**A Project Report**

**On**

**Smart Agriculture System based on IOT**

**BY**

**S.Sahana**

**Intern Of**

**SMART BRIDGE**

ACKNOWLEDGEMENT

I an intern Of Smart Bridge has successfully completed the

project under the guidance of my Mentor of this institute.

I would like to thank my Mentors for their valuable guidance and advice for making me to finish this Project and for providing assistance.They are extremely motivating and helping me for Project work by giving Mentor Session and for Providing informations and Links.

**ABSTRACT**

Agriculture is the broadest economic sector and plays an important role in the overall economic development of a nation. Technological advancements in the arena of agriculture will ascertain to increase the competence of certain farming activities. In this paper, we have proposed a novel methodology for smart agriculture by linking a smart sensing system through wireless communication technology. Our system focuses on the measurement of physical parameters such as soil moisture content, humidity and temperature that plays a vital role in farming activities. Based on the essential physical and chemical parameters of the soil measured, the required quantity of green manure, compost, and water is splashed on the crops using a smart irrigator, which is mounted on a movable overhead crane system. The detailed demonstaration of sensor and Weather monitoring of the area and controlling the motor are demonstrated in this paper.

TABLE OF CONTENTS

1. CHAPTER 1:**INTRODUCTION**

1.1 Overview

1.2 Purpose

2. CHAPTER 2:**LITERATURE SURVEY**

2.1 Existing Problem

2.2 Proposed Solution

3. CHAPTER 3:**THEORITICAL ANALYSIS**

3.1 Block Diagram

3.2 Hardware/Software Designing

4. CHAPTER 4:**EXPERIMENTAL INVESTIGATION**

5.CHAPTER 5:**FLOWCHART**

6. CHAPTER 6:**RESULT**

7. CHAPTER 7:**ADVANTAGES AND DISADVANTAGES**

8.CHAPTER 8:**APPLICATIONS**

9. CHAPTER 9:**CONCLUSION**

10.CHAPTER 10:**FUTURE SCOPE**

11. CHAPTER 11:**BIBLIOGRAPHY**

**APPENDIX**

A Source code

1. CHAPTER 1: **INTRODUCTION**

1.1 **Overview**

The objectives of this report is to proposed IoT based Smart Agriculture System which will enable farmers to have live data of soil moisture environment temperature at very low cost so that live monitoring can be done.

**1.2 Purpose**

Agriculture is the root to country’s economic development. In recent times, huge scientific advancement has been implemented in various agricultural fields for the betterment of the future. Despite of various researches, proper assessment and productivity couldn’t be reached. We have tried to focus on different scientific applications which could be put together in agricultural field for better accuracy with better productivity using less man-power. Moreover we include a method for monitoring the agricultural fields from any remote location and assess the basic condition of the field.Smart Agriculture System based on IoT can monitor soil moisture and climatic conditions to grow and yield a good crop.The farmer can also get the realtime weather forecasting data by using external platforms like Open Weather API.Farmer is provided a mobile app using which he can monitor the temperature,humidity and soil moisture parameters along with weather forecasting details.Based on all the parameters he can water his crop by controlling the motors using the mobile application.Even if the farmer is not present near his crop he can water his crop by controlling the motors using the mobile application from anywhere.Here we are using the Online IoT simulator for getting the Temperature,Humidity and Soil Moisture values.

2. CHAPTER 2:**LITERATURE SURVEY**

**2.1 Existing Problem**

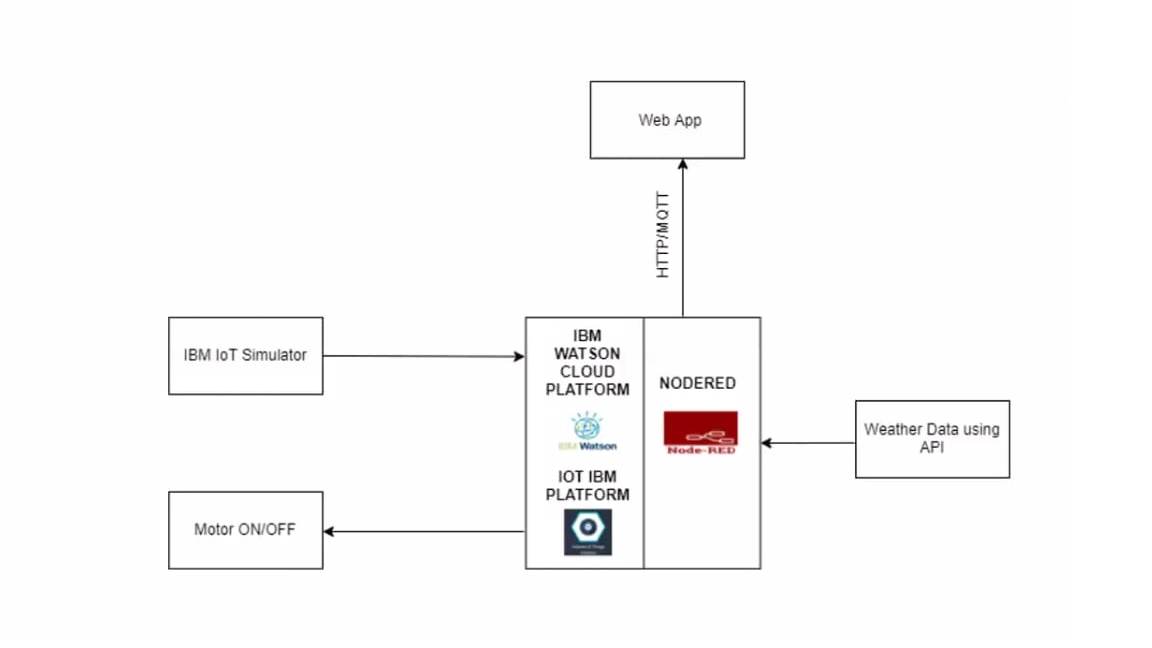
Farmers had to go to the land and check for cultivation,Temperature,Humidty and Soil moisture and to act based on the Weather conditions like high temperature,pressure etc.They had to go to the area and check for irrigation and control the motors even when they are ill or at bad climatic conditions.They need someting to lease their work andthe crops also must be cultivated properly.

**2.2 Proposed Solution**

Smart Agriculture System based on IoT can monitor soil moisture and climatic conditions to grow and yield a good crop.The farmer can also get the realtime weather forecasting data by using external platforms like Open Weather API.Farmer is provided a mobile app using which he can monitor the temperature,humidity and soil moisture parameters along with weather forecasting details.Based on all the parameters he can water his crop by controlling the motors using the mobile application.Even if the farmer is not present near his crop he can water his crop by controlling the motors using the mobile application from anywhere.Here we are using the Online IoT simulator for getting the Temperature,Humidity and Soil Moisture values.

3. CHAPTER 3:**THEORITICAL ANALYSIS**

**3.1 Block Diagram**



4. CHAPTER 4:**EXPERIMENTAL INVESTIGATION**

**IBM CLOUD**

**IBM Cloud** is a set of [cloud computing](https://en.wikipedia.org/wiki/Cloud_computing) services for business offered by the information technology company [IBM](https://en.wikipedia.org/wiki/IBM).[[1]](https://en.wikipedia.org/wiki/IBM_Cloud#cite_note-1) It combines [platform as a service](https://en.wikipedia.org/wiki/Platform_as_a_service) (PaaS) with [infrastructure as a service](https://en.wikipedia.org/wiki/Infrastructure_as_a_service) (IaaS). The platform scales and supports both small development teams and organizations, and large enterprise businesses. It is globally deployed across [data centers](https://en.wikipedia.org/wiki/Data_centers) around the world.[[2]](https://en.wikipedia.org/wiki/IBM_Cloud#cite_note-2) IBM's main competitors in the cloud computing market include [Amazon Web Services](https://en.wikipedia.org/wiki/Amazon_Web_Services), [Microsoft Azure](https://en.wikipedia.org/wiki/Microsoft_Azure) and [Google Cloud Platform](https://en.wikipedia.org/wiki/Google_Cloud_Platform).[[](https://en.wikipedia.org/wiki/IBM_Cloud#cite_note-3)

**IBM IOT Platform**

IBM Watson IoT Platform is a managed, cloud-hosted service designed to make it simple to derive value from your IoT devices. Watson IoT Platform and its additional add on services - Blockchain service and analytic service - enable organizations to capture and explore data for devices, equipment, and machines, and discover insights that can drive better decision-making.

**Node-red**

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.

Node-RED provides a browser-based flow editor that makes it easy to wire together flows using the wide range of nodes in the palette. Flows can be then deployed to the runtime in a single-click.

JavaScript functions can be created within the editor using a rich text editor.

A built-in library allows you to save useful functions, templates or flows for re-use.

he flows created in Node-RED are stored using JSON which can be easily imported and exported for sharing with others.

An online flow library allows you to share your best flows with the world.

The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud.

With over 225,000 modules in Node's package repository, it is easy to extend the range of palette nodes to add new capabilities.

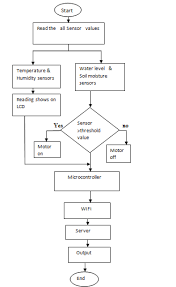
**PYTHON**

**Python** is an [interpreted](https://en.wikipedia.org/wiki/Interpreted_language), [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [general-purpose](https://en.wikipedia.org/wiki/General-purpose_programming_language) [programming language](https://en.wikipedia.org/wiki/Programming_language). Created by [Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum) and first released in 1991, Python's design philosophy emphasizes [code readability](https://en.wikipedia.org/wiki/Code_readability) with its notable use of [significant whitespace](https://en.wikipedia.org/wiki/Off-side_rule). Its [language constructs](https://en.wikipedia.org/wiki/Language_construct) and [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) approach aim to help programmers write clear, logical code for small and large-scale projects

Python is [dynamically typed](https://en.wikipedia.org/wiki/Dynamic_programming_language) and [garbage-collected](https://en.wikipedia.org/wiki/Garbage_collection_(computer_science)). It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigms), including [structured](https://en.wikipedia.org/wiki/Structured_programming) (particularly, [procedural](https://en.wikipedia.org/wiki/Procedural_programming)), object-oriented, and [functional programming](https://en.wikipedia.org/wiki/Functional_programming). Python is often described as a "batteries included" language due to its comprehensive [standard library](https://en.wikipedia.org/wiki/Standard_library)

Python [interpreters](https://en.wikipedia.org/wiki/Interpreter_(computing)) are available for many [operating systems](https://en.wikipedia.org/wiki/Operating_system). A global community of programmers develops and maintains [CPython](https://en.wikipedia.org/wiki/CPython), an [open source](https://en.wikipedia.org/wiki/Open-source_software) [reference implementation](https://en.wikipedia.org/wiki/Reference_implementation). A [non-profit organization](https://en.wikipedia.org/wiki/Nonprofit_organization), the [Python Software Foundation](https://en.wikipedia.org/wiki/Python_Software_Foundation), manages and directs resources for Python and CPython development.

5. CHAPTER 5:**FLOW CHART**



6. CHAPTER 6:**RESULT**

The global population is predicted to touch [9.6 billion by 2050](http://www.computerweekly.com/news/2240239484/IoT-could-be-key-to-farming-says-Beecham-Research) – this poses a big problem for the agriculture industry. Despite combating challenges like extreme weather conditions, rising climate change, and farming’s environmental impact, the demand for more food has to be met. To meet these increasing needs, agriculture has to turn to new technology.

7. CHAPTER 7:**ADVANTAGE AND DISADVANTAGE**

**Advantage**

* User Friendly.
* Remote Monitoring.
* Increased productivity.
* Enhanced safety.
* Easier agriculture procedures.
* Instant interventions around the clock.
* Advanced lifestyle.

**Disadvantage**

* May give variable accuracy.
* Little variation in sensors reading when exposed to different environments.
* Implementation on large scale strenuous.

8. CHAPTER 8:**APPLICATIONS**

* By finding suitable environmental conditions can accelerate farming.
* Site specific data such as soil moisture and yield can be collected comprehensively.
* Digitized and efficient farming practice.

9. CHAPTER 9:**CONCLUSION**

The existing system works in a manner in which it firstly does data collection from the farm via the help of sensors, then it sends the data to the server side from where and on which further actions can be taken. The final output of this system is displayed after getting processed by the server side and then displayed onto the mobile phone. Thus, on the basis of literature survey and by analyzing the existing system, we have come to a conclusion that the proposed system will not only aid the farmers but will also help them to digitize their farming practice and in turn help them to yield the best from that soil without being dependent on the climatic conditions.

10. CHAPTER 10:**FUTURE SCOPE**

Future work would be focused more on increasing sensors on this system to fetch more data especially with regard to Pest Control and by also integrating GPS module in this system to enhance this Agriculture IoT Technology to full-fledged Agriculture Precision ready

product.

11. CHAPTER 11:**BIBLIOGRAPHY**

**APPENDIX A**

**SOURCE CODE**

import time

import sys

import ibmiotf.application # to install pip install ibmiotf

import ibmiotf.device

#Provide your IBM Watson Device Credentials

organization = "3mfyqe" #replace the ORG ID

deviceType = "NodeMCU"#replace the Device type with you device

deviceId = "Node12"#replace Device ID

authMethod = "token"

authToken = "123456789" #Replace the authtoken

def myCommandCallback(cmd): # function for Callback

print("Command received: %s" % cmd.data)

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

print("MOTOR IS ON")

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

print("MOTOR is oFF")

if cmd.command == "setInterval":

if 'interval' not in cmd.data:

print("Error - command is missing required information: 'interval'")

else:

interval = cmd.data['interval']

elif cmd.command == "print":

if 'message' not in cmd.data:

print("Error - command is missing required information: 'message'")

else:

output=cmd.data['message']

print(output)

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:

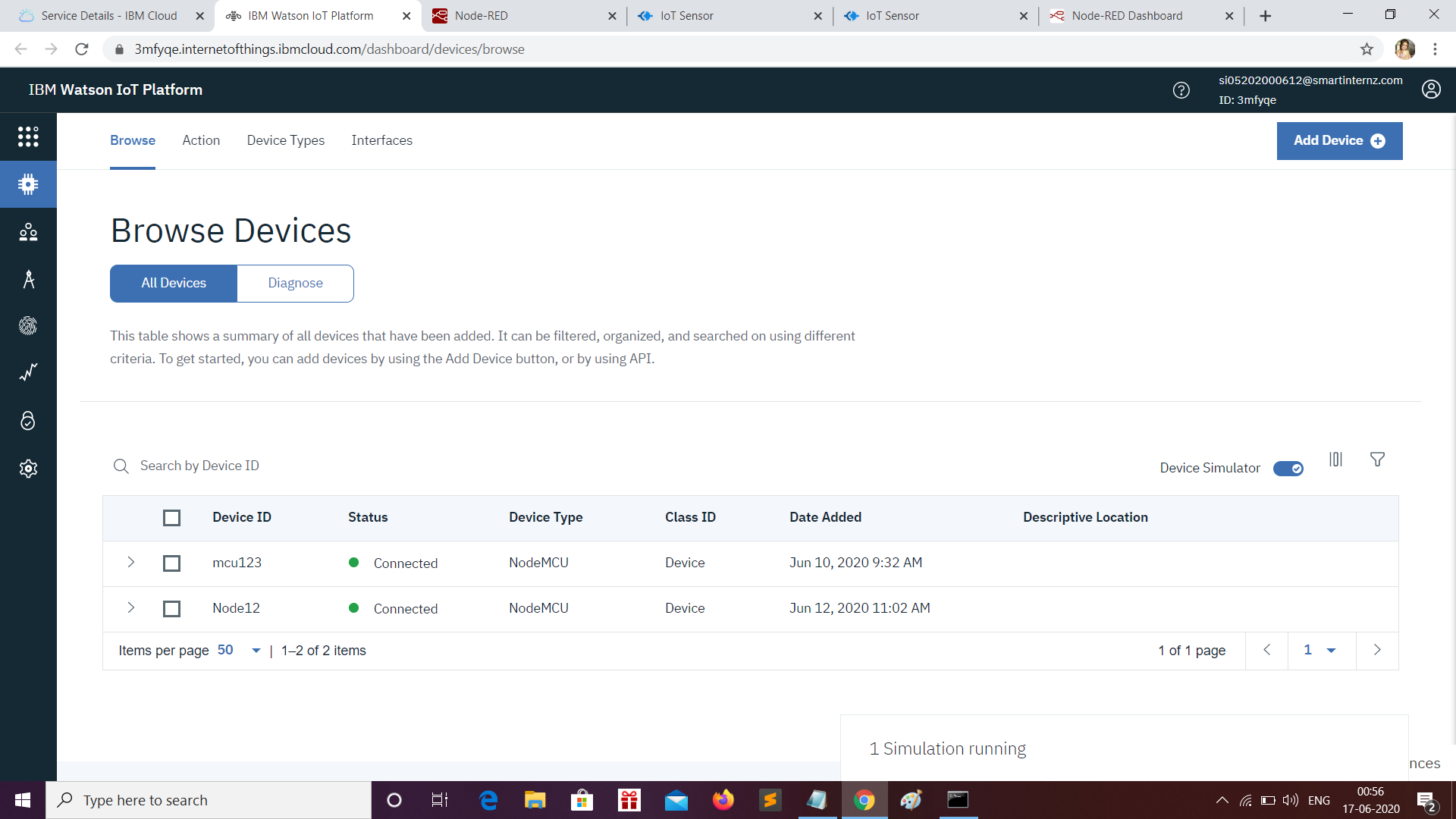
deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud

deviceCli.disconnect()

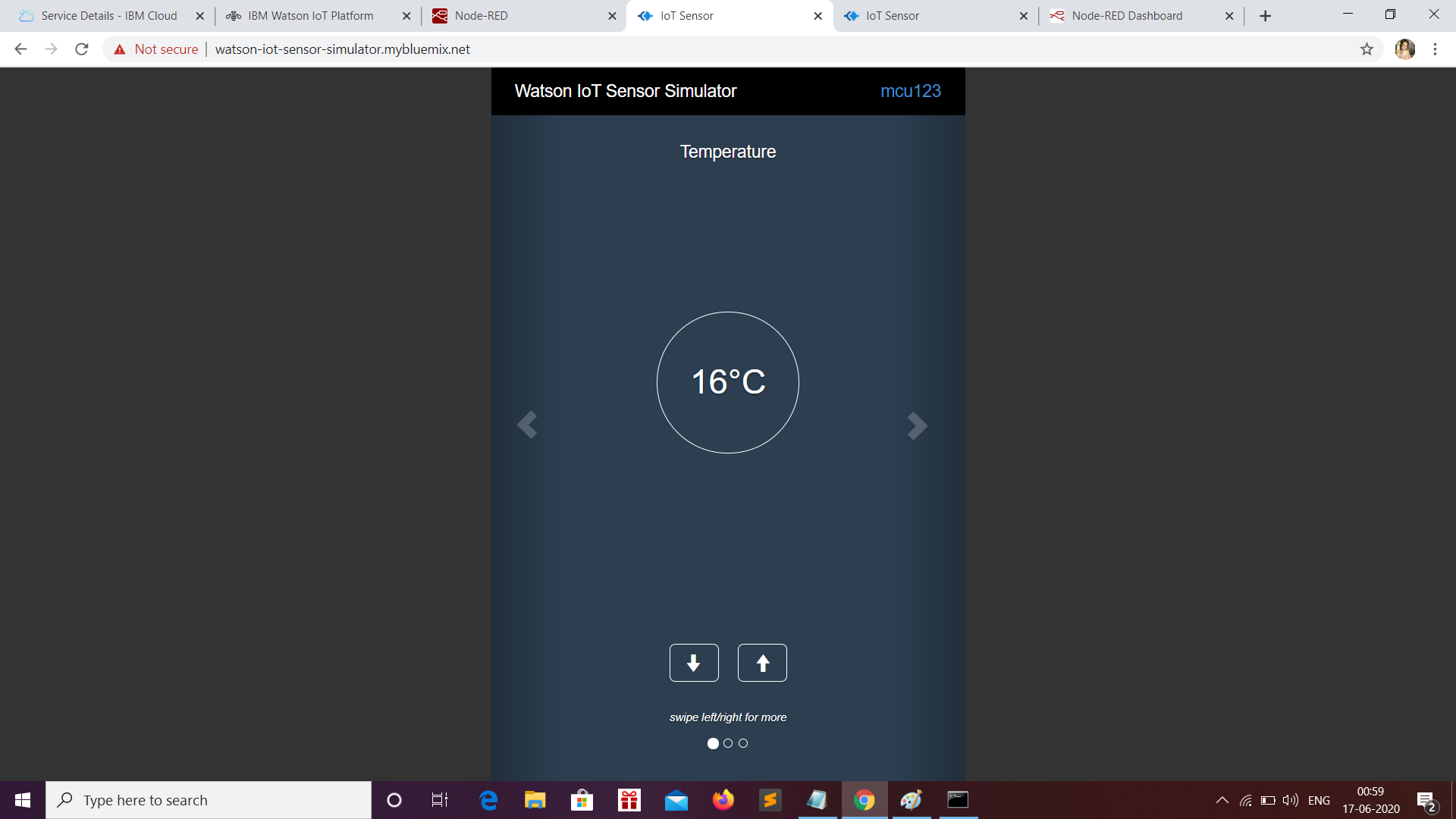
**OUTPUT**

Watson IBM IOT Platform

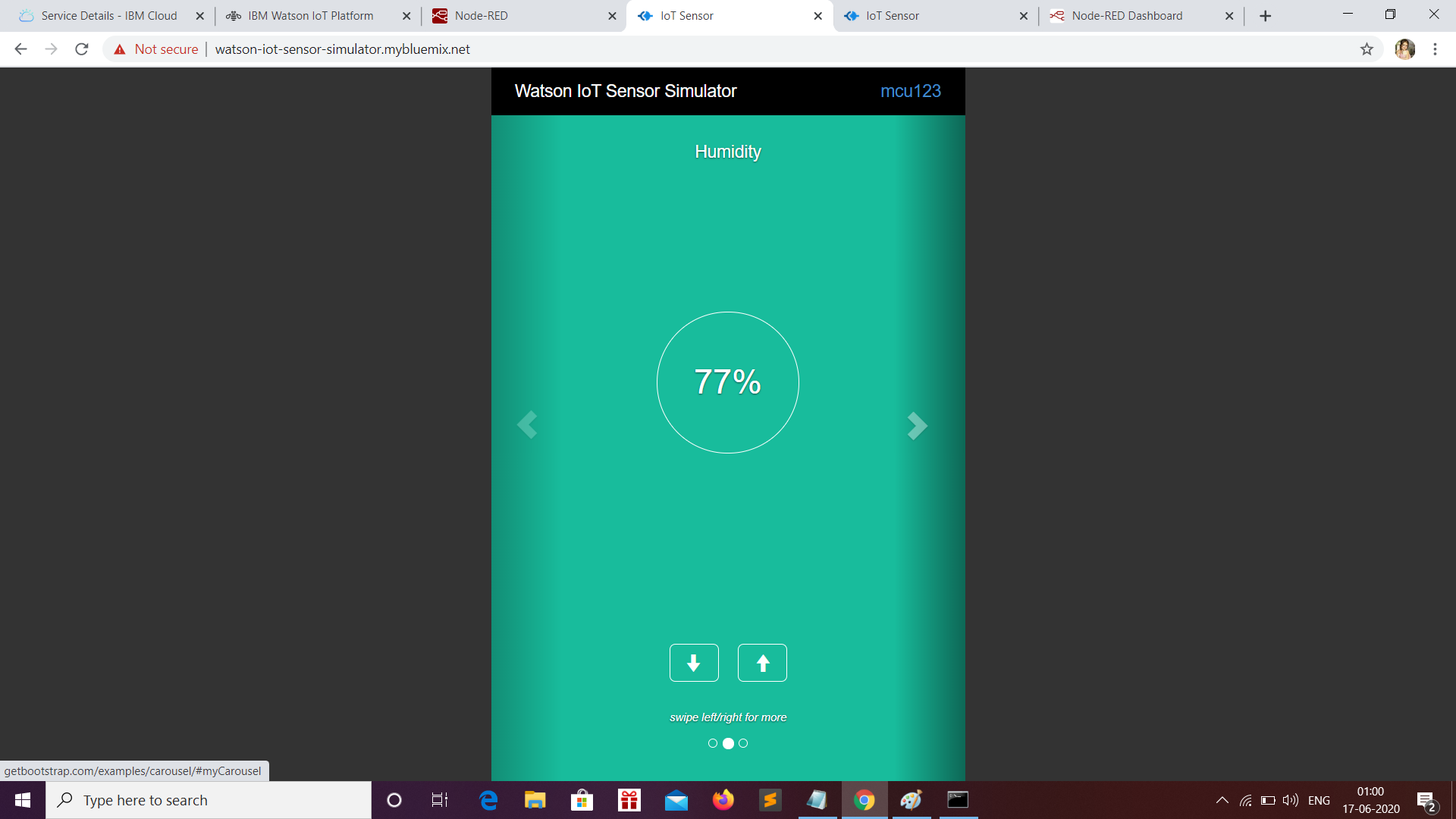


Watson IOT Sensor Simulator

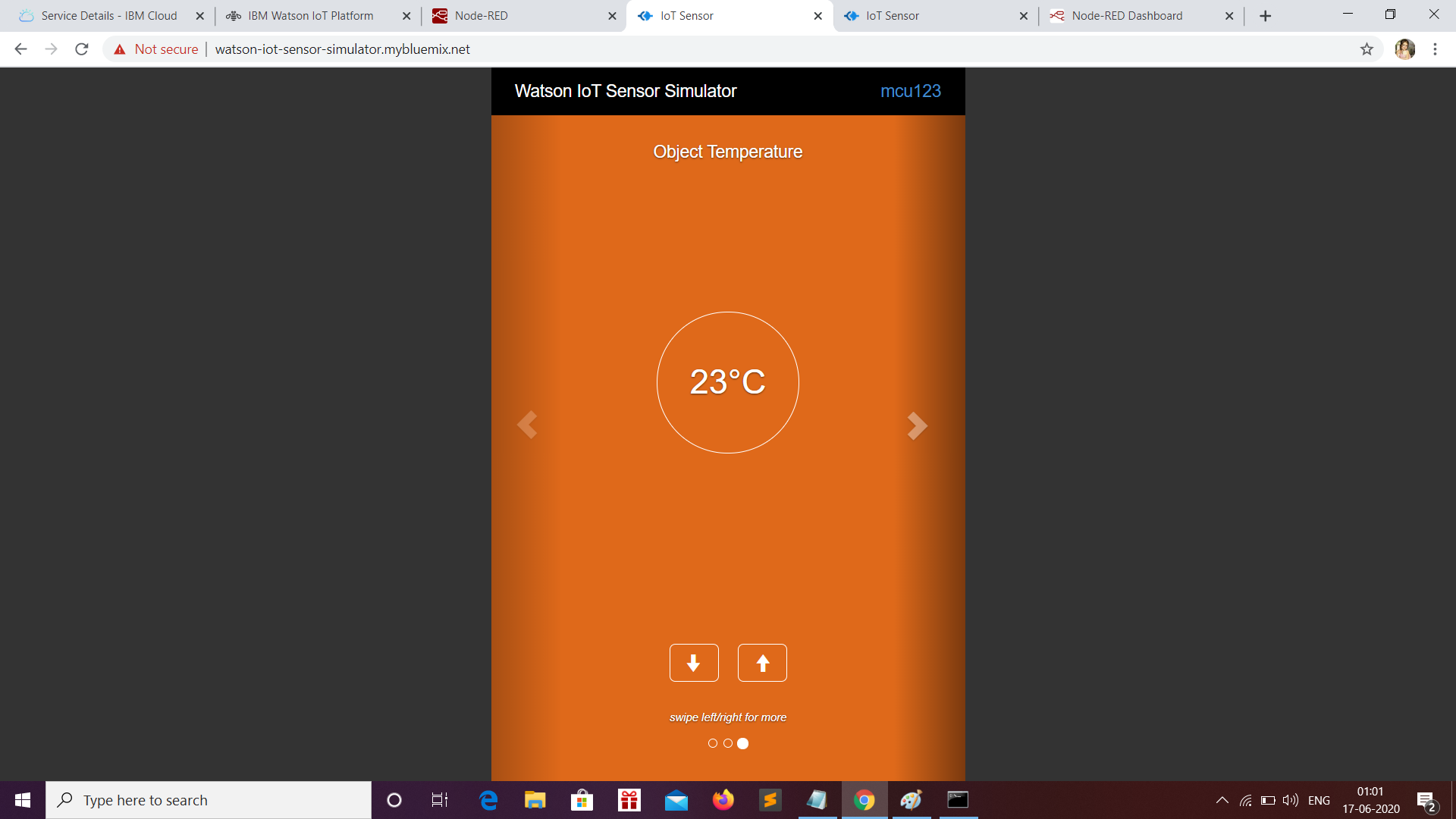
**Temperature**



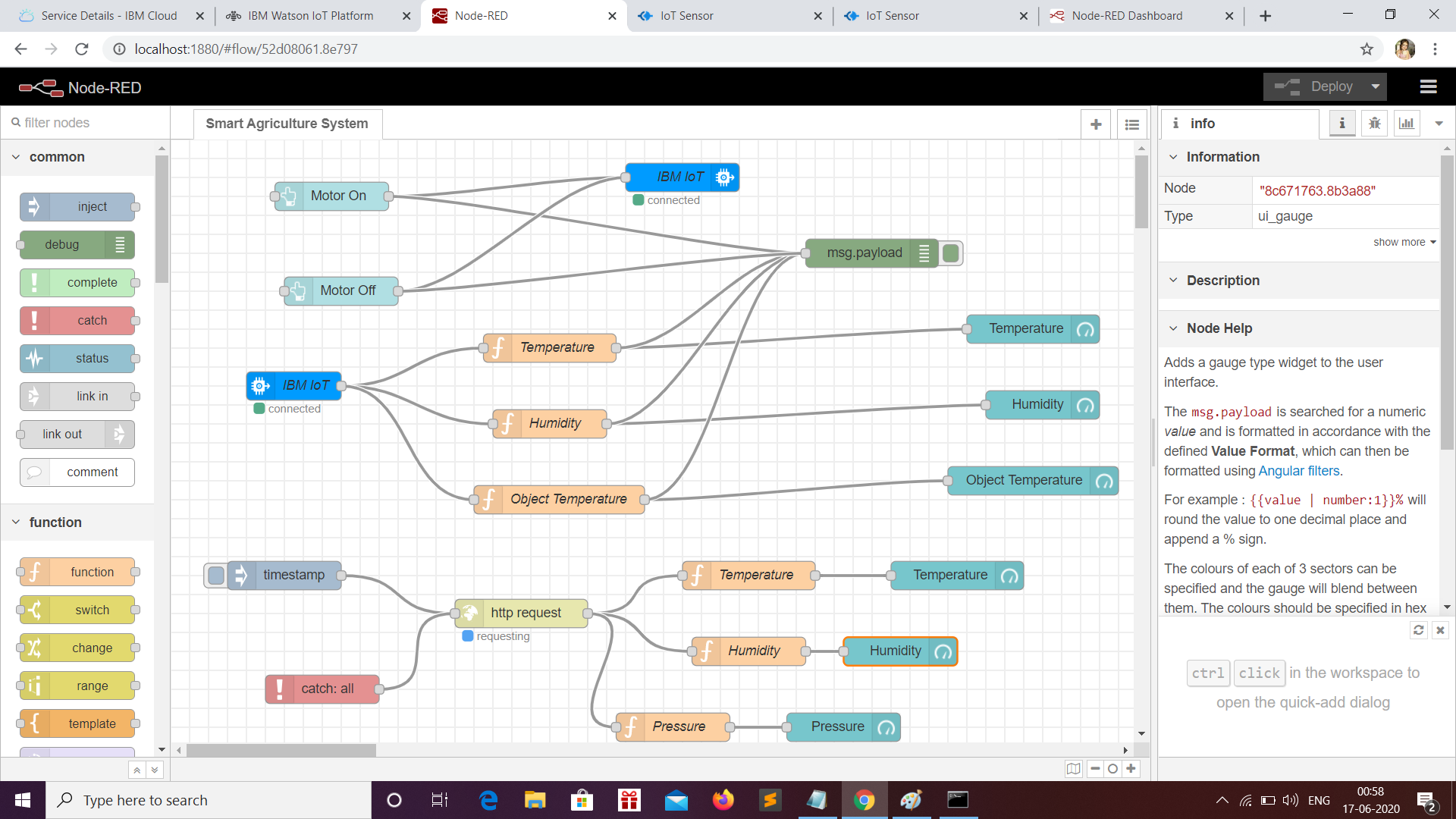
Humidity



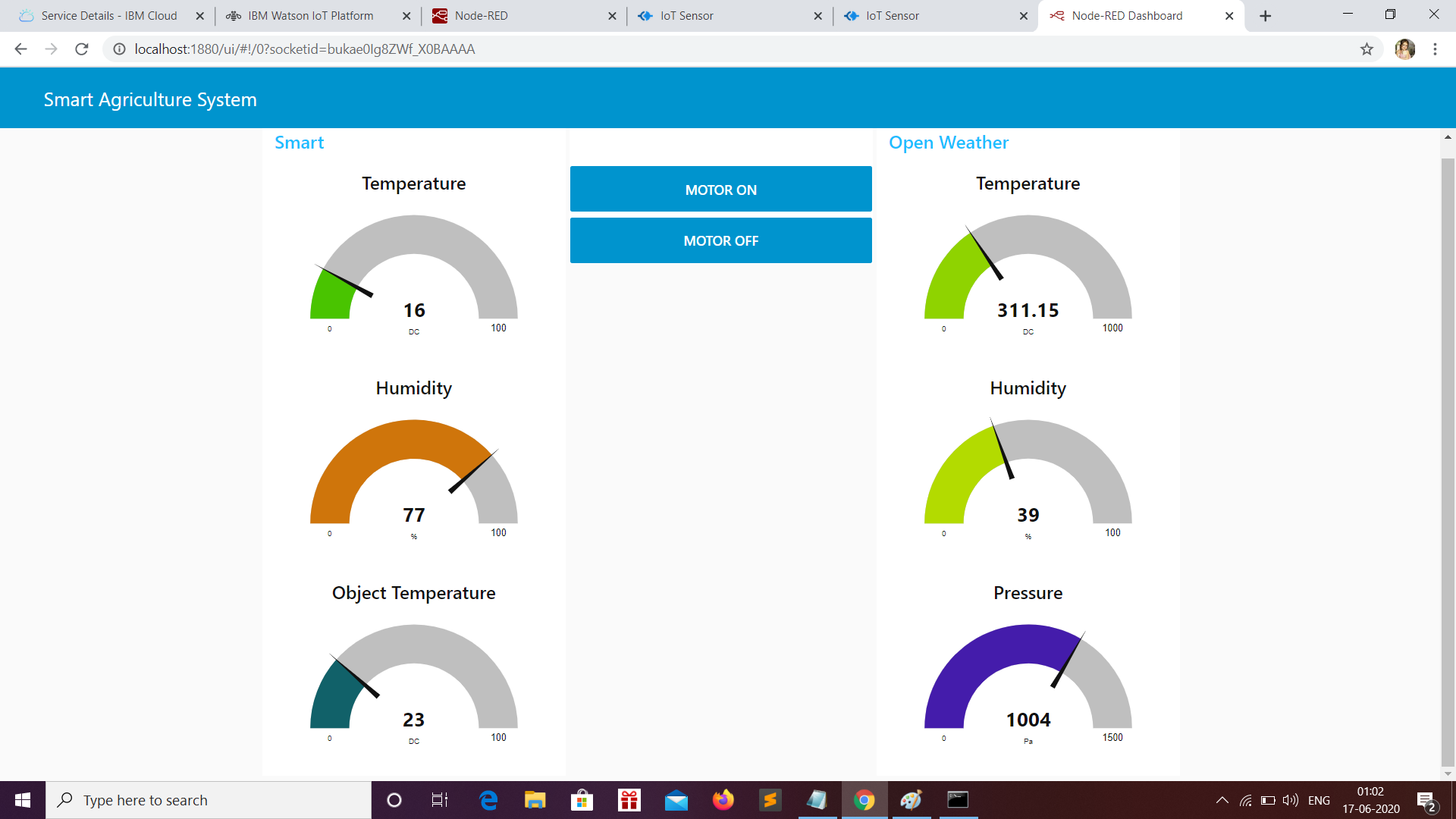
Object Temperature



Node-red



UI



PYTHON OUTPUT

