# IOT(Internet of Things)

# Project on smart irrigation system using IOT

# by team 14

1)Y.B.N.V.S Suneela (17481A04K7)

2)T.L.Aishwarya Lakshmi (17481A04I4)

3)T.Venkata Lakshmi (17481A04I6)

4)V.D.N.V Prasanthi (16481A04M9)

**TABLE OF CONTENTS:**

**1 INTRODUCTION**

**1.1 OverView**

**1.2 Purpose**

**2 LITERATURE SURVEY**

**2.1 Existing problem**

**2.2 Proposed solution**

**3 THEORITICAL ANALYSIS**

**3.1 Block diagram**

**3.2 Hardware/software designing**

**4 EXPERIMENTAL INVESTIGATION**

**5 FLOWCHART**

**6 RESULT**

**7 ADVANTAGES & DISADVANTAGES**

**8 APPLICATIONS**

**9 CONCLUSION**

**10 FUTURE SCOPE**

**11 BIBLIOGRAPHY**

**APPENDIX**

**A.SOURCE CODE**

**1.INTRODUCTION :**

**1.1 OVERVIEW:**

Agriculture is the backbone for a country’s development. Basically, agriculture

depends on the monsoons which do not have enough water sources. To overcome

this problem, the irrigation system is employed in the field of agriculture. In this

system, based on the soil type, water will be provided to the field. In agriculture,

there are two things, namely, the moisture content of the soil as well as the fertility

of the soil. At the present time, there are several types of techniques available for

irrigation to reduce the need for rain. This type of technique is driven by on/off

schedule using electrical power. This project is about implementation of a Smart

Irrigation System using IoT that can serve as a helping hand to farmer.

**1.2 PURPOSE:**

This project is used for the proper monitoring if water supply to the crops at

times of altering weather conditions. So as to provide the required amount of water

to the crop based on temperature, humidity and the moisture level of the soil.

**2. LITERATURE SURVEY :**

**2.1 EXISTING PROBLEM :**

The major operating parameters which influence the smart irrigation system performance are:

1) Inadequate trenching

2) Lack of head-to-head design

3) Incorrect pipe sizes used

4) Poor quality cabling

**2.2 PROPOSED SOLUTION :**

This Project examines and compares some IOT regression methods.

• Wireless sensor network makes the irrigation smarter and easier. The entire

field is monitored continuously through the sensor network.

• The soil moisture levels are transferred from routers to the main coordinator.

• At the main controller the temperature is monitored, based on the

temperature and soil moisture content the motor is made to ON/OFF.

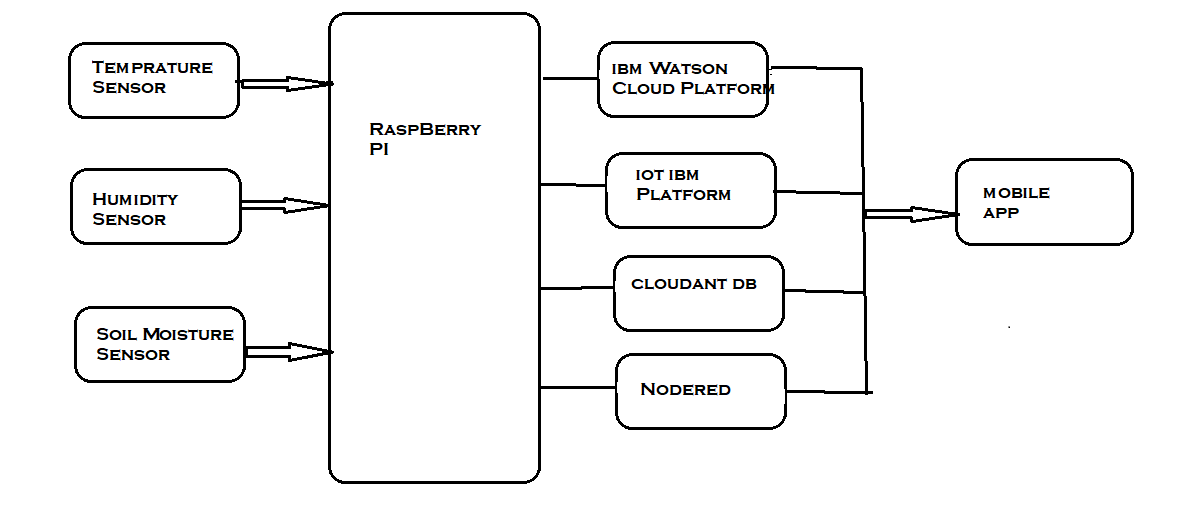
• Temperature, Humidity and Soil Moisture is visualized in the mobile app

and also stored in the database.

• Motor can be turned ON/OFF using the mobile app.

**3.THEORITICAL ANALYSIS :**

**3.1 BLOCK DIAGRAM :**



**Fig:Block diagram of smart irrigation system using iot**

**3.2 HARDWARE /SOFTWARE DESIGNING :**

**SOFTWARE DESIGNING :**

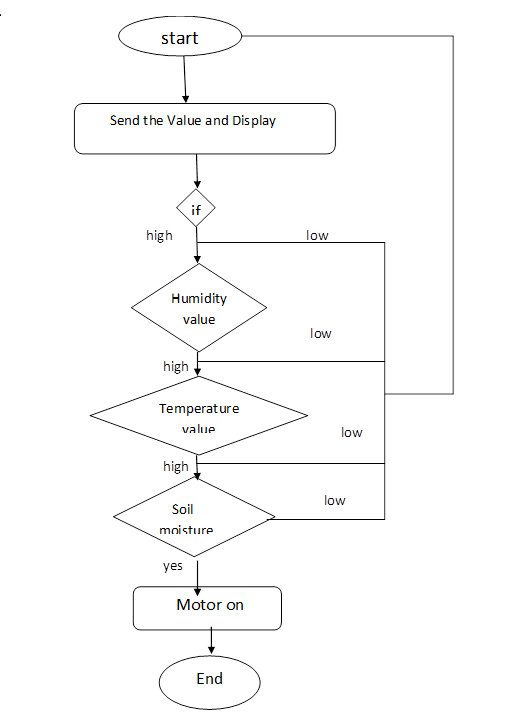
By using

1. Python
2. IBM Cloud account
3. IBM IOT Platform
4. Nodered
5. MIT App Inventor

**4. EXPERIMENTAL INVESTIGATION:**

The term “IoT” stands for the internet of things, can be defined as the interconnection between the individually identifiable embedded computing apparatus in the accessible internet infrastructure. The ‘IoT’ connects various devices and transportations with an help of internet as well as electronic sensors. Please refer to this link to know more about Experts Opinion on Application of Internet of Things (IoT) in Future.The Smart Irrigation System is an IoT based device which is capable of automating the irrigation process by analyzing the moisture of soil and the climate condition (like raining).Also the data of sensors will be displayed in graphical form on IBM cloud page

**5.FLOWCHART :**



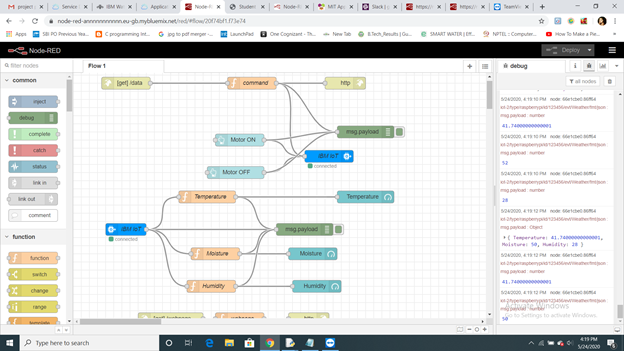
**6.RESULT:**

This system also allows controlling the amount of water delivered to the plants when it is needed based on types of plants by monitoring soil moisture and temperature.

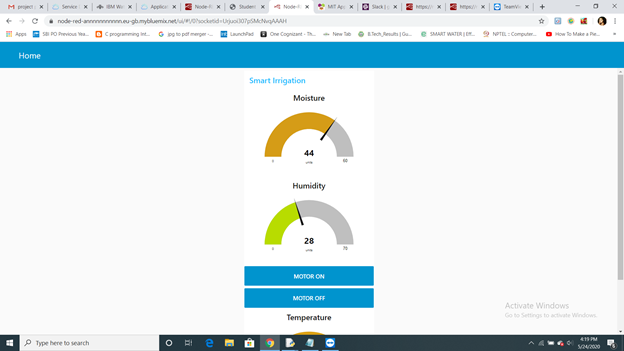
**Python Code Output:**

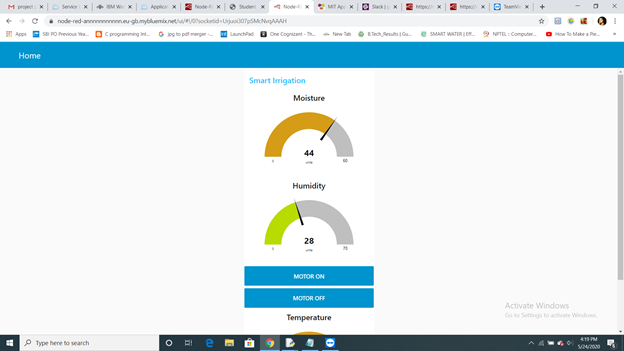


**NodeRed:**

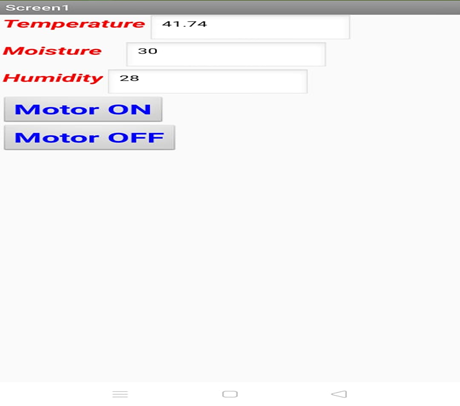


**NodeRed UI:**





**MIT APP:**



**7.ADVANTAGES AND DISADVANTAGES:**

**Advantages:**

1. Ability to save water
2. precision watering
3. Irrigating crops as per schedule
4. Decreasing water overflows
5. Ascertaining precise soil dampness levels

**Disadvantages:**

1. There could be wrong Analysis of Weather Conditions
2. Indian farmers are not used to these high-end technologies
3. Difficult in case of failure og GSM modem

**8.APPLICATIONS:**

1. It can be implemented in the modern irrigation system
2. It may also be used in rural areas
3. May be implemented by small agriculturists

**9.CONCLUSION:**

A system to monitor moisture levels in the soil was designed and the project provided an opportunity to study the existing systems, along with their features and drawbacks. The proposed system can be used to switch on/off the water sprinkler according to soil moisture levels thereby automating the process of irrigation which is one of the most time consuming activities in farming. Agriculture is one of the most water-consuming activities. The system uses information from soil moisture sensors to irrigate soil which helps to prevent over irrigation or under irrigation of soil thereby avoiding crop damage. The farm owner can monitor the process online through a website. Through this project it can be concluded that there can be considerable development in farming with the use of IOT and automation. Thus, the system is a potential solution to the problems faced in the existing manual and cumbersome process of irrigation by enabling efficient utilization of water resources.

**10.FUTURE SCOPE:**

To improve the efficiency and effectiveness of the system, the following recommendations can be put into consideration. Option of controlling the water pump can be given to the farmer i.e. he can switch on/off the pump in order to start/stop the process of irrigation without being present at the farm. The farmer may choose to stop the growth of crops or the crops may get damaged due to adverse weather conditions. In such cases farmer may need to stop the system remotely. The idea of using IOT for irrigation can be extended further to other activities in farming such as cattle management, fire detection and climate control. This would minimalize human intervention in farming activities. .

**11.BIBLIOGRAPHY :**

<https://cloud.ibm.com/registration>

<https://cloud.ibm.com/catalog/services/watson-studio>

<https://node-red-annnnnnnnnnn.eu-gb.mybluemix.net/red/>

<http://ai2.appinventor.mit.edu/>

**APPENDIX :**

**A.SOURCE CODE**

import requests

import sys

import time

import ibmiotf.application

import ibmiotf.device

import random

r=requests.get('http://api.openweathermap.org/data/2.5/weather?q=Guntur,IN&appid=42a67b9e8ecd9620c2fe1471361c3e53')

#Provide your IBM Watson Device Credentials

organization = "w1gnzn"

deviceType = "raspberrypi"

deviceId = "123456"

authMethod = "token"

authToken = "123456789"

def myCommandCallback(cmd):

print("Command received: %s" % cmd.data['command'])

if cmd.data['command']=='motoron':

print("Motor is ON")

elif cmd.data['command']=='motoroff':

print("Motor is OFF")

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()

#print("response is")

#print(r.json())

#for i in r.json():

#print(i)

#print(r.json()["main"])

#print("temparature value:")

#print(r.json()["main"]["temp"])

while True:

print("humidity value:")

print(r.json()["main"]["humidity"])

hum=r.json()["main"]["humidity"]

temk=r.json()["main"]["temp"]

#print("temperature in kelvin is:",temk)

temperature=temk-272.15

print("temperature in celcius is:",temperature)

mois=random.randrange(20,60,2)

print("moisture level of soil is:",mois)

if(temperature>32 | mois<35):

req\_sms=requests.get('https://www.fast2sms.com/dev/bulk?authorization=TPnud1eh5Bfyt2FpHoWXGwlC7NSsKYLmIz6MEvRi8a93jgAZbDDvuxwEg9eBdjmP7OLRpJ2MsIhoZ54a&sender\_id=FSTSMS&message=temp%20high&language=english&route=p&numbers=7075001212,9121852344,')

data = { 'Temperature' : temperature, 'Moisture': mois, 'Humidity': hum }

#print (data)

def myOnPublishCallback():

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

success = deviceCli.publishEvent("Weather", "json", data, qos=0, on\_publish=myOnPublishCallback)

if not success:

print("Not connected to IoTF")

time.sleep(2)

deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud

deviceCli.disconnect()