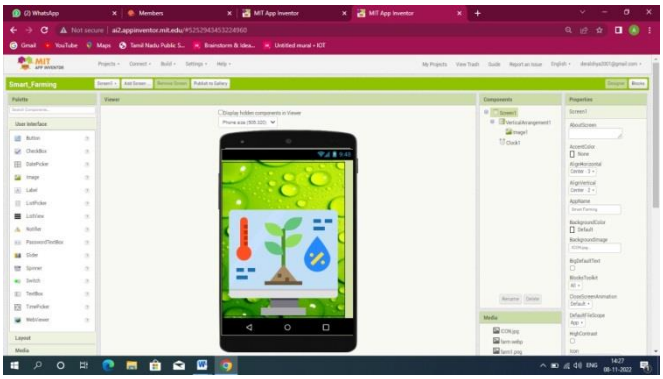
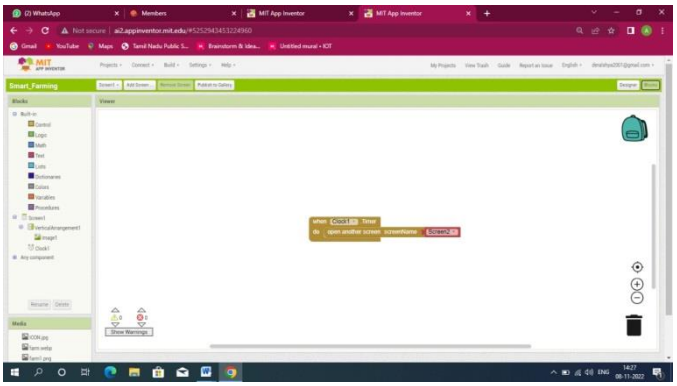
Sprint-3

TEAM ID	***PNT2022TMID34551***
PROJECT NAME	***Smart Farmer-IOT Enabled Smart Farming Application***

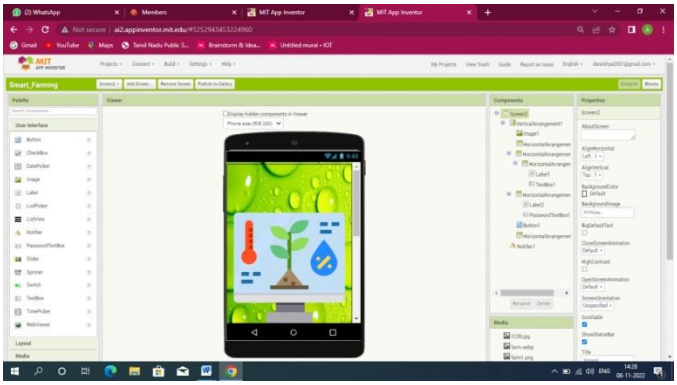
MIT APP INVENTOR: ***ICON PAGE***:

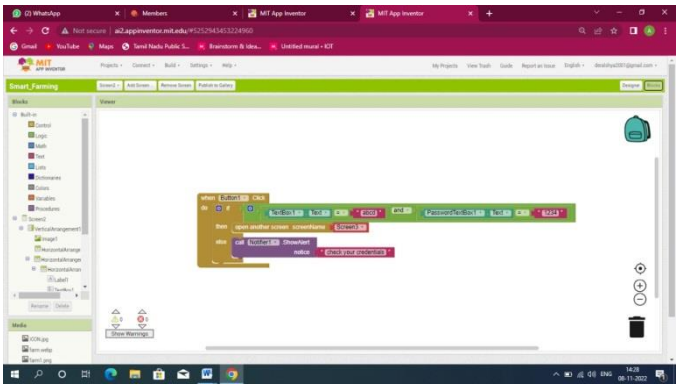
For Screen 1:



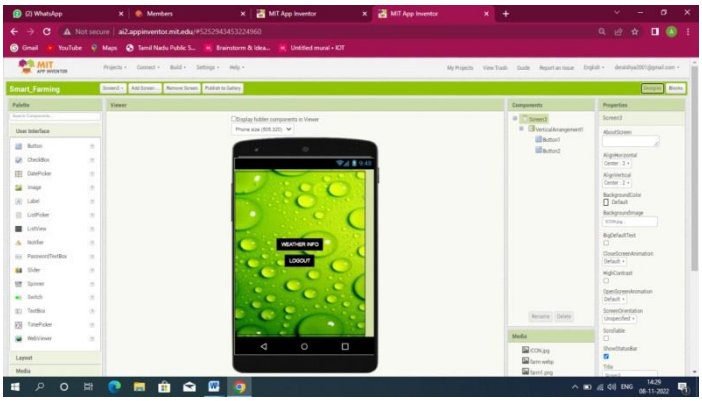


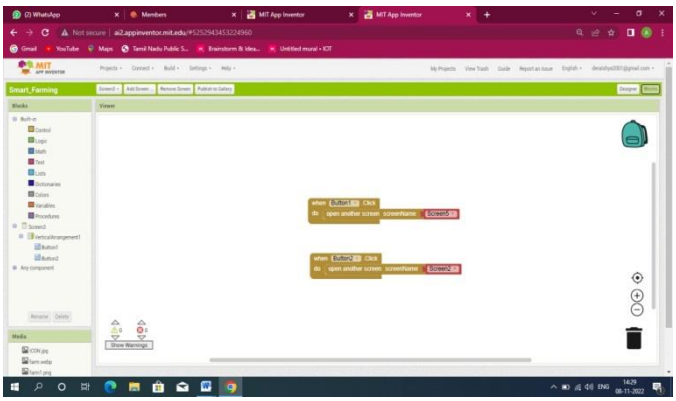
For Screen 2:



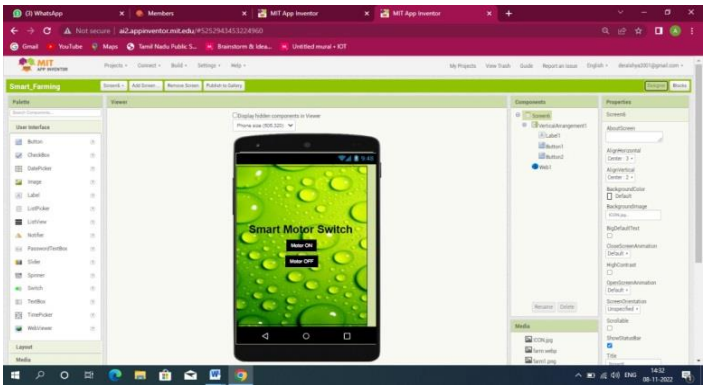


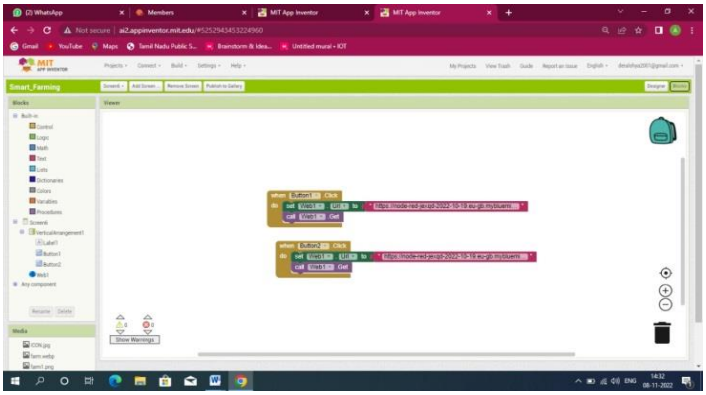
For Screen 3:



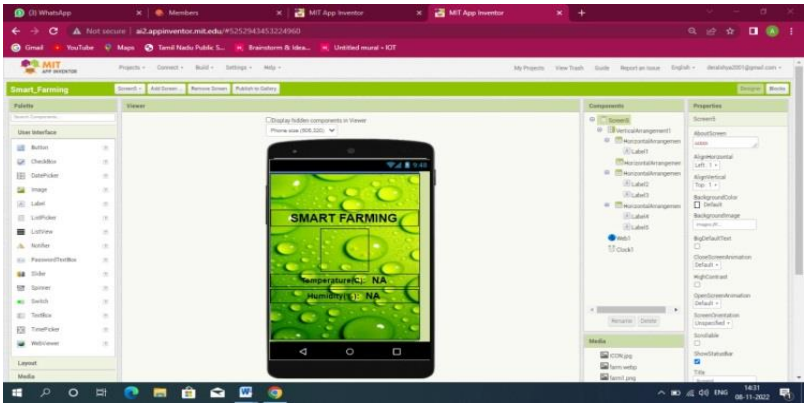


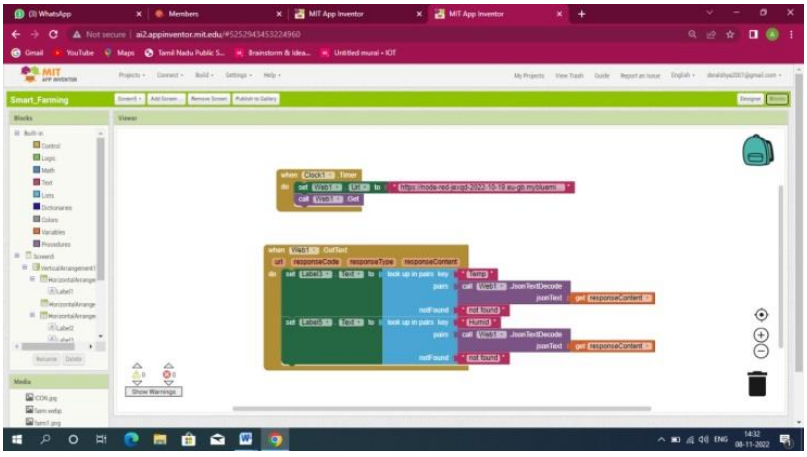
For Screen 4:





For Screen 5:





Code in Python IDLE:

PROGRAM:

smartfarmingiot.py - C:\Users\kavipriya devi\AppData\Local\Programs\Python\Python37\smartfarmingiot.py (3.7.0)

File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "b84wgs"
deviceType = "abi"
deviceId = "12345678"
authMethod = "token"
authToken = "87654321"

# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print ("Motor is ON")
    else :
        print ("Motor is OFF")

    #print(cmd)



try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....................................

except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11

        Temp=random.randint(0,100)
        Humid=random.randint(0,100)
```

Ln: 17 Col: 0

smartfarmingiot.py - C:\Users\kavipriya devi\AppData\Local\Programs\Python\Python37\smartfarmingiot.py (3.7.0)

File Edit Format Run Options Window Help

```
# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print ("Motor is ON")
    else :
        print ("Motor is OFF")

    #print(cmd)



try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....................................

except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11

        Temp=random.randint(0,100)
        Humid=random.randint(0,100)

        data = { 'Temp' : Temp, 'Humid': Humid }
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % Temp, "Humidity = %s %%" % Humid, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(1)

        deviceCli.commandCallback = myCommandCallback
```

Ln: 17 Col: 0

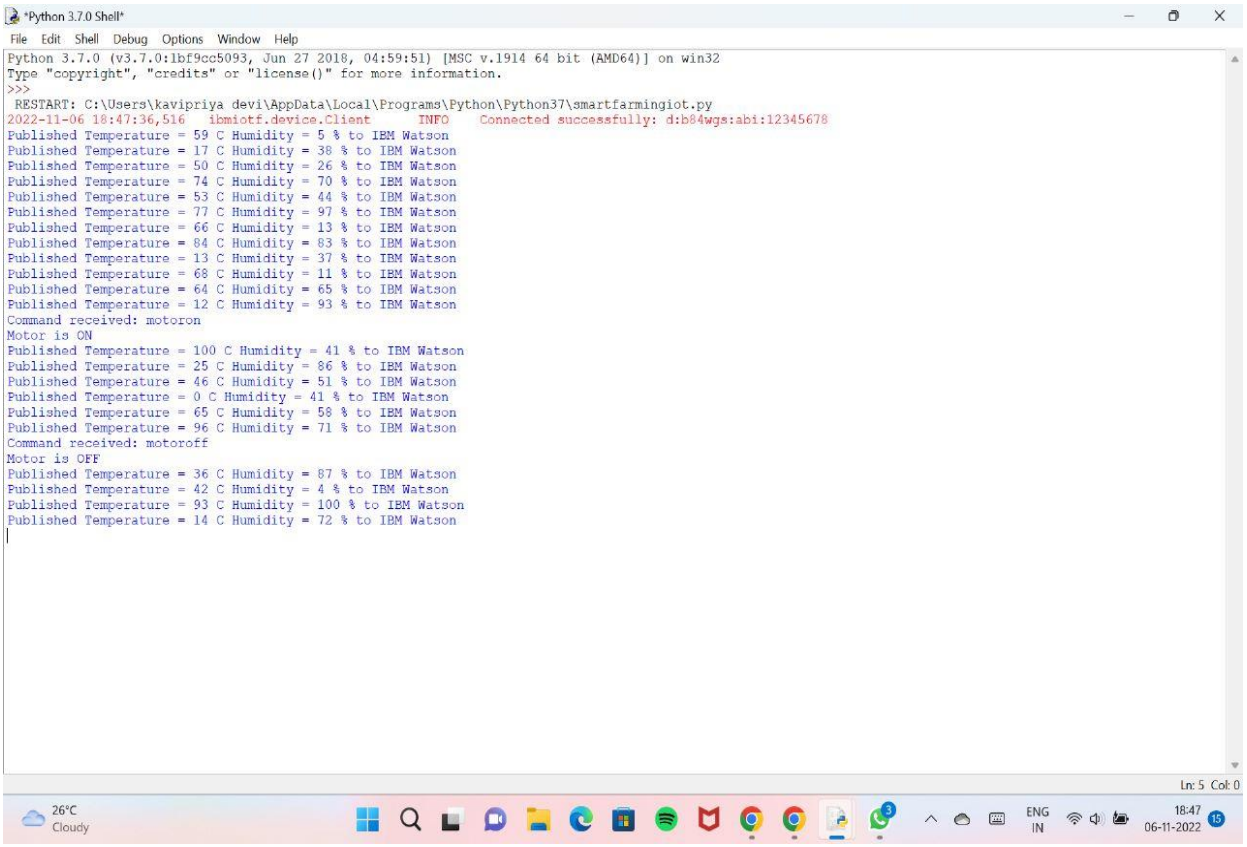
26°C Cloudy ENG IN 18:48 06-11-2022

Program used in the code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "b84wgs"
deviceType = "abi"
deviceId = "12345678"
authMethod = "token"
authToken = "87654321"


# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    else :
        print ("led is off")
    #print(cmd)
     try:
```

```
        deviceOptions = {"org": organization, "type": deviceType,
"id": deviceId, "auth-method": authMethod, "auth-token":
authToken}

        deviceCli = ibmiotf.device.Client(deviceOptions)

        #.....................................

except Exception as e:

        print("Caught exception connecting device: %s" % str(e))

        sys.exit()

# Connect and send a datapoint "hello" with value "world" into
the cloud as an event of type "greeting" 10 times

deviceCli.connect()

while True:

        #Get Sensor Data from DHT11

        temp=random.randint(0,100)

        Humid=random.randint(0,100)

        data = { 'temp' : temp, 'Humid': Humid }

        #print data

        def myOnPublishCallback():

            print ("Published Temperature = %s C" % temp,
"Humidity = %s %%" % Humid, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data,
qos=0, on_publish=myOnPublishCallback)

        if not success:

            print("Not connected to IoTF")

        time.sleep

        deviceCli.commandCallback = myCommandCallback
```

OUTPUT:


Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\kavipriya devi\AppData\Local\Programs\Python\Python37\smartfarmingiot.py
2022-11-06 18:47:36,516 ibmiotf.device.Client INFO Connected successfully: d:b84wgs:abi:12345678
Published Temperature = 59 C Humidity = 5 % to IBM Watson
Published Temperature = 17 C Humidity = 38 % to IBM Watson
Published Temperature = 50 C Humidity = 26 % to IBM Watson
Published Temperature = 74 C Humidity = 70 % to IBM Watson
Published Temperature = 53 C Humidity = 44 % to IBM Watson
Published Temperature = 77 C Humidity = 97 % to IBM Watson
Published Temperature = 66 C Humidity = 13 % to IBM Watson
Published Temperature = 84 C Humidity = 83 % to IBM Watson
Published Temperature = 13 C Humidity = 37 % to IBM Watson
Published Temperature = 68 C Humidity = 11 % to IBM Watson
Published Temperature = 64 C Humidity = 65 % to IBM Watson
Published Temperature = 12 C Humidity = 93 % to IBM Watson
Command received: motoron
Motor is ON
Published Temperature = 100 C Humidity = 41 % to IBM Watson
Published Temperature = 25 C Humidity = 86 % to IBM Watson
Published Temperature = 46 C Humidity = 51 % to IBM Watson
Published Temperature = 0 C Humidity = 41 % to IBM Watson
Published Temperature = 65 C Humidity = 58 % to IBM Watson
Published Temperature = 96 C Humidity = 71 % to IBM Watson
Command received: motoroff
Motor is OFF
Published Temperature = 36 C Humidity = 87 % to IBM Watson
Published Temperature = 42 C Humidity = 4 % to IBM Watson
Published Temperature = 93 C Humidity = 100 % to IBM Watson
Published Temperature = 14 C Humidity = 72 % to IBM Watson
Ln: 5 Col: 0
26°C
Cloudy
ENG
IN
18:47
06-11-2022