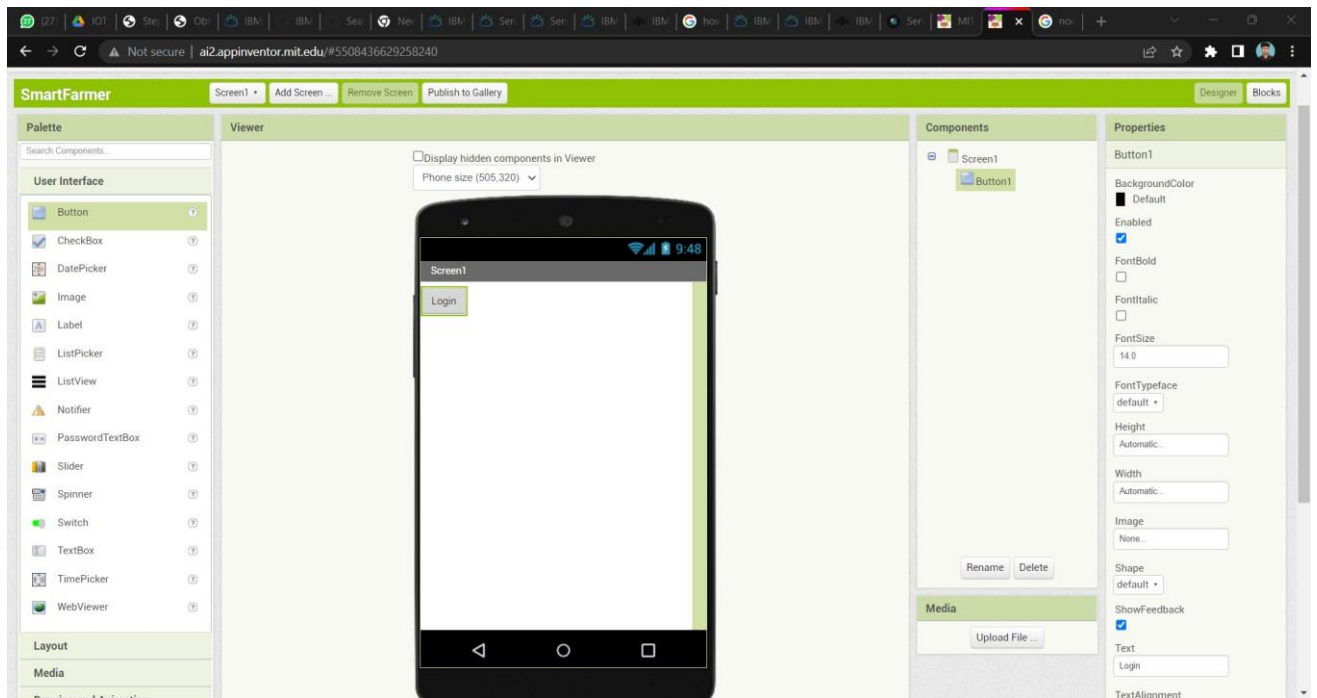
### IBM CLOUD DASHBOARD:



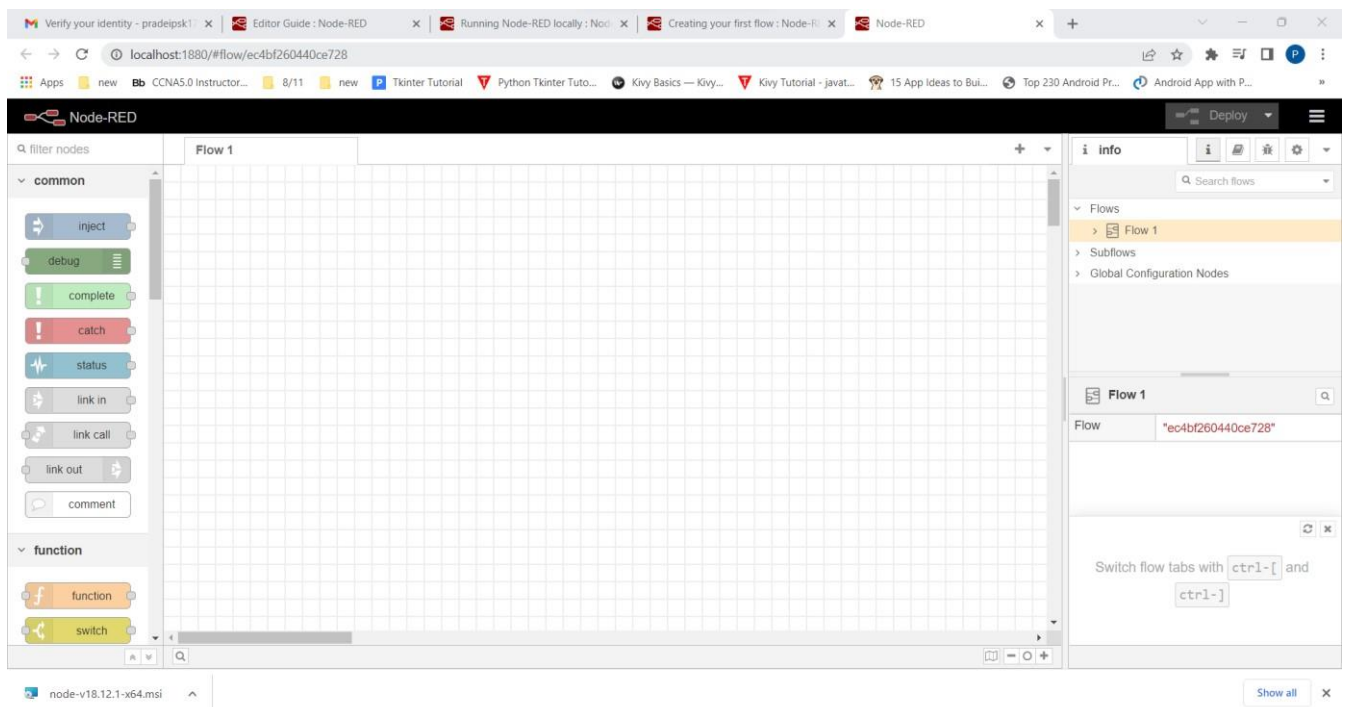
### IBM WATSON IOT PLATFORM:



### MIT APP INVENTOR:



## NODE-RED:



## MEASURING TEMPERATURE AND HUMIDITY VALUES WITH ESP 32:

WOKWI

sketch.ino diagram.json Library Manager

```

1 #include <Adafruit_Sensor.h>
2 #include <DHT.h>
3 #include <DHT_U.h>
4 #define DHTPIN 4
5 #define DHTTYPE DHT22
6 DHT_Unified dht(DHTPIN, DHTTYPE);
7 uint32_t delayMS;
8 void setup() {
9   Serial.begin(9600);
10  // Initialize device.
11  dht.begin();
12  Serial.println(F("DHTxx Unified Sensor Example"));
13  sensor_t sensor;
14  dht.temperature().getSensor(&sensor);
15  Serial.println(F("-----"));
16  Serial.println(F("Temperature Sensor"));
17  Serial.print(sensor.resolution);
18  Serial.println(F("°C"));
19  Serial.println(F("-----"));
20  dht.humidity().getSensor(&sensor);
21  Serial.println(F("Humidity Sensor"));
22  Serial.print(sensor.resolution);
23  Serial.println(F("%"));
24  Serial.println(F("-----"));
25  delayMS = sensor.min_delay / 1000;
26 }
27 void loop() {
28   delay(delayMS);
29   sensors_event_t event;
30   dht.temperature().getEvent(&event);
31   if (isnan(event.temperature)) {
32     Serial.println(F("Error reading temperature!"));
33   }
34   else {
35     Serial.print(F("Temperature: "));

```

Simulation

ESP32

DHT22

0.10°C

-----

Humidity Sensor

0.10%

-----

Temperature: 24.00°C

Humidity: 40.00%

## Program:

```

#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>
#define DHTPIN 4
#define DHTTYPE DHT22
DHT_Unified dht(DHTPIN, DHTTYPE);
uint32_t delayMS; void setup() {
  Serial.begin(9600); // Initialize
  device.
  dht.begin();
  Serial.println(F("DHTxx Unified Sensor Example")); sensor_t
  sensor;
  dht.temperature().getSensor(&sensor);
  Serial.println(F("-----"));
  Serial.println(F("Temperature Sensor"));
  Serial.print(sensor.resolution);
  Serial.println(F("°C"));
  Serial.println(F("-----"));
  dht.humidity().getSensor(&sensor);
  Serial.println(F("Humidity Sensor"));
  Serial.print(sensor.resolution);
  Serial.println(F("%"));
  Serial.println(F("-----"));
  delayMS = sensor.min_delay / 1000;
} void loop() { delay(delayMS);
  sensors_event_t event;
  dht.temperature().getEvent(&event);
  if (isnan(event.temperature)) {

```

```
Serial.println(F("Error reading temperature!"));
} else
{
Serial.print(F("Temperature: "));
Serial.print(event.temperature);
Serial.println(F("°C"));
} dht.humidity().getEvent(&event); if
(isnan(event.relative_humidity)) {
Serial.println(F("Error reading humidity!"));
} else
{
Serial.print(F("Humidity: "));
Serial.print(event.relative_humidity);
Serial.println(F("%"));
}
}
```