REALTIME WEATHER BASED SPRINKLER SYSTEM FOR GOLF COURSE

Submitted by:

Harshitha B

Mridula Chandrakar

Sunitha V

1.INTRODUCTION:

1.1 Overview:

Continuous monitoring and storage of weather and soil moisture information. Alert is generated if the soil moisture is above the threshold value. The water sprinklers can be controlled remotely using mobile app. Less latency in communication from device to cloud with MQTT.

1.2 Purpose:

By using this project one can view the temperature, humidity and the soil moisture details in real time. He/she can also switch on or off the motor and sprinkler using a web application or mobile application.

2. LITERATURE SURVEY:

2.1 Existing Problem:

The owner of the golf course has to appoint a person to properly turn on or off the sprinkler but in doing so the person may not know if rain is going to come or not also he doesn't calculate the moisture content present in the soil. Therefore, if the person switches on the sprinkler then the soil may get more wet than it is supposed to be.

2.2 Proposed Solution:

The solution proposed to solve the problem is an integrated system using IBM IoT Platform where all the devices are connected through IBM Watson and two interfaces one is website and other is a mobile application is created to monitor the current

temperature and humidity readings and with the help of the readings soil moisture can be calculated thus sprinkler can be turned on and off at any time from any place.

3. THEORITICAL ANALYSIS:

3.1 Block Diagram

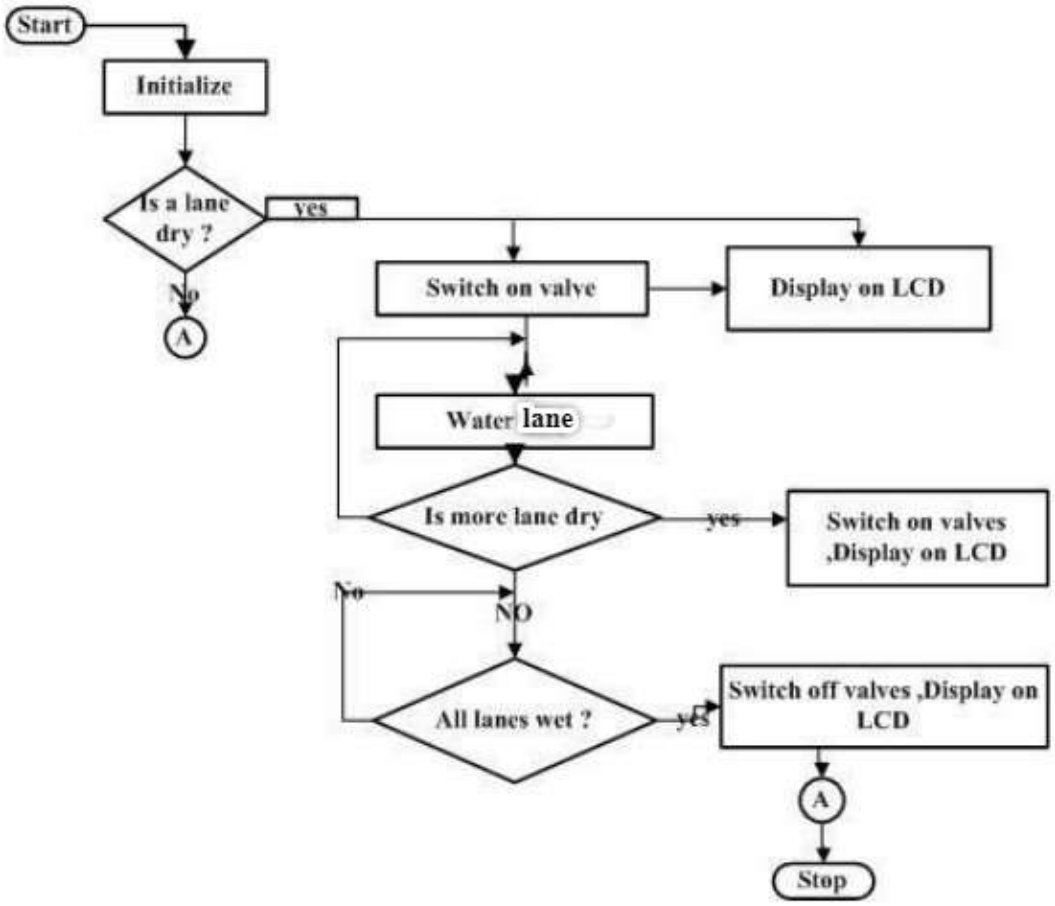
3.2 Software Designing:

In the software designing part create a IBM cloud platform. In this design the raspberry pi model is used. the software should be design by taking avalues from the dht11 sensor and then sent to the IBM cloud services and then the data send to the mobile application which was developed using MIT app inventor. Here we use python language for coding, Node-Red ,etc.

4. EXPERIMENTAL INVESTIGATIONS:

By integrating the applications using Watson IoT all the devices can be connected and thus they can be controlled from any place and at any time. Real time data can be seen from the mobile application and the online web application.

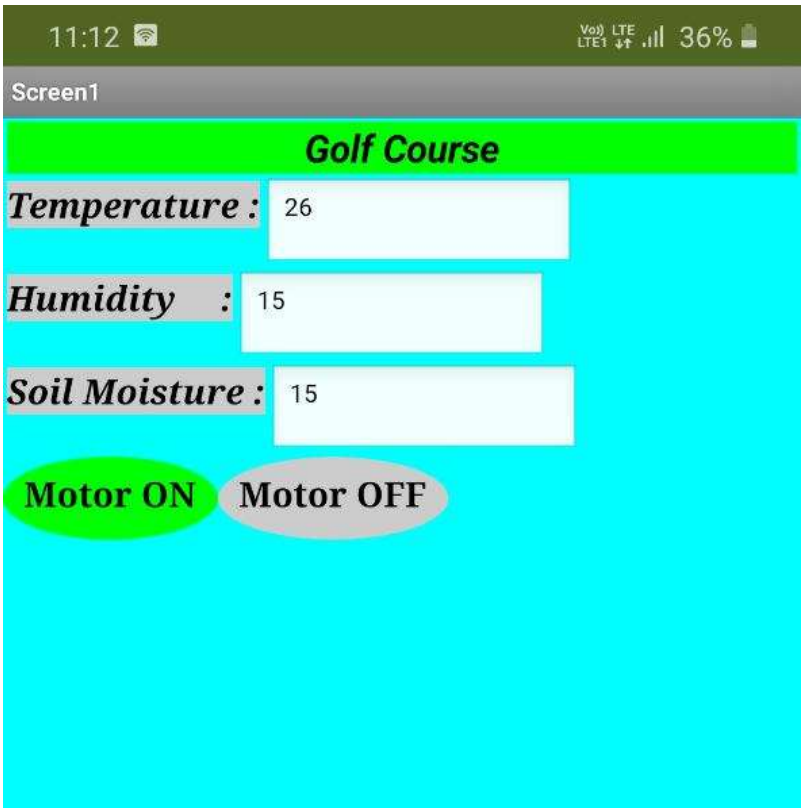
5.FLOWCHART:

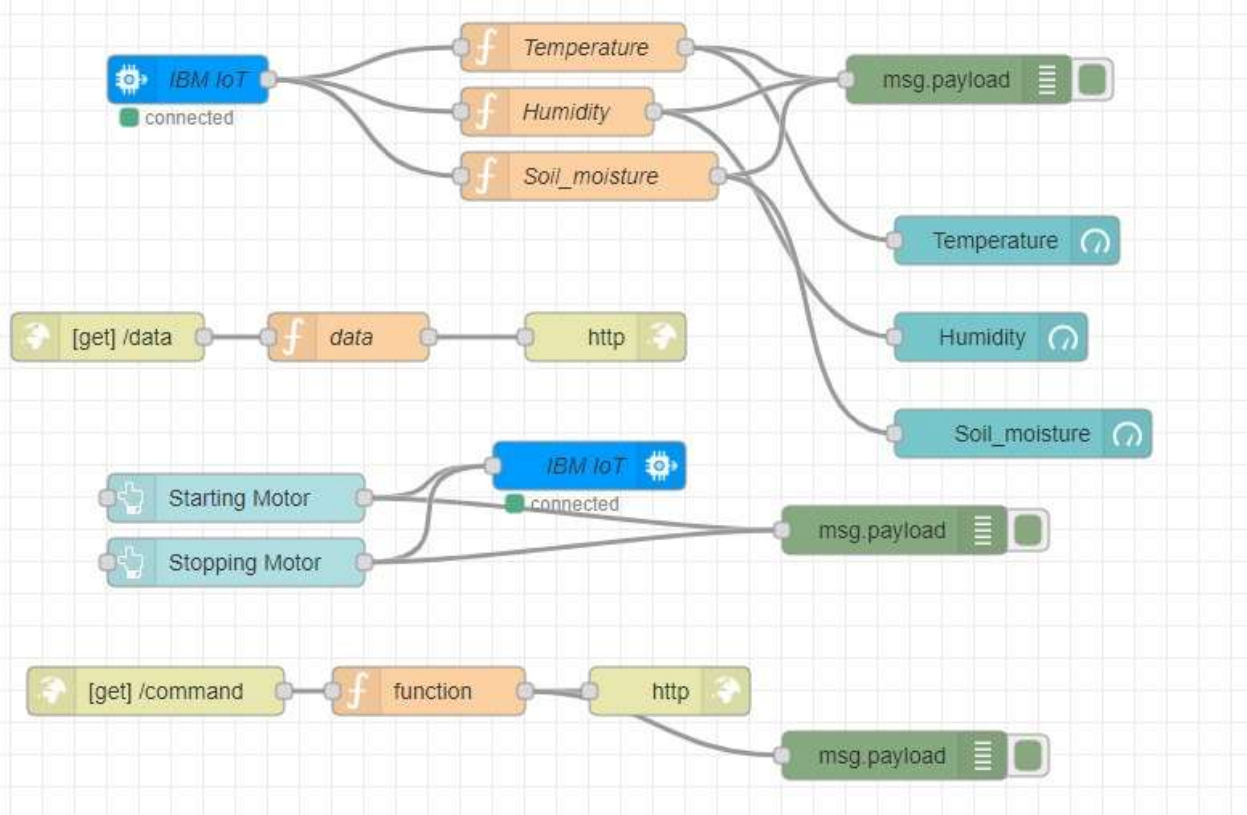

Start
Initialize
Is a lane dry ?
yes
No
A
Switch on valve
Display on LCD
Water lane
Is more lane dry
yes
Switch on valves ,Display on LCD
No
NO
All lanes wet ?
yes
Switch off valves ,Display on LCD
A
Stop

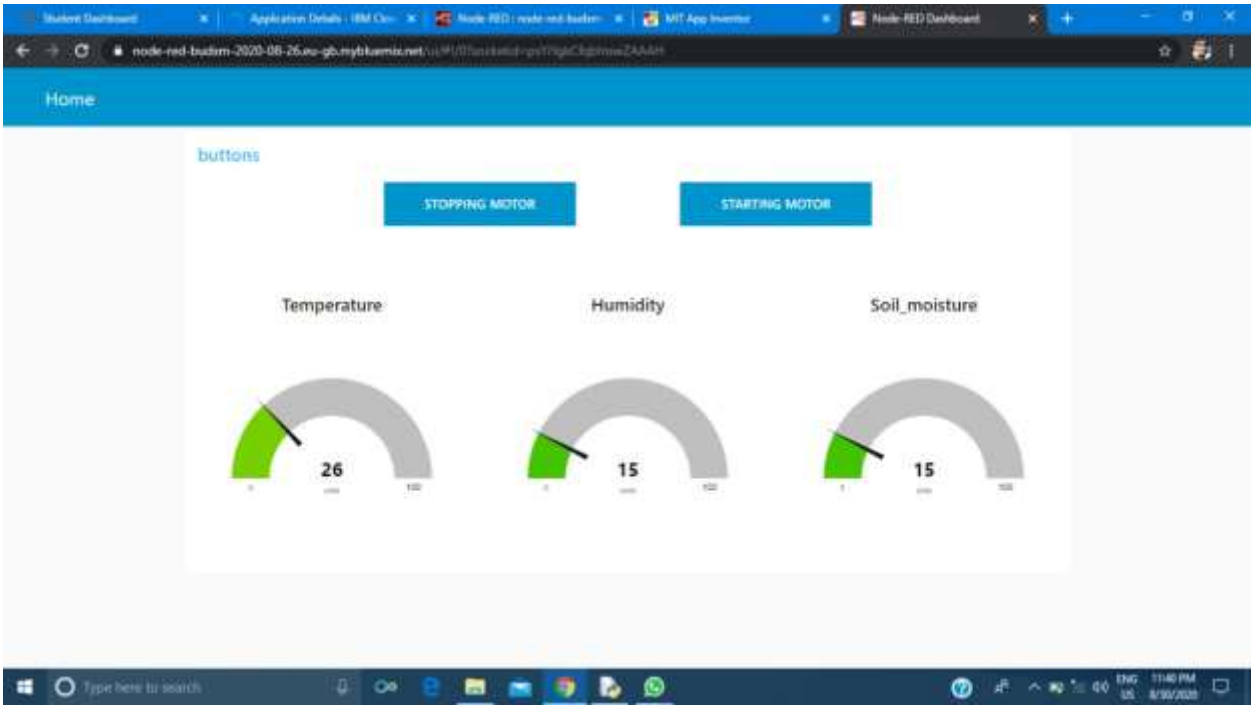
.

6. RESULT:

One can view the temperature, humidity and soil moisture anywhere and anytime and can control the sprinkler to spray water in the golf course.




IBM IoT
connected
Temperature
Humidity
Soil_moisture
msg.payload
Temperature
Humidity
Soil_moisture
[get] /data
data
http
Starting Motor
Stopping Motor
IBM IoT
connected
msg.payload
[get] /command
function
http
msg.payload


Home
buttons
STOPPING MOTOR
STARTING MOTOR
Temperature
Humidity
Soil_moisture
26
15
15

7. ADVANTAGES:

- **Efficient resource utilization:** If we know the functionality and the way that how each device work we definitely increase the efficient resource utilization as well as monitor natural resources.
- **Minimize human effort:** As the devices of IoT interact and communicate with each other and do lot of task for us, then they minimize the human effort.
- **Save time:** As it reduces the human effort then it definitely saves out time. Time is the primary factor which can save through IoT platform.
- **Improve security:** Now, if we have a system that all these things are interconnected then we can make the system more secure and efficient.
- **Enhance data collection**
- **Supervision by online:** We get regular weather report.

Because the temperature and moisture sensor will give regular notifications.

- Save a ton of money by reducing water waste.
- Enhanced landscape health and beauty.
- Helps us to prepare for the future of water.

DISADVANTAGES:

- The weather based smart water sprinkler system is a bit expansive depending on the size of your ground, we will need more systems.

• The Golf course has the very large area. To spread over all the ground, we require the more systems and observation become difficult.

• User must have some basic knowledge to use the web and mobile applications.

• The server must be running all time and it must be maintained properly.

8. APPLICATIONS:

• Easy and intuitive app controls.

• Real time monitoring of weather and weather awareness.

• Anytime and anywhere access.

• Also can be used for smart irrigation.

9. CONCLUSION:

Real time weather based smart sprinkler for golf course was developed, web and mobile applications were created to monitor the temperature, humidity and soil moisture content. By this we can decrease the Water wastage.We observe the weather conditions in the Golf course by online.We switch ON/OFF of water sprinkler by online using IOT device.

10. FUTURE SCOPE:

- In future the application can be modified to be used in smart irrigation systems to save water.
- Voice controls can also be added as additional features for the system
- And also in future everywhere in every field people use the Internet of Things projects.
- Because the projects will reduce the human efforts and wastage of nature.

11. BIBILOGRAPHY:

1. www.wikipedia.com
2. Smartbridge vidoes
3. www.google.com

APPENDIX:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
import requests
from pprint import pprint

#Mentioning IBM IOT Credentials
organisation= "yhxaeo"
deviceType= "raspberrypi"
```

```
deviceId= "Sensor"
authMethod= "token"
authToken= "9449320176"
data=0

#function to define commands from the Web UI
def myCommandCallback(cmd):
   print("Command received: %s" %cmd.data)

#connecting the IBM device to our python code
try:
   deviceOptions = {"org": organisation, "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()




deviceCli.connect()

while True:
   hum= random.randint(0,100)
   temp= random.randint(0,50)

   #code for Soil moisture
   Adcvalue = random.randint(511,1023)
   output_V = Adcvalue/1023
   Moisture = round(100- (output_V*100))
   #Generate Message
   if Moisture <15 and hum >95:
         Message= "Moisture is less but weather will take care."
   elif (Moisture >25 and hum >95) or Moisture >25:
         Message= "Dont turn the sprinklers on even by mistake!"
```

```
    elif (Moisture > 15 and Moisture <25):
            Message= "Perfect Soil"
    elif Moisture <15 and hum <60:
            Message= "Turn the sprinklers on"
    elif Moisture <15:
            print("Starting Motor")
    elif Moisture >25:
            print("Stopping Motor")



    #print(data)
    def myOnPublishCallback():

        print("Soil Moisture= %s %%" %Moisture)
        print("Published temperature= %s C" %temp, "Humidity= %s %%" %hum,"to IBM
WATSON")
    #code for generating message and toggling motors
        if Moisture <15 and hum >95:
            print("Moisture is less but weather will take care." )
        elif (Moisture >25 and hum >95) or Moisture >25:
            print("Dont turn the sprinklers on even by mistake!")
            print("Stopping Motor")
        elif (Moisture > 15 and Moisture <25):
            print("Perfect Soil")

        elif Moisture <15 and hum <60:
            print("Turn the sprinklers on")
            print("Starting Motor")
        elif Moisture <15:
            print("Starting Motor")
        elif Moisture >25:
            print("Stopping Motor")

    data= { 'Temperature': temp, 'Humidity': hum,'Soil_moisture': Moisture,'Suggestion':
Message }
```

```
   success=          deviceCli.publishEvent("Smart","json",data,qos=          0,on_publish=
myOnPublishCallback )
   if not success:
      print("Not connected to IoTF")
   time.sleep(10)
   deviceCli.commandCallback= myCommandCallback
deviceCli.disconnect()
```