**Sprint 1**

|  |  |
| --- | --- |
| **Team ID** | PNT2022TMID17579 |
| **Project Name** | IoT-Enabled Smart Farming Application |

# Problem Statement

* Farmers are to be present at farm for its maintenance irrespective of the weather conditions. They have to ensure that the crops are well watered and the farm status is monitored by them physically.
* Farmer have to stay most of the time in field in order to get a good yield. In difficult times like in the presence of pandemic also they have to work hard in their fields risking their lives to provide food for the country.

**PROGRAM CODING:**

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

#IBM

organization = "3nc6qc" deviceType = "node"

deviceId = "008"

authMethod = "use-token-auth" authToken = "6383637992"

#Gpio

def mycommandCallback(cmd): print("Command Received: %s" %cmd.data['command'])

status = cmd.data['command'] if status=="lighton": print("LED is ON")

elif status=="lightoff":

print("LED is OFF") else:

print("please send proper command") 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()

#CONNECCT

deviceCli.connect()

while True:

temp=random.randint(0,100) hum=random.randint(0,100) data={'temp':temp,'hum':hum}

def myOnPublishCallback():

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

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

time.sleep(10)

deviceCli.commandCallback = mycommandCallback

#Disconnect deviceCli.disconnect()

**Screenshots:**

