

Project Development Phase

Sprint-4

Date	24 November 2022
Team ID	PNT2022TMID08255
Project Name	Signs with Smart Connectivity for Better Road Safety

Sprint Target:

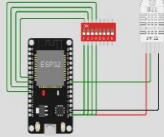
Sprint	Functional Requirements	User Story Number	Task/User Story
Sprint-4	MIT app inventor, Node red	USR-4	MIT app and Design and results

Python code in Wokwi:

sketch.ino • diagram.json libraries.txt Library Manager ▾

```
1 #include <WiFi.h>//library for wifi
2 #include <PubSubClient.h>//library for MQTT
3 #include "DHT.h"// Library for dht11
4 #define DHTPIN 5 // what pin we're connected to
5 #define DHTTYPE DHT22 // define type of sensor DHT 11
6
7 DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and type of dht connect
8
9 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
10
11 //-----credentials of IBM Accounts-----
12
13 #define ORG "g4jr03"//IBM ORGANITION ID
14 #define DEVICE_TYPE "abcd"//Device type mentioned in ibm watson IOT Platform
15 #define DEVICE_ID "12345"//Device ID mentioned in ibm watson IOT Platform
16 #define TOKEN "12345678" //Token
17 String data3;
18 float h, t;
19
20 //----- Customise the above values -----
21
22 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
23 char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event perform a
24 char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT command type AND
25 char authMethod[] = "use-token-auth";// authentication method
26 char token[] = TOKEN;
27 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
28
29 //-----
30
31 WiFiClient wifiClient; // creating the instance for wificlient
32 PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client
33
34
35 void setup()// configuring the ESP32
```

Simulation



```
{ "temp":37.40,"humidity":86.00,"North":0,"South":0,"East":0,"West":0}
Publish ok
temp:37.40
humidity:86.00
Sending payload:
{"temp":37.40,"humidity":86.00,"North":0,"South":0,"East":0,"West":0}
Publish ok
```

NIKIL R
nikilagin2020@gmail.com

Discord

<> My projects

∞ The Club

📄 Feature Roadmap

🌐 Language

🔌 Logout

IBM Cloud :

The screenshot displays the IBM Cloud console interface. At the top, a browser window shows the URL `cloud.ibm.com`. The console header includes the IBM Cloud logo, a search bar, and navigation links for Catalog, Manage, and the user account (Gokulnath N's Account). A left-hand navigation menu contains icons for various services like Build, Watson, and Cloud Object Storage.

The main content area is titled "For you" and features several service tiles:

- Build**: Explore IBM Cloud with this selection of easy starter tutorials and services.
- Build a web app with Watson Speech to Text**: Deploy a conversational interface compatible with any application, device, or channel. (Getting started, 15 min)
- Get Started with Watson Studio**: Get started with using AI and Cloud Object Storage in 15 minutes. (Popular, 2 hr)
- Build a virtual machine**: Lift and shift your VMware workloads to the IBM Cloud. (Getting started, 7 min)
- Get started with Watson Discovery**: Get up to speed on Watson Discovery with step-by-step tutorials, deep-dive videos, and complete example working code. (Recommended, 2 hr)

Below the "For you" section, there are three main categories:

- User access**: Manage users. Enter email addresses below to jump directly into the invite user setup.
- News**: View all. Includes "IBM Tech Now: November 21, 2022" and "Introducing Badges to IBM Cloud Certification".
- Planned maintenance**: View all.

The bottom of the screen shows a Windows taskbar with open applications: Sprint-3 (1).pdf, sprint 1.docx, and sprint 1.pdf. The system clock indicates 9:14 PM on 11/24/2022.

IoT Device Creation:

The screenshot displays the IBM Watson IoT Platform dashboard. The top navigation bar includes tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains icons for various IoT functions. The main content area shows a list of devices, with one device selected and its details expanded. The 'Recent Events' tab is active, displaying a table of events.

Device List:

ID	Name	Status	Type	Last Seen
12345	abcd	Disconnected	Device	Nov 19, 2022 10:10 PM
abcd	1234	Disconnected	Device	Oct 22, 2022 8:32 PM

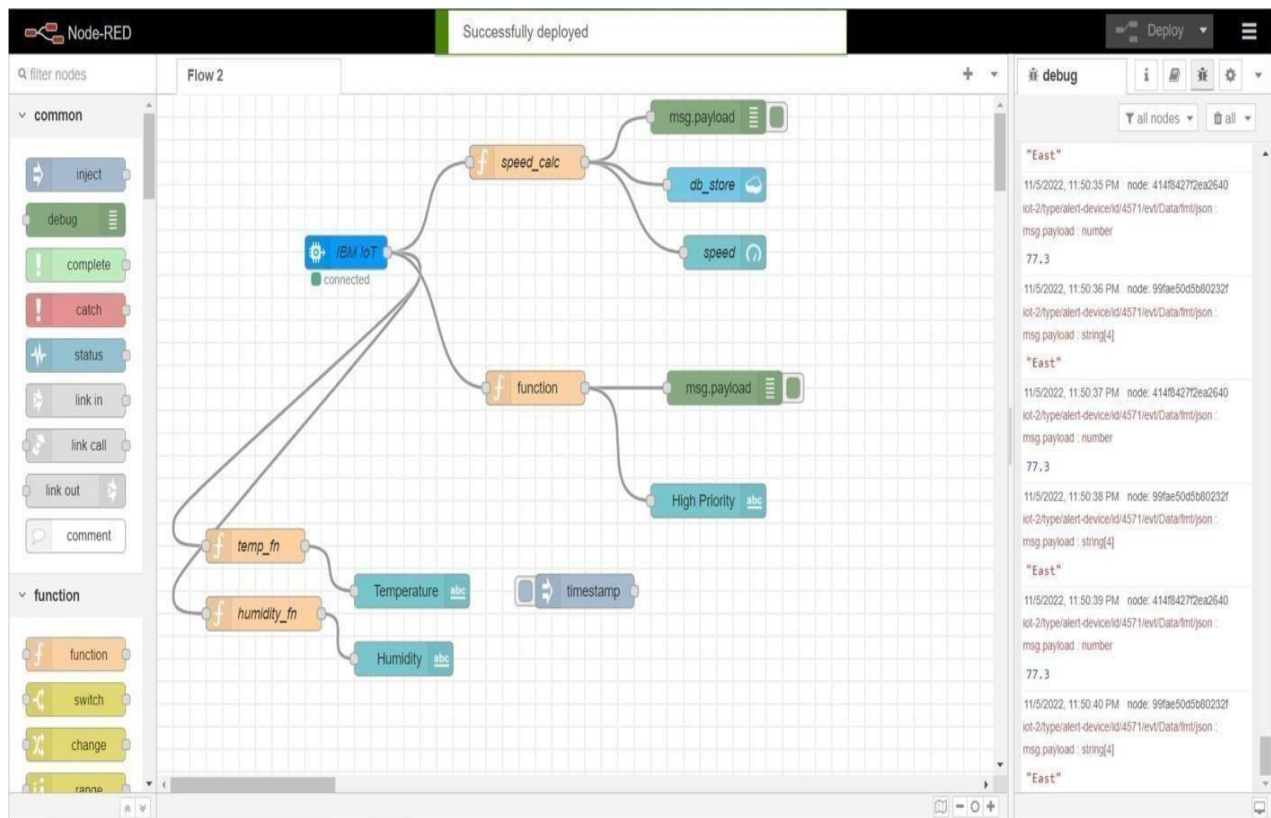
Recent Events:

The recent events listed show the live stream of data that is coming and going from this device.

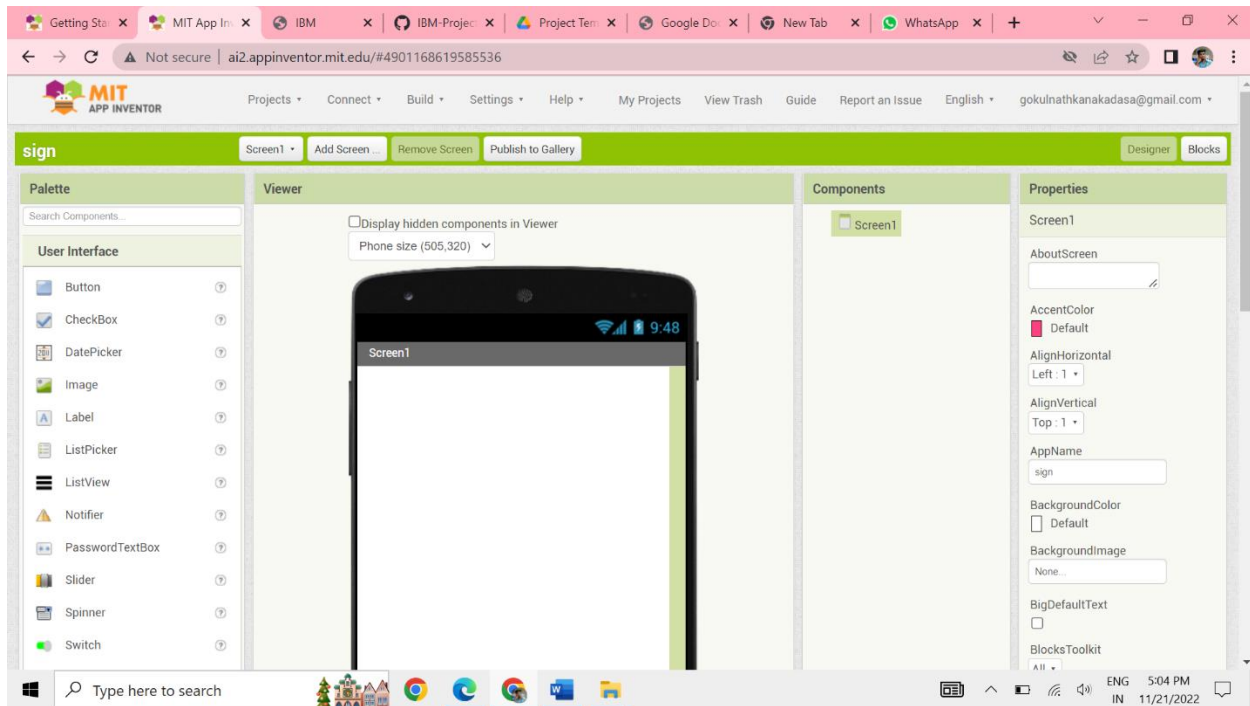
Event	Value	Format	Last Received
event_1	{\"randomNumber\":98,\"temp\":29,\"hum\":32}	json	a few seconds ago
event_1	{\"randomNumber\":43,\"temp\":12,\"hum\":36}	json	a few seconds ago
event_1	{\"randomNumber\":87,\"temp\":37,\"hum\":65}	json	a few seconds ago
event_1	{\"randomNumber\":84,\"temp\":64,\"hum\":94}	json	a few seconds ago
event_1	{\"randomNumber\":95,\"temp\":52,\"hum\":31}	json	a few seconds ago

2 Simulations running

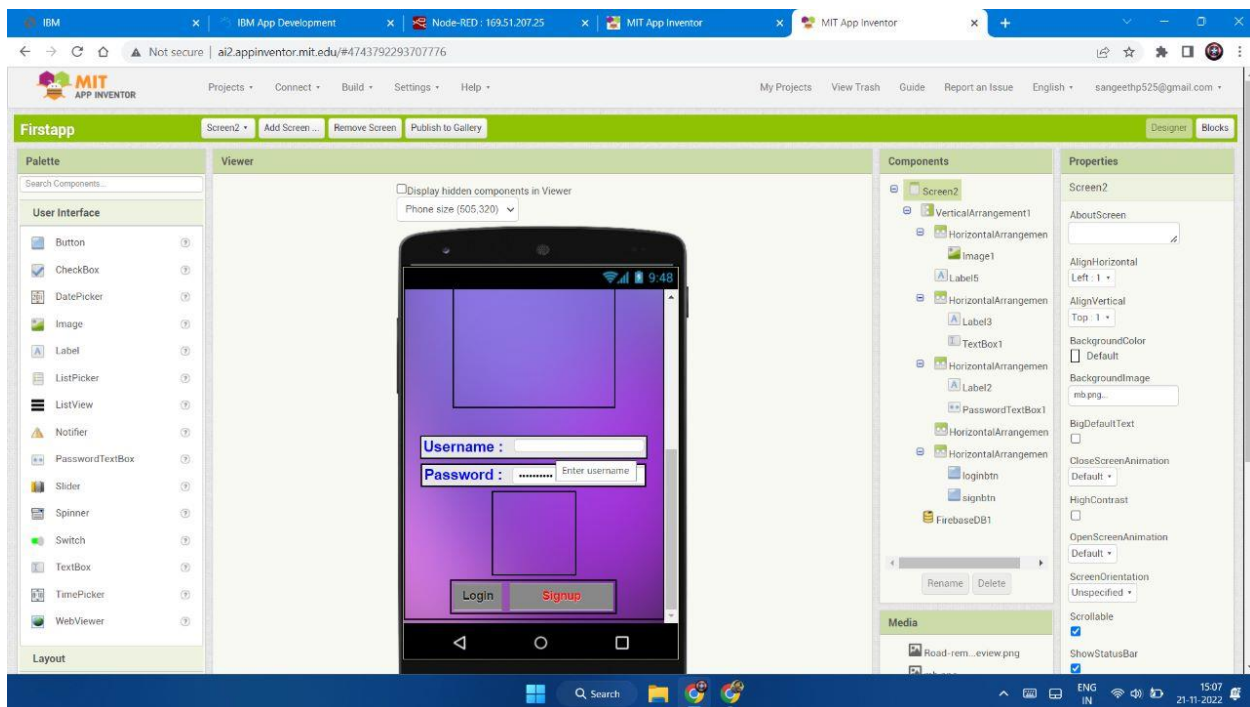
Node-Red:



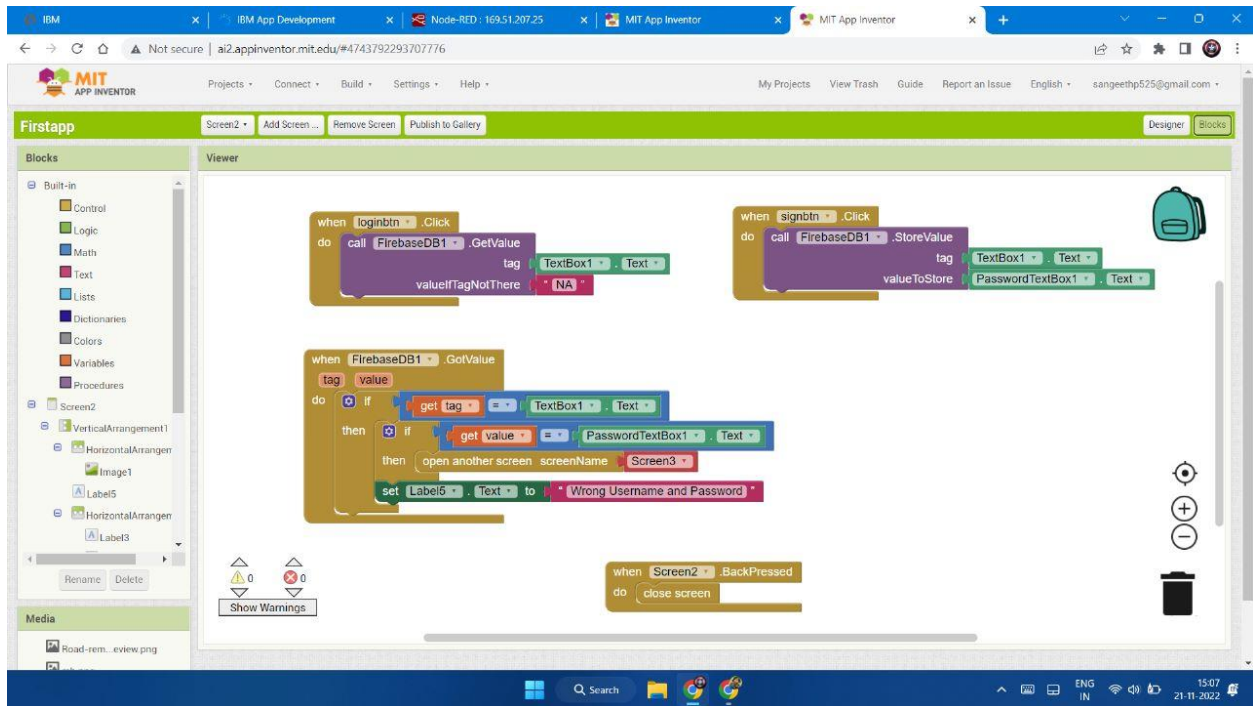
MIT App Inventor:



MIT App Inventor Working Design:



Design:



Node-Red Web UI and Out Put:

