**SRM VALLIAMMAI ENGINEERING COLLEGE**

**Department of Computer Science and Engineering**

# Smart Farmer-IOT Enabled Smart Farming Application

**IBM NALAIYATHIRAN**

# Source Code

| | |
|---|---|
| **TITLE** | **Smart Farmer-IOT Enabled Smart Farming Application** |
| **DOMAIN NAME** | INTERNET OF THINGS |
| **TEAM ID** | PNT2022TMID21697 |
| **LEADER NAME** | RAJALAKSHMI V |
| **TEAM MEMBER NAME** | HARIPRASATH.T<br>PORSELVI.S<br>MUTHUSATHISH |
| **MENTOR NAME** | LALITHA.A |

# SOURCE CODE

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "nckdv7"
deviceType = "NodeMCU"
deviceId = "12345"
authMethod = "token"
authToken = "12345678"
# 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)
pulse=random.randint(0,100)
moisture= random.randint(0,100)
humidity=random.randint(0,100);
lat = 17
lon = 18
data = { 'temp' : temp, 'humidity' : humidity, 'Soil Moisture' :
moisture}
#print data
def myOnPublishCallback():
print ("Published Temperature = %s C" % temp, "Humidity
= %s %%" % humidity, "Soil Moisture = %s %%" % moisture,"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
# Disconnect the device and application from the cloud
deviceCli.disconnect()
```