# PROJECT DEVELOPMENT PHASE

**SPRINT-3 CODING**

Date 08 November 2022

Team ID PNT2022TMID12310

Project Name IoT Based Smart Crop Protection System for Agriculture

Maximum Marks 8 Marks

**CODING:**

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

#Provide your IBM Watson Device Credentials organization = "mw0wqj" deviceType = "real" deviceId = "realtime" authMethod = "token" authToken = "vasuki123"

# 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, "authmethod": 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

ph=random.randint(0,14) turb=random.randint(0,100)

data = { 'ph' : ph, 'turb': turb }

#print data def myOnPublishCallback(): print ("Published PH Level

= %s C" % ph, "Turbidity = %s C" % turb, "to IBM Watson")

success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on\_publish=myOnPublishCallback) if not success:

print("Not connected to IoTF") time.sleep(5)

deviceCli.commandCallback = myCommandCallback

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