Project Development Phase
Sprint 1

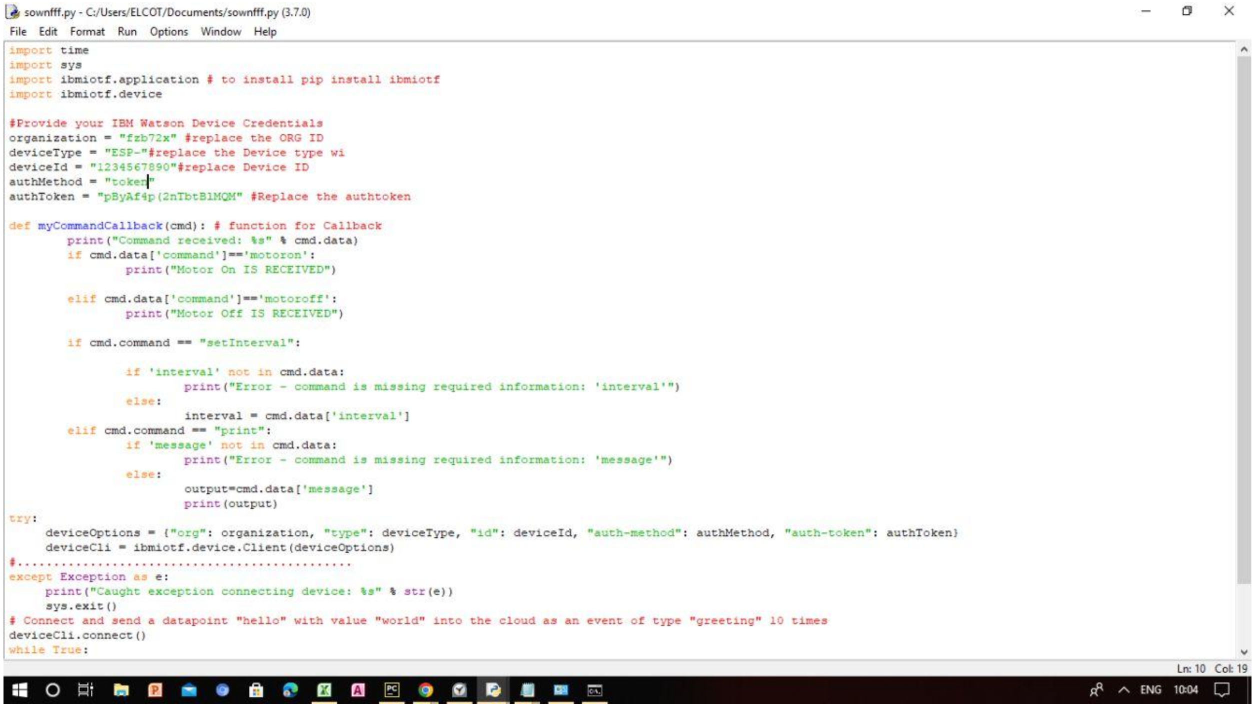
Team ID	PNT2022TMID30928
Project Title	Iot Based Smart Crop Protection System for Agriculture
Date	21 October 2022

In sprint 1, we designed the python code that links with the sensors to the device and processes the required output. The python code and their output after execution are given below.

Python code:

```
import time
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "fzb72x" #replace the ORG ID
deviceType = "ESP-"#replace the Device type wi
deviceId = "1234567890"#replace Device ID
authMethod = "token"
authToken = "pByAf4p(2nTbtBIMQM" #Replace the authtoken
def myCommandCallback(cmd): # function for Callback
    print("Command received: %s" % cmd.data)
    if cmd.data['command']=='motoron':
        print("Motor On IS RECEIVED")

    elif cmd.data['command']=='motoroff':
        print("Motor Off IS RECEIVED")

    if cmd.command == "setInterval":

        if 'interval' not in cmd.data:
            print("Error - command is missing required information: 'interval'")
        else:
            interval = cmd.data['interval']
    elif cmd.command == "print":
        if 'message' not in cmd.data:
            print("Error - command is missing required information: 'message'")
        else:
            output=cmd.data['message']
            print(output)

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

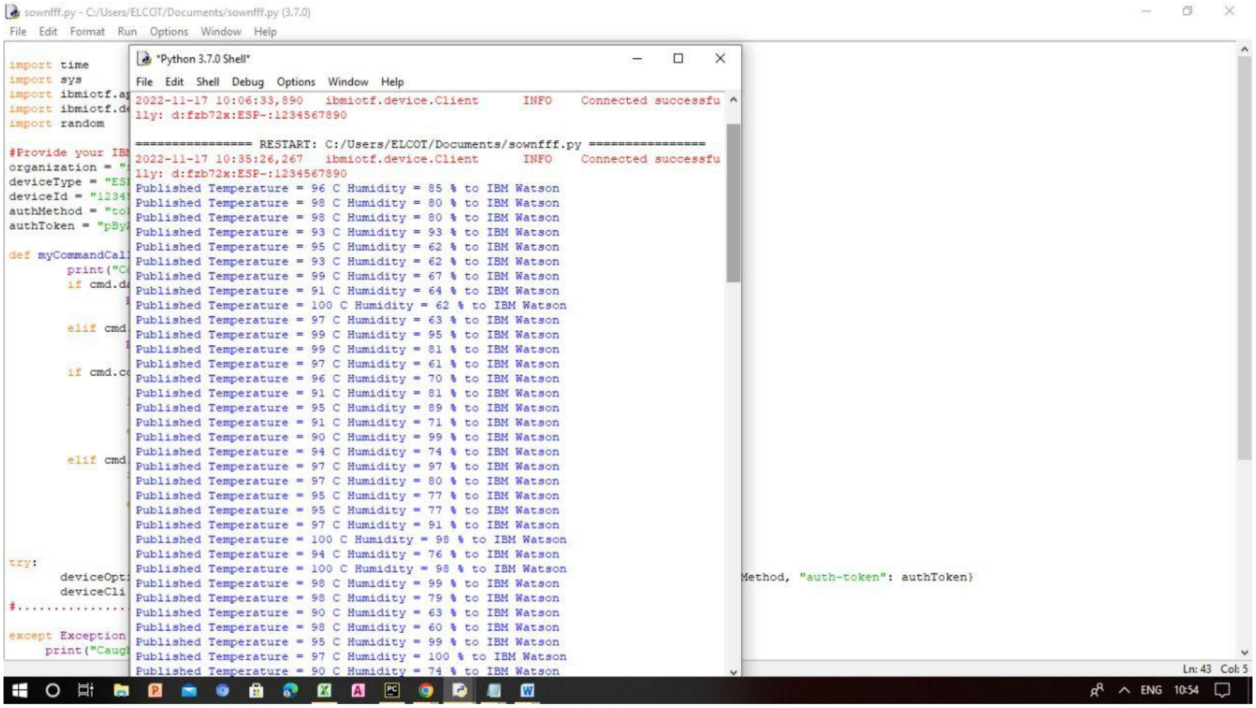
```
#.........................................


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:
        temp=random.randint(90,100)
        Humid=random.randint(60,100)
        data = {'temp': temp, 'Humid': Humid}
        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(2)
        deviceCli.commandCallback = myCommandCallback


# Disconnect the device and application from the cloud
deviceCli.disconnect()
```



sownfff.py - C:/Users/ELCOT/Documents/sownfff.py (3.7.0)

```
import time
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device

#Provide your IBM Watson Device Credentials
organization = "fzb72x" #replace the ORG ID
deviceType = "ESP-"#replace the Device type wi
deviceId = "1234567890"#replace Device ID
authMethod = "token"
authToken = "pByAf4p(2nTbtB1MQM" #Replace the authtoken

def myCommandCallback(cmd): # function for Callback
        print("Command received: %s" % cmd.data)
        if cmd.data['command']=='motoron':
                print("Motor On IS RECEIVED")

        elif cmd.data['command']=='motoroff':
                print("Motor Off IS RECEIVED")

        if cmd.command == "setInterval":

                if 'interval' not in cmd.data:
                        print("Error - command is missing required information: 'interval'")
                else:
                        interval = cmd.data['interval']
        elif cmd.command == "print":
                if 'message' not in cmd.data:
                        print("Error - command is missing required information: 'message'")
                else:
                        output=cmd.data['message']
                        print(output)
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:
```

OUTPUT:



Python 3.7.0 Shell

```
2022-11-17 10:06:33,890   ibmiotf.device.Client     INFO    Connected successfu
lly: d:fzb72x:ESP-:1234567890

================ RESTART: C:/Users/ELCOT/Documents/sownfff.py ================
2022-11-17 10:35:26,267   ibmiotf.device.Client     INFO    Connected successfu
lly: d:fzb72x:ESP-:1234567890
Published Temperature = 96 C Humidity = 85 % to IBM Watson
Published Temperature = 98 C Humidity = 80 % to IBM Watson
Published Temperature = 98 C Humidity = 80 % to IBM Watson
Published Temperature = 93 C Humidity = 93 % to IBM Watson
Published Temperature = 95 C Humidity = 62 % to IBM Watson
Published Temperature = 93 C Humidity = 62 % to IBM Watson
Published Temperature = 99 C Humidity = 67 % to IBM Watson
Published Temperature = 91 C Humidity = 64 % to IBM Watson
Published Temperature = 100 C Humidity = 62 % to IBM Watson
Published Temperature = 97 C Humidity = 63 % to IBM Watson
Published Temperature = 99 C Humidity = 95 % to IBM Watson
Published Temperature = 99 C Humidity = 81 % to IBM Watson
Published Temperature = 97 C Humidity = 61 % to IBM Watson
Published Temperature = 96 C Humidity = 70 % to IBM Watson
Published Temperature = 91 C Humidity = 81 % to IBM Watson
Published Temperature = 95 C Humidity = 89 % to IBM Watson
Published Temperature = 91 C Humidity = 71 % to IBM Watson
Published Temperature = 90 C Humidity = 99 % to IBM Watson
Published Temperature = 94 C Humidity = 74 % to IBM Watson
Published Temperature = 97 C Humidity = 97 % to IBM Watson
Published Temperature = 97 C Humidity = 80 % to IBM Watson
Published Temperature = 95 C Humidity = 77 % to IBM Watson
Published Temperature = 95 C Humidity = 77 % to IBM Watson
Published Temperature = 97 C Humidity = 91 % to IBM Watson
Published Temperature = 100 C Humidity = 98 % to IBM Watson
Published Temperature = 94 C Humidity = 76 % to IBM Watson
Published Temperature = 100 C Humidity = 98 % to IBM Watson
Published Temperature = 98 C Humidity = 99 % to IBM Watson
Published Temperature = 98 C Humidity = 79 % to IBM Watson
Published Temperature = 90 C Humidity = 63 % to IBM Watson
Published Temperature = 98 C Humidity = 60 % to IBM Watson
Published Temperature = 95 C Humidity = 99 % to IBM Watson
Published Temperature = 97 C Humidity = 100 % to IBM Watson
Published Temperature = 90 C Humidity = 74 % to IBM Watson
```

sownfff.py - C:/Users/ELCOT/Documents/sownfff.py (3.7.0)
File Edit Format Run Options Window Help
File Edit Shell Debug Options Window Help
Ln: 43 Col: 5
ENG 10:54