IoT Based Health Monitoring System And Analysis

Summer Internship Project - 2020

Submitted By -

Utkarsh Maheria

Khyama Chandra Shekhar

Shankar Prasad Talam

1. INTRODUCTION

1.1 OVERVIEW

While technology can't stop the population from ageing and eradicate chronic diseases at once, it can at least make healthcare services easier to avail and in terms of accessibility it can be improved through a smartphone application to which a medical device can be connected which can collect medical and other required health data by using data connection of smartphone, mostly the internet to transfer the collected information to the doctors through a cloud database.

1.2 PURPOSE

With the use of such technology based healthcare methods there are numerous benefit which could improve the quality and efficiency of treatments and improve the health of patients.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

Nowadays, healthcare services are costlier than ever, global population is ageing and number of chronic diseases are on all time high. Everyone is needs of basic health care amenities and a service which ensures that it reaches most of the people.

2.2 PROPOSED SOLUTION

The right way to diagnose a patient will also lessen the need of hospitalization when the issue is not serious enough. Through Internet of Things (IoT) we have an extensive applicability in numerous areas including healthcare. So, we aim to develop a system which consists of a smartphone application to monitor a person's health conditions.

3. THEORITICAL ANALYSIS

Software and Services Used:

1. IBM Cloud
1. IBM IoT Platform – For connecting devices to the cloud through internet.
2. Cloudant DB – For storing health data & person's information.
3. Cloud Object Storage – For ML model.
4. Watson Studio (ML Model) – For deploying the ML model and classifying the dataset based upon appropriate algorithm to generate accurate results.
5. NodeRed Editor – For connecting and sending data from devices to IoT platform to the MIT app inventor and vice versa.
2. MIT App Inventor – To design the mobile application necessary for filling out person's details and obtaining the status
3. Python IDE – For connecting sensors to the IBM cloud and sending values.

Hardware Used:

1. Temperature Sensor
2. Pulse rate sensor

3. Blood pressure sensor

4. FLOWCHART

The flowchart presents the working of the system and gives an overview about how different processes will occur simultaneously one after another and help in obtaining the desired results in order to provide accurate data for the next step so as to achieve the outcomes associated with the system.
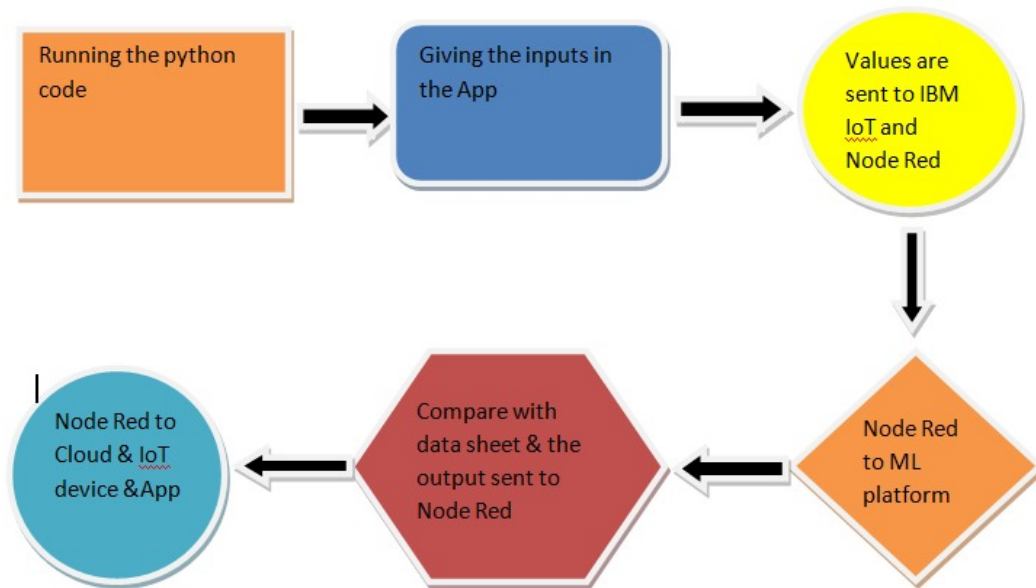
5. RESULT

A smartphone application is created through which patient's health condition is stated by considering the values of pulse, blood pressure, systolic and diastolic as well as the age.

These details along with name and gender are entered through app as well as through the device. By clicking the submit button, the user can obtain its status.

Save Changes   Cancel   ⬆ Upload Attachment   ⟳ Clone Document   🗑 Delete

```
1  {
2     "_id": "26a144c8e2353c7909422773558c9060",
3     "_rev": "1-4835a2c72ed2a37eb7d0d92d972f7d90",
4     "topic": "",
5     "payload": {
6        "name": "Utkarsh",
7        "age": "45",
8        "gender": "male",
9        "Temperature": 35.8,
10       "Systolic": 83,
11       "Diastolic": 56,
12       "Pulse": 148,
13       "output": "Critical"
14    }
15 }
```

6. ADVANTAGES & DISADVANTAGES

ADVANTAGES

1. Simultaneous reporting and monitoring of health condition.

2. End-to-end connectivity and affordability.

3. Tracking and alerts.

4. Fast and on the spot analysis.

DISADVANTAGES

1. Data overloading might occur at times.

2. Inaccuracy in results at times.

7. APPLICATIONS

1. Senior citizens living alone or living with 1 or 2 members can be benefited. This project really proves helpful when family members need to go out for some emergency work

2. Disabled patients who find it really difficult to go to doctors on a daily basis or for those patients who need continuous monitoring from the doctor.

3. One can carry the device to anywhere they want to and test themselves

from any location they want to, given that they have an internet connection.

8. CONCLUSION

Internet of Things changes the way facilities are delivered to healthcare industry. These technologies improve the product causing a greater impact by integrating together a multi model system ranging from hardware to software.

IOT platform provides a simple environment to connect the hardware devices through the cloud and users. Then by using IOT we can make any type of devices by different type of sensors and modules.

1. This project is useful for Patient who required regular check the parameters.

2. The cloud data storage facility provides to the doctors to better treatment environment by this friendly device.

3. The simpler working and Mobile app GUI provides the friendly connection between the devices to users.

Finally we can say, this is a complete healthcare device.

9. FUTURE SCOPE

In the upcoming world this kind of technology will increase its mass adoption and healthcare centres across the globe will transform and create solutions to streamline operations.

1. For more reliable and good reach of the data much more efficient and secure protocols can be used.
2. Application can be increased by adding the more modules and sensors.
3. Accuracy can be increased by using high sensitive sensors and more reliable modules.

10. BIBLIOGRAPHY

1. www.wikipedia.org

2. https://cloud.ibm.com/

3. http://ai2.appinventor.mit.edu/#5369944233869312

4. https://thesmartbridge.com/documents/pdf/Deploy-a-ml-model-in-watson-studio.pdf

5. https://smartbridge.teachable.com/

APPENDIX

Code :

import time

import sys

import random

import ibmiotf.application

import ibmiotf.device

```python
#Provide your IBM Watson Device Credentials

organization = "n3z7ci" # repalce it with organization ID

deviceType = "project" #replace it with device type

deviceId = "1002" #repalce with device id

authMethod = "token"

authToken = "1234567890"#repalce with token


def myCommandCallback(cmd):

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

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

        print("Details Submitted ")


try:

   deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,
"auth-method": authMethod, "auth-token": authToken}

   deviceCli = ibmiotf.device.Client(deviceOptions)
```

```python
    #..........................................

except Exception as e:

    print("Caught exception connecting device: %s" % str(e))

    sys.exit()

deviceCli.connect()

while True:


    Temp =round(random.uniform(35,41) ,1)

    systolic =random.randint(60,162)

    diastolic=random.randint(40,102)

    pulse = random.randint(30,180)



        data = {'d':{'Temperature' : Temp, 'Systolic': systolic, 'Diastolic':diastolic,
'Pulse': pulse  }}
```

```python
    #print data

    def myOnPublishCallback():

        print ("Temperature = %s C" % Temp, "Systolic = %s " %
systolic,"Diastolic = %s " % diastolic, "Pulse = %s " % pulse)

        success = deviceCli.publishEvent("event", "json", data, qos=0,
on_publish=myOnPublishCallback)

        if not success:

            print("Not connected to IoTF")

        time.sleep(20)

        deviceCli.commandCallback = myCommandCallback


# Disconnect the device and application from the cloud

deviceCli.disconnect()
```
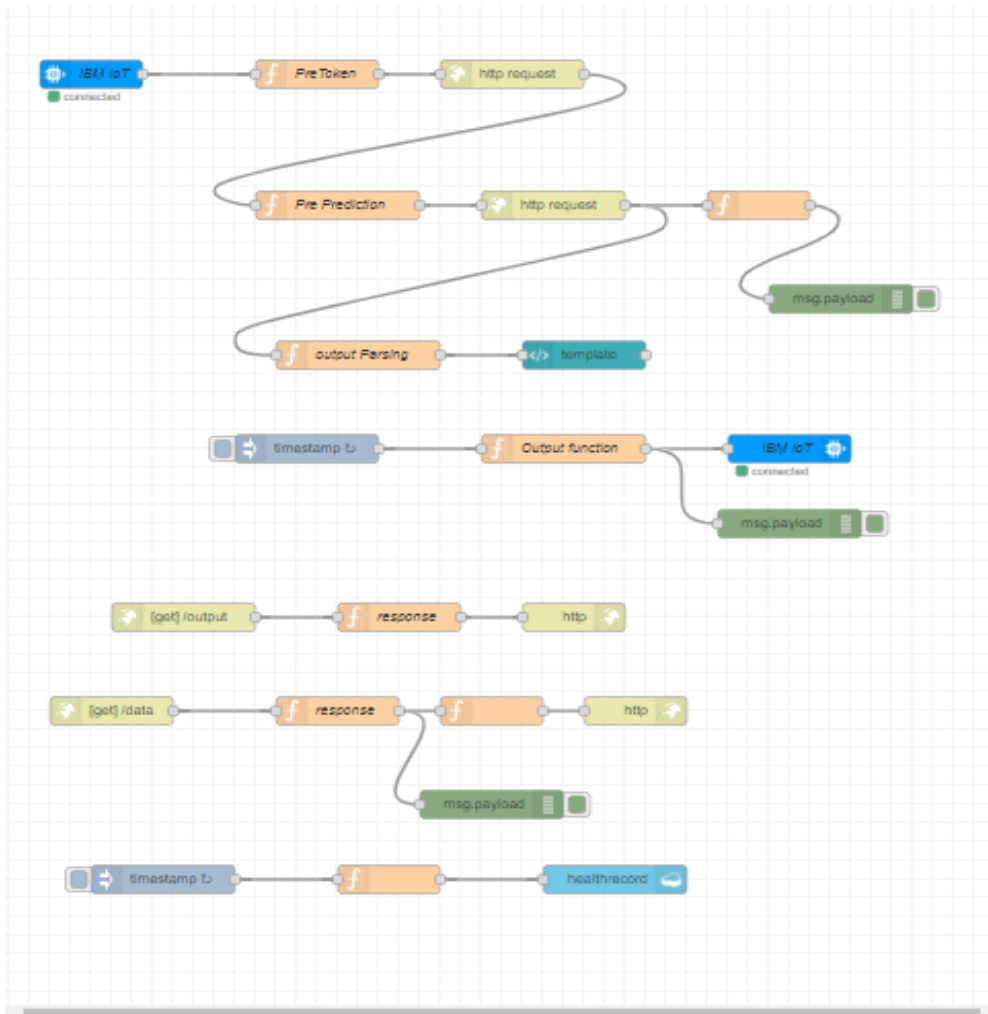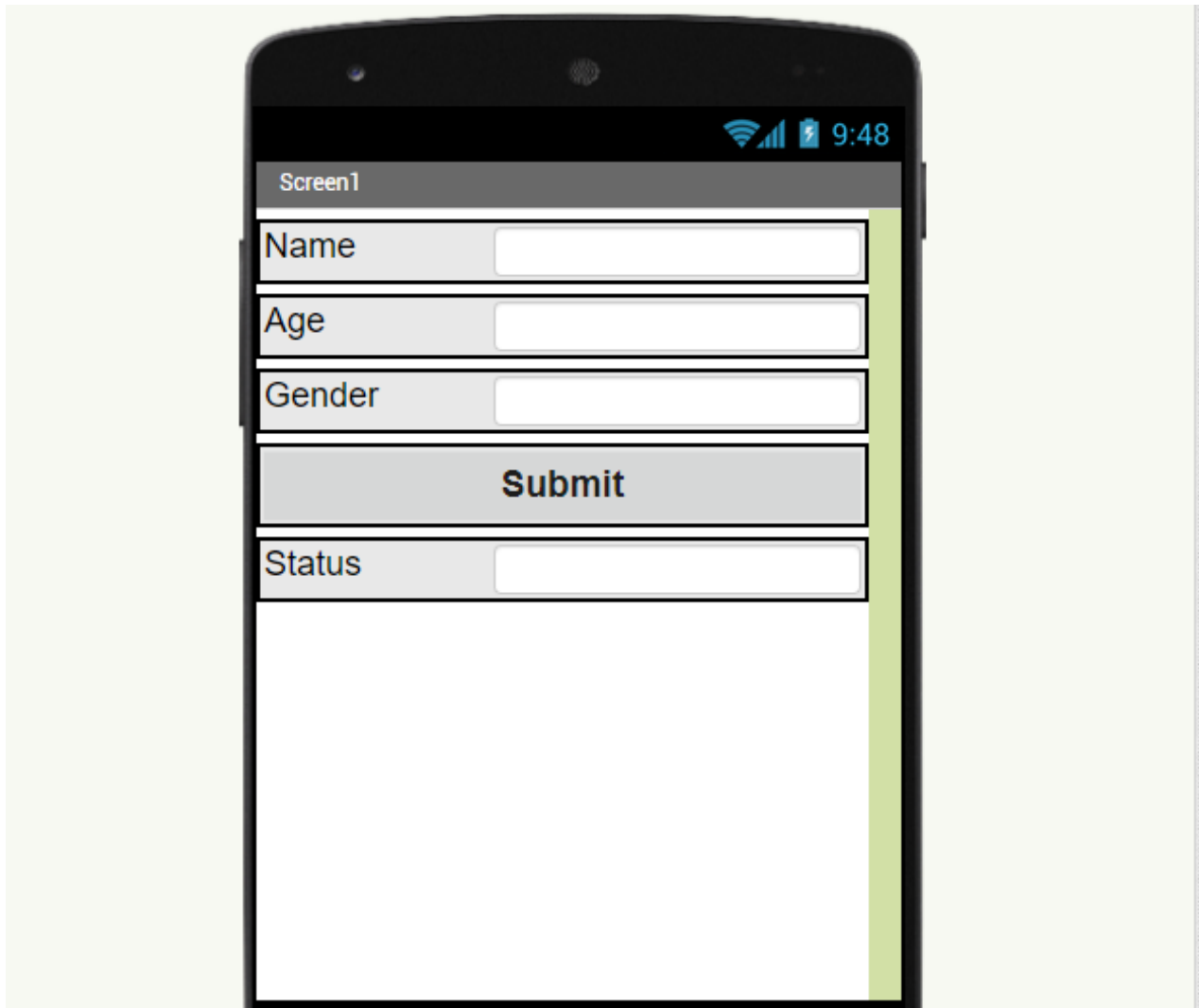
PHOTOS :

Node Red Flow -

MIT APP Front End

MIT APP Backend -

```
when  Clock2 ▾ .Timer
do    set  Web1 ▾ . Url ▾  to   " https://node-red-lbjbi.eu-gb.mybluemix.net/output "
      call  Web1 ▾ .Get


when  Button1 ▾ .Click
do    set  Web2 ▾ . Url ▾  to   ⊙ join  " https://node-red-lbjbi.eu-gb.mybluemix.net/data?... "
                                        TextBox2 ▾ . Text ▾
                                        " &name= "
                                        TextBox1 ▾ . Text ▾
                                        " &gender= "
                                        TextBox3 ▾ . Text ▾
      call  Web2 ▾ .Get


when  Web1 ▾ .GotText
      url   responseCode   responseType   responseContent
do    set  TextBox4 ▾ . Text ▾  to   look up in pairs  key   " Health condition "
                                                       pairs  call  Web1 ▾ .JsonTextDecode
                                                                                  jsonText  get responseContent ▾
                                                    notFound  " not found "
```