CODE FOR THE PROJECT

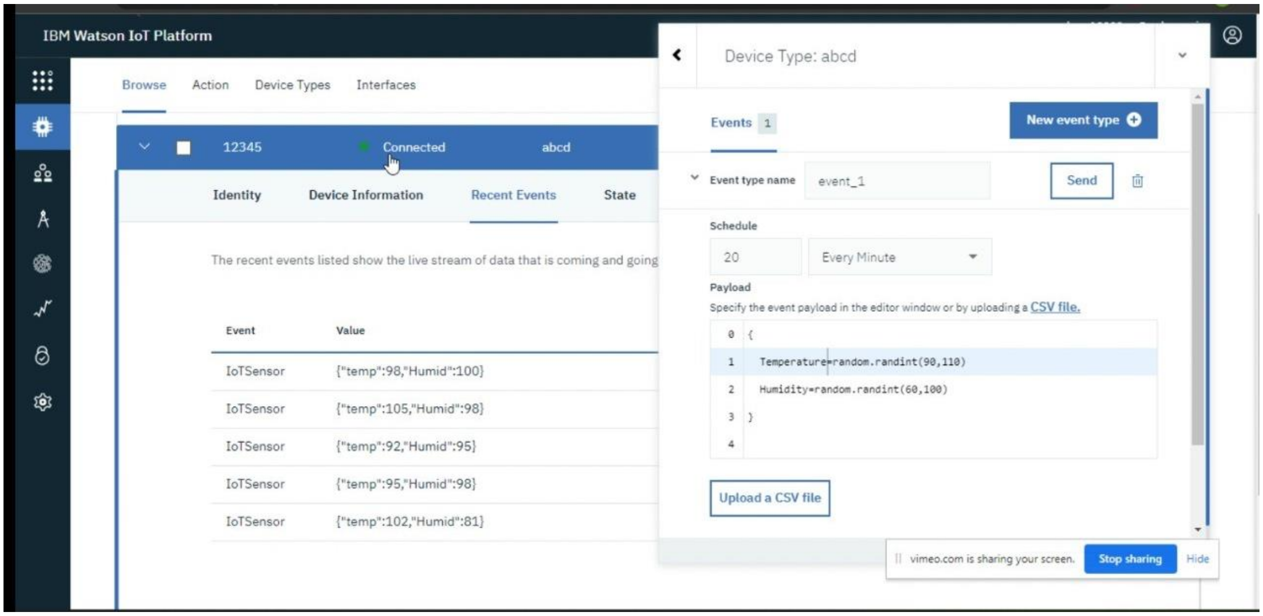
SOURCE CODE:

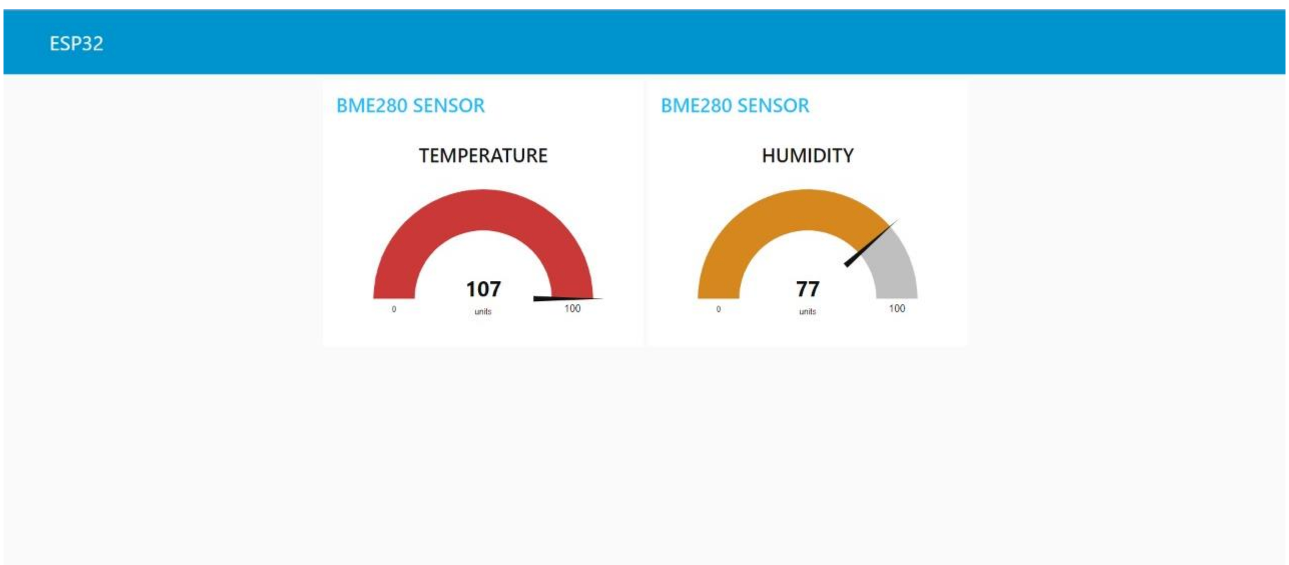
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "de9kb6"
deviceType = "123"
deviceId = "123"
authMethod = "token"
authToken = "12345678"

# Initialize 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()

# 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(90,110)
        Humid=random.randint(60,100)

        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(10)

        deviceCli.commandCallback = myCommandCallback

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

OUTPUT:


IBM Watson IoT Platform
Browse
Action
Device Types
Interfaces
12345
Connected
abcd
Identity
Device Information
Recent Events
State
The recent events listed show the live stream of data that is coming and going
Event
Value
IoTSensor
{"temp":98,"Humid":100}
IoTSensor
{"temp":105,"Humid":98}
IoTSensor
{"temp":92,"Humid":95}
IoTSensor
{"temp":95,"Humid":98}
IoTSensor
{"temp":102,"Humid":81}
Device Type: abcd
Events 1
New event type
Event type name
event_1
Send
Schedule
20
Every Minute
Payload
Specify the event payload in the editor window or by uploading a CSV file.
{
Temperature=random.randint(90,110)
Humidity=random.randint(60,100)
}
Upload a CSV file
vimeo.com is sharing your screen.
Stop sharing
Hide


ESP32
BME280 SENSOR
TEMPERATURE
107
units
0
100
BME280 SENSOR
HUMIDITY
77
units
0
100