Covid patient health monitoring system

1. Introduction

a. Overview

- The mobile app is used to keep track of the covid patients.
- Patients are given a wearable device that measures their body temperature, pulse rate, and blood pressure.
- The health parameters are sent to the IBM IoT platform by the wearable device.
- Nurses and doctors can use the app to keep track on their patients' health even when they are not present.
- If a patient's health metrics are abnormal, they will be notified by text message.

b. Purpose

To monitor and manage the potential infected patients of COVID-19 without failing the COVID protocols by controlling and monitoring the affected ones remotely.

2. Literature Survey

a. Existing problem

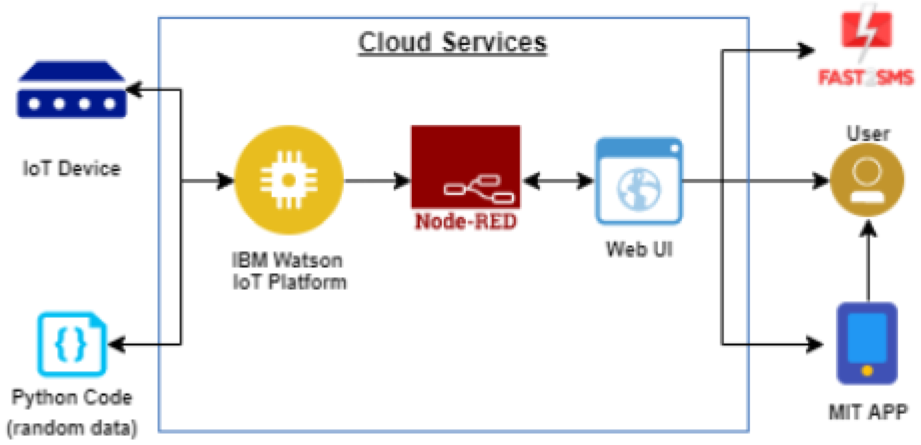
In a COVID 19 affected casualty having dense population, controlling affected ones is a humanly impossible task. The protocols for patients are difficult to be followed under such circumstances. The front-line workers like the doctors and nurses have been exposed to this virus and losing them is a certainty.

b. Proposed solution

IoT based wearable monitoring device is designed to help in monitoring the COVID affected patients round the clock in-home isolation or in quarantine centres to check the seriousness of their conditions, using their vital parameters like body temperature, blood pressure and pulse rate. The wearable sensor placed on the body is connected to edge node in IoT cloud where the data is processed and analysed to define the state of health condition. The proposed system is implemented with three layered functionalities as wearable IoT sensor layer, cloud layer with Application Peripheral Interface (API) and Android web layer for mobile phones. Each layer has individual functionality, first the data is measured from IoT sensor layer to define the health symptoms. The next layer is used to store the information in the cloud database for preventive measures, alerts, and immediate actions. The Android mobile application layer is responsible for providing notifications alert to the doctor.

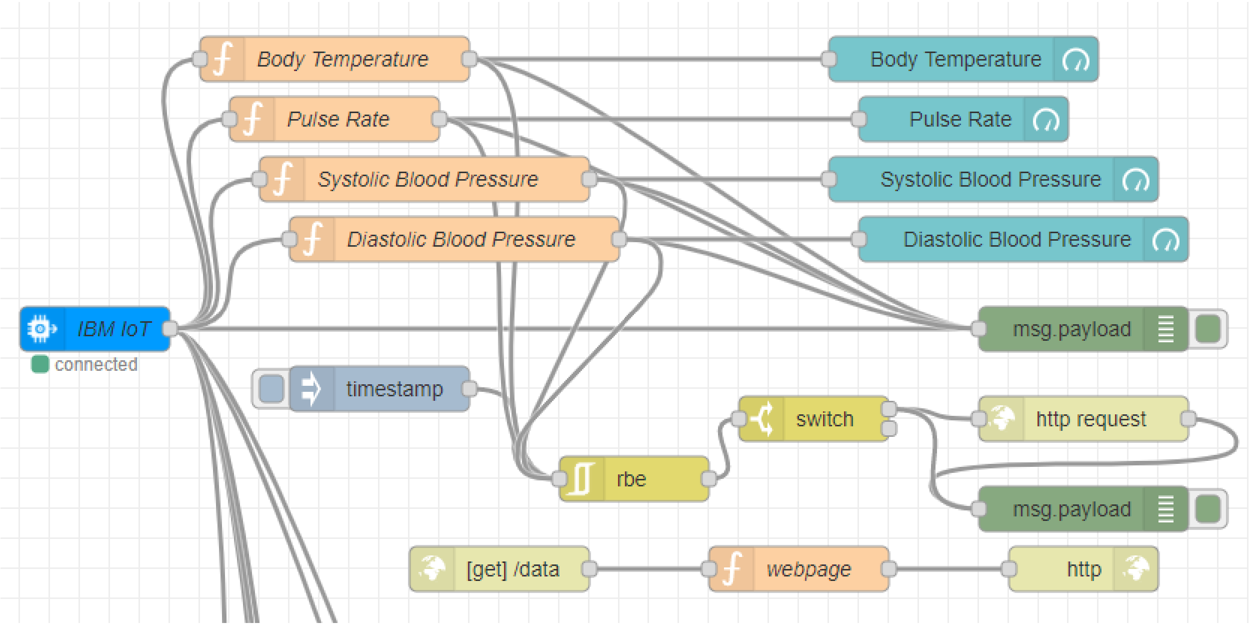
3. Theoretical Analysis

a. Block diagram

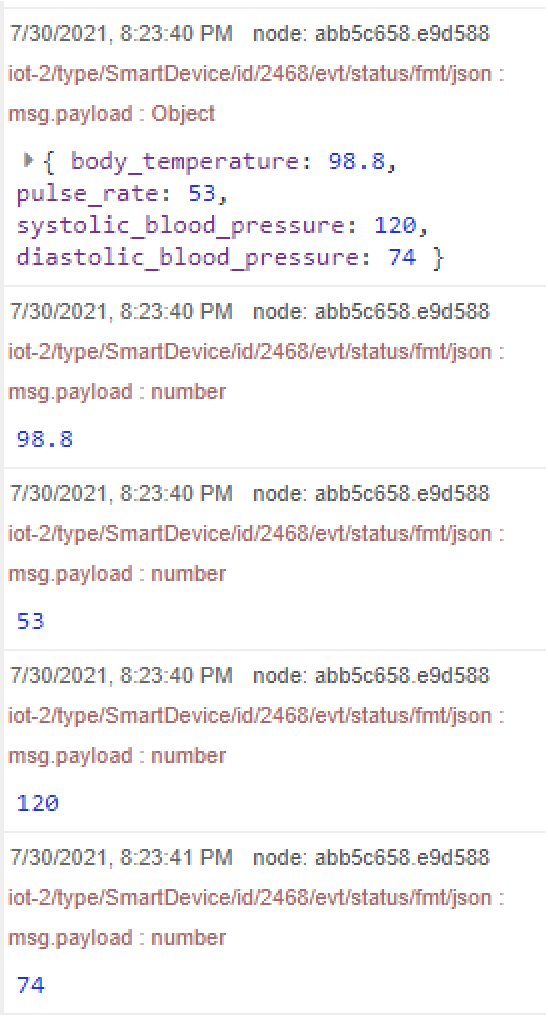


b. Hardware / Software designing

Node-red design:

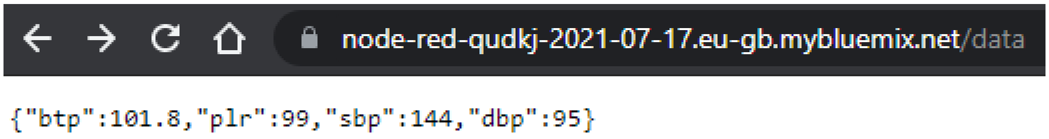


Node-red output:



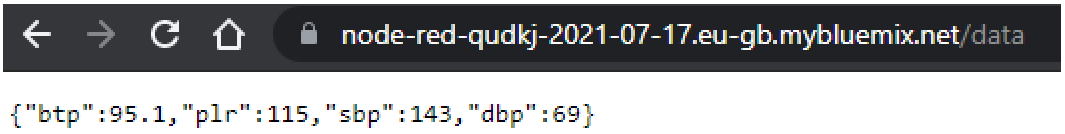
```
7/30/2021, 8:23:40 PM   node: abb5c658.e9d588
iot-2/type/SmartDevice/id/2468/evt/status/fmt/json :
msg.payload : Object
 ▸{ body_temperature: 98.8,
 pulse_rate: 53,
 systolic_blood_pressure: 120,
 diastolic_blood_pressure: 74 }

7/30/2021, 8:23:40 PM   node: abb5c658.e9d588
iot-2/type/SmartDevice/id/2468/evt/status/fmt/json :
msg.payload : number
 98.8

7/30/2021, 8:23:40 PM   node: abb5c658.e9d588
iot-2/type/SmartDevice/id/2468/evt/status/fmt/json :
msg.payload : number
 53

7/30/2021, 8:23:40 PM   node: abb5c658.e9d588
iot-2/type/SmartDevice/id/2468/evt/status/fmt/json :
msg.payload : number
 120

7/30/2021, 8:23:41 PM   node: abb5c658.e9d588
iot-2/type/SmartDevice/id/2468/evt/status/fmt/json :
msg.payload : number
 74
```

Node-red url output:



node-red-qudkj-2021-07-17.eu-gb.mybluemix.net/data

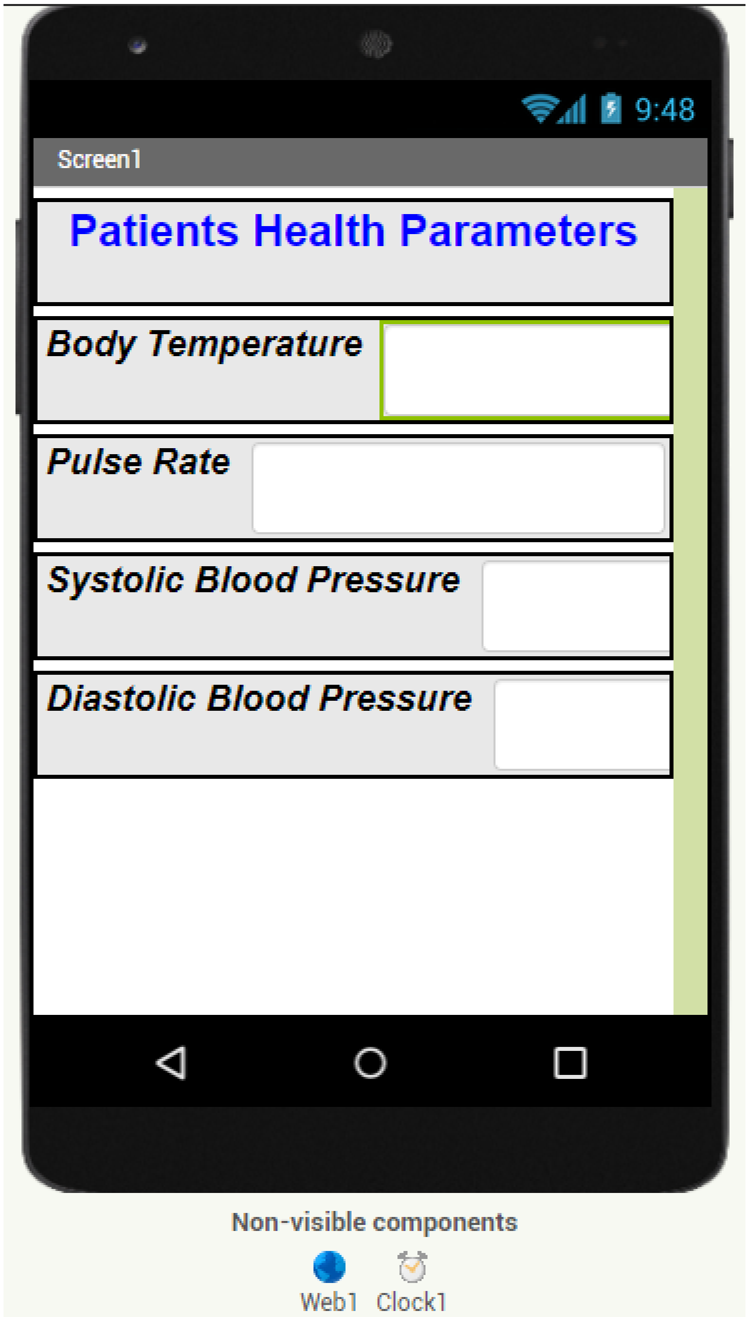
```
{"btp":101.8,"plr":99,"sbp":144,"dbp":95}
```



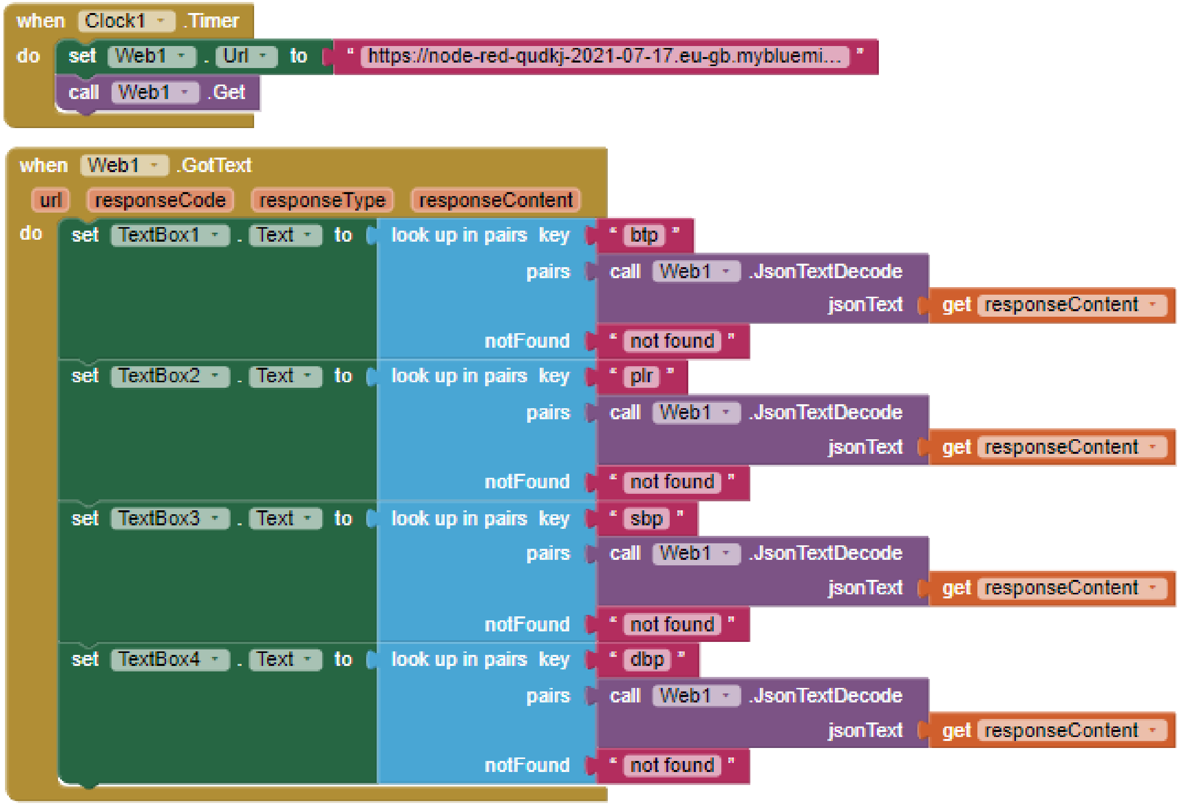
node-red-qudkj-2021-07-17.eu-gb.mybluemix.net/data

```
{"btp":95.1,"plr":115,"sbp":143,"dbp":69}
```

MIT App Design:



MIT App Blocks Design:

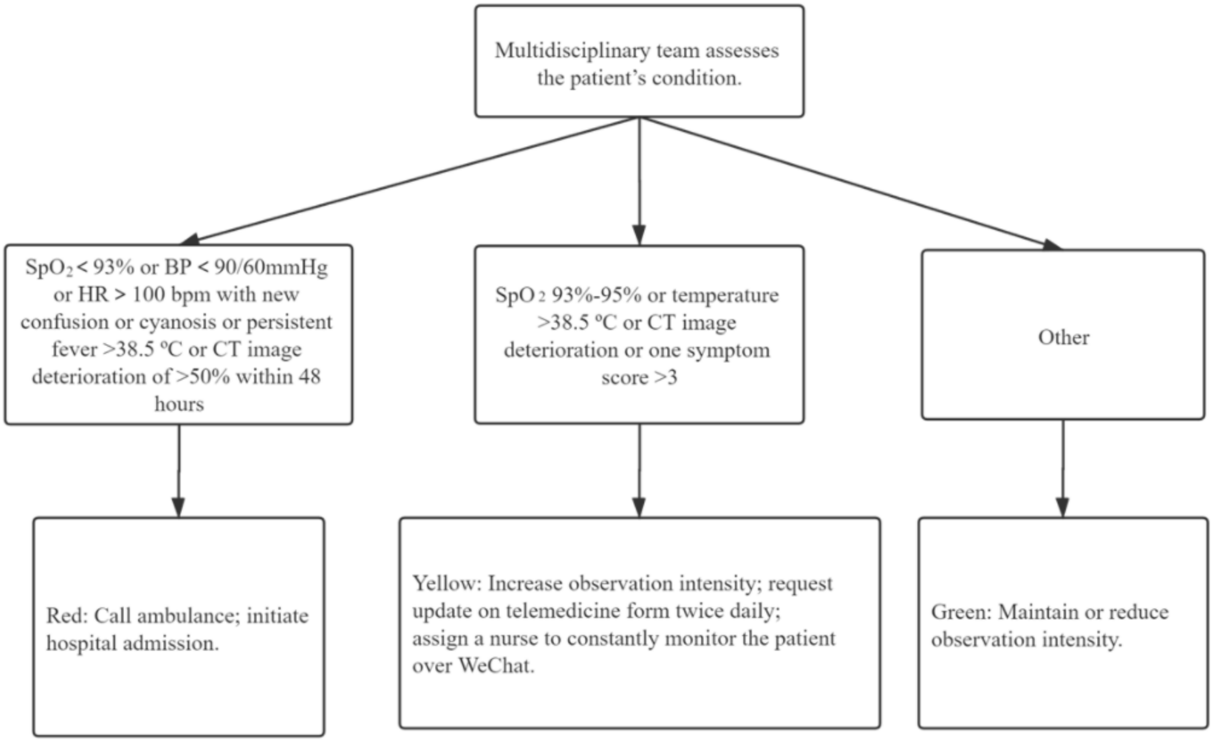


```
when Clock1 .Timer
do   set Web1 . Url to " https://node-red-qudkj-2021-07-17.eu-gb.mybluemi... "
     call Web1 .Get

when Web1 .GotText
  url   responseCode   responseType   responseContent
do   set TextBox1 . Text to   look up in pairs  key   " btp "
                                               pairs   call Web1 .JsonTextDecode
                                                                        jsonText   get responseContent
                                            notFound   " not found "
     set TextBox2 . Text to   look up in pairs  key   " plr "
                                               pairs   call Web1 .JsonTextDecode
                                                                        jsonText   get responseContent
                                            notFound   " not found "
     set TextBox3 . Text to   look up in pairs  key   " sbp "
                                               pairs   call Web1 .JsonTextDecode
                                                                        jsonText   get responseContent
                                            notFound   " not found "
     set TextBox4 . Text to   look up in pairs  key   " dbp "
                                               pairs   call Web1 .JsonTextDecode
                                                                        jsonText   get responseContent
                                            notFound   " not found "
```

4. Experimental Investigations

The display software should have features like patient details and the important health parameters. In case of any emergency different alerts are to be generated to caution the front-line workers. Once we achieve this task, the doctors can download the data on their mobile phone, laptop/desktop through application software (App). In case of abnormalities in the parameters, the doctor can interact with the patient over the phone and provide medical advice. This will help our front-line warriors to avoid exposure to the infected patients and also reduce the usage of PPE kits which they had to wear for a long duration. The wireless wearable which should be small, portable, and easy to use will enable the patients to better access the health care, improve the quality of healthcare and also provide healthcare workers with better patient care through constant monitoring of patients. This will decreases the emergency room visits and the admission to the hospitals, in turn, providing a better capacity to treat more patients.The integrated system has both API and mobile application synchronized with each other for predicting and alarming the situation. The design serves as an essential platform that defines the measured readings of COVID-19 symptoms for monitoring, management, and analysis. Furthermore, the work disseminates how digital remote platform as wearable device can be used as a monitoring device to track the health and recovery of a COVID-19 patient.

5. Flowchart


Multidisciplinary team assesses the patient's condition.
SpO2 < 93% or BP < 90/60mmHg or HR > 100 bpm with new confusion or cyanosis or persistent fever >38.5 ºC or CT image deterioration of >50% within 48 hours
SpO 2 93%-95% or temperature >38.5 ºC or CT image deterioration or one symptom score >3
Other
Red: Call ambulance; initiate hospital admission.
Yellow: Increase observation intensity; request update on telemedicine form twice daily; assign a nurse to constantly monitor the patient over WeChat.
Green: Maintain or reduce observation intensity.

6. Result

We have developed the IoT based wearable monitoring device to help in monitoring the COVID affected patients round the clock in-home isolation or in quarantine centres to check the seriousness of their conditions, using their vital parameters like body temperature, blood pressure and pulse rate and the Remote Patient Monitoring doctors quickly developed care plans specific to patients with COVID-19.

7. Advantages & Disadvantages

Advantages:

- This will help our front-line warriors to avoid exposure to the infected patients and also reduce the usage of PPE kits which they had to wear for a long duration.
- better patient care through constant monitoring of patients.
- This will decreases the emergency room visits and the admission to the hospitals.
- providing a better capacity to treat more patients

Disadvantages:

- Security and privacy: Security and privacy remain a major concern deterring users from using IoT technology for medical purposes, as healthcare monitoring solutions have the potential to be breached.
- Risk of failure: Failure or bugs in the hardware or even power failure can impact the performance of sensors and connected equipment, placing healthcare operations at risk.
- Integration: There is no consensus regarding IoT protocols and standards, so devices produced by different manufacturers may not work well together.
- Cost: While IoT promises to reduce the cost of healthcare in the long term, the cost of its implementation in hospitals and staff training is quite high

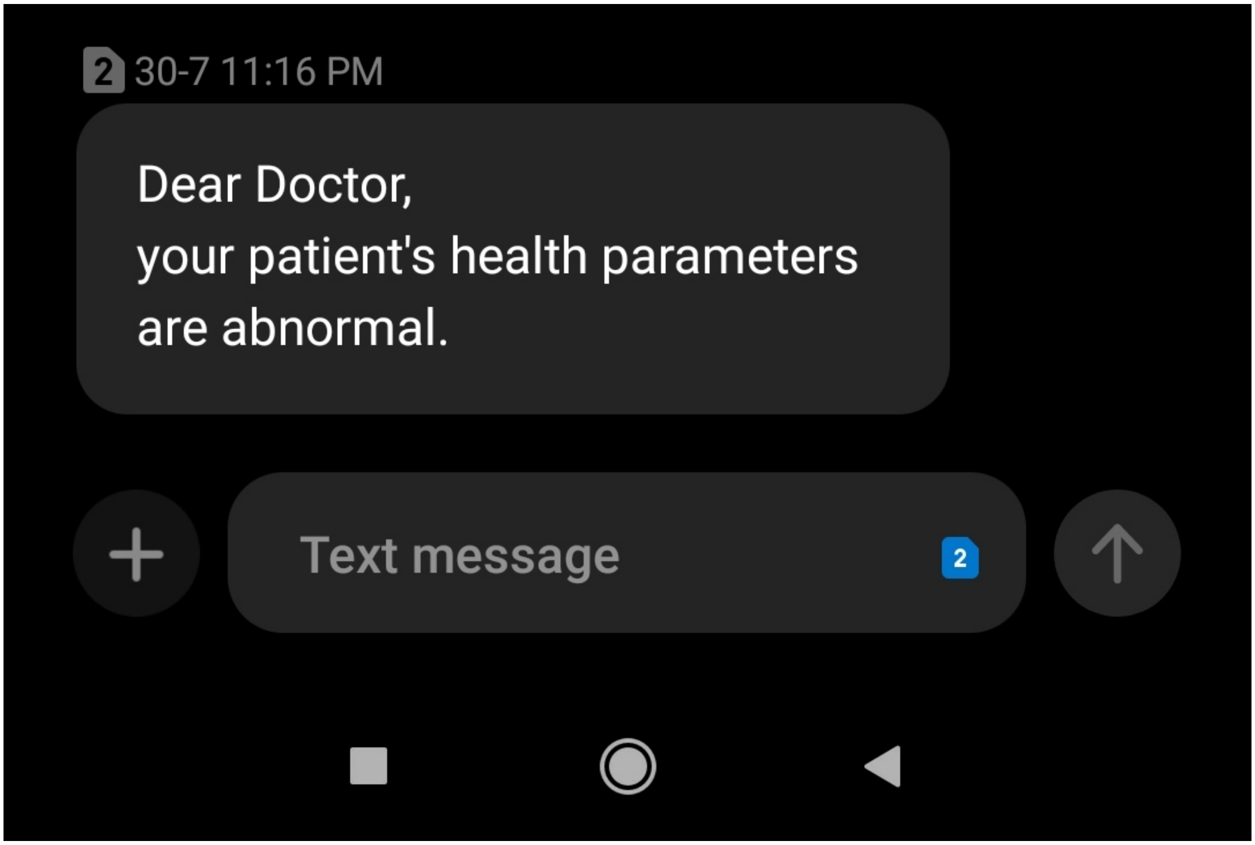
8. Applications

After rigorous testing and evaluation, the proposed system can be deployed in hospitals for use in various units. The designed mobile and web application, once fully developed, can be plugged into existing web domains of hospitals as a portal and can be launched as a fresh application for hospitals without existing domains. It is also recommended that new features such as a physiological data capturing device be incorporated into the current system.

9. Conclusion

In this Project, an IoT based wearable monitoring device is designed to measure various vital signs related to COVID-19 and notify the doctor if the health parameters are abnormal.

Fast2sms output:



Dear Doctor,
your patient's health parameters
are abnormal.

10. Future Scope

As a future direction, we plan to extend our application beyond Android platform to other IOS platforms for wide adaptability. With the efficient technique presented in this project, it is believed that this research can be extended to other areas of IoT such as agriculture for monitoring of livestock and consultation of farmers with veterinary doctors towards diagnosis, prescription and treatment of diseases in livestock in farms. Also, the new system can be extended for use in the pharmaceutical sector. The doctors can send prescriptions to the pharmacist for recommendation of dosage and possible dispensing of medications to patients. Finally, it will be of interest to carry out an evaluation of the overall performance of the proposed system using different mathematical and statistical evaluation tools.

11. Bibliography

[1] https://smartinternz.com/Student/guided_project_workspace/4792
[2] https://smartinternz.com/Student/guided_project_info/4792#
[3] https://www.python.org/downloads/windows/

[4] https://www.youtube.com/watch?v=2xIvEuFcUJY&feature=youtu.be

[5] https://youtu.be/R46iK-XYKW4

[6] https://youtu.be/Doh9ADU8CyU

[7] http://ai2.appinventor.mit.edu/

[8]http://www.iotgyan.com/learning-resource/sending-sensor-data-from-raspberry-pi-to-ibm-watson-

[9] https://github.com/gnaneshwarbandari/IOT/blob/main/ibm_code.py

[10]https://thesmartbridge.com/documents/projects/SmartHomeAutomationusingIBMCloud.pdf

[11] cloud.ibm.com

[12]https://node-red-qudkj-2021-07-17.eu-gb.mybluemix.net/red/#flow/c52daca6.86e1f

[13]https://smartbridge.teachable.com/courses/1450127/lectures/33562536

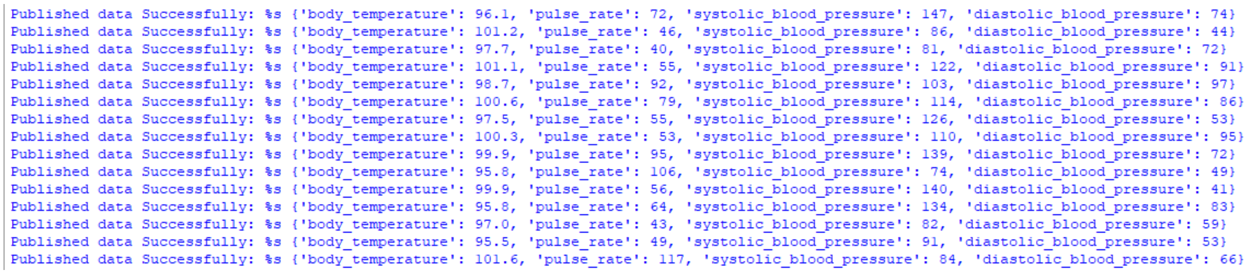
[14] https://www.fast2sms.com/dashboard/dev-api

12. Appendix

a. Source code

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "jmhx1u",
        "typeId": "SmartDevice",
        "deviceId":"2468"
    },
    "auth": {
        "token": "246810xd"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect()

while True:
    btp=round(random.uniform(95.0,102.0), 1)
    plr=random.randint(40,120)
    sbp=random.randint(60,150)
    dbp=random.randint(40,100)
    myData={'body_temperature':btp, 'pulse_rate':plr,
'systolic_blood_pressure':sbp, 'diastolic_blood_pressure':dbp}
    client.publishEvent(eventId="status", msgFormat="json", data=myData,
qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

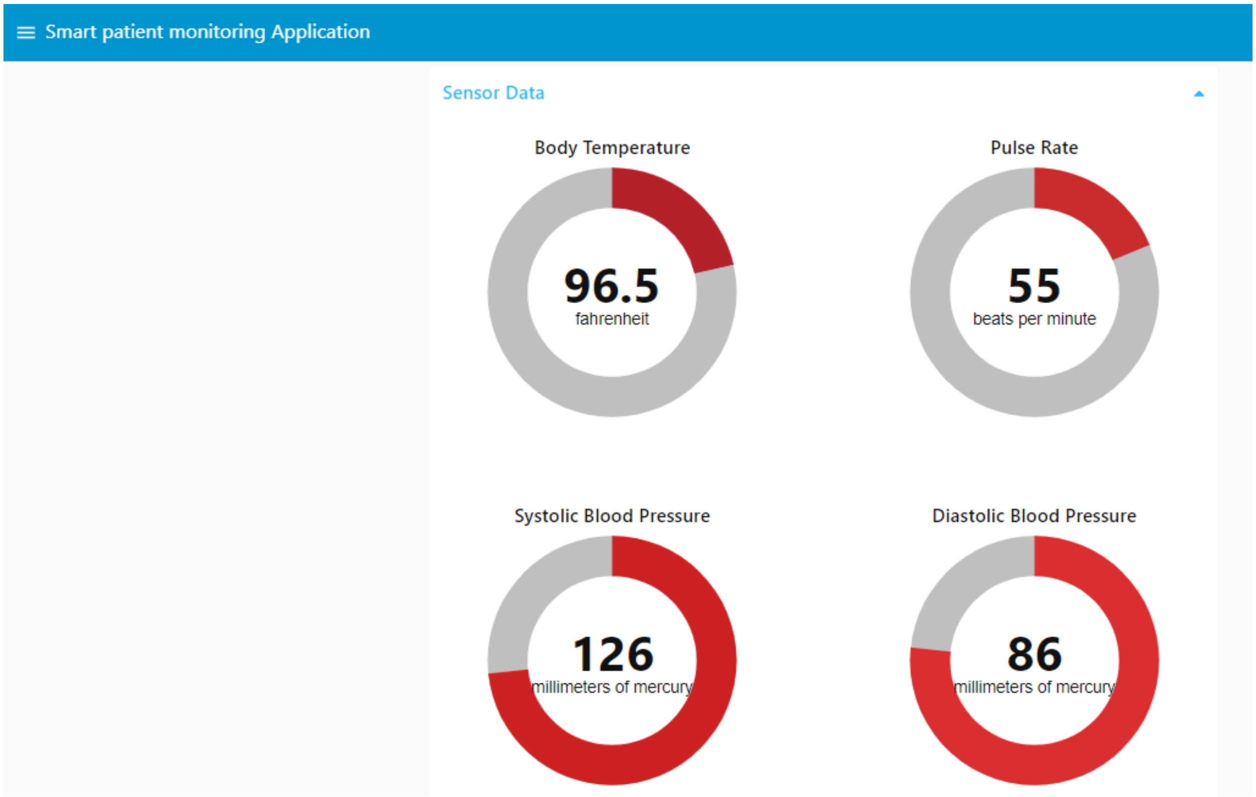
IDLE output:



```
Published data Successfully: %s {'body_temperature': 96.1, 'pulse_rate': 72, 'systolic_blood_pressure': 147, 'diastolic_blood_pressure': 74}
Published data Successfully: %s {'body_temperature': 101.2, 'pulse_rate': 46, 'systolic_blood_pressure': 86, 'diastolic_blood_pressure': 44}
Published data Successfully: %s {'body_temperature': 97.7, 'pulse_rate': 40, 'systolic_blood_pressure': 81, 'diastolic_blood_pressure': 72}
Published data Successfully: %s {'body_temperature': 101.1, 'pulse_rate': 55, 'systolic_blood_pressure': 122, 'diastolic_blood_pressure': 91}
Published data Successfully: %s {'body_temperature': 98.7, 'pulse_rate': 92, 'systolic_blood_pressure': 103, 'diastolic_blood_pressure': 97}
Published data Successfully: %s {'body_temperature': 100.6, 'pulse_rate': 79, 'systolic_blood_pressure': 114, 'diastolic_blood_pressure': 86}
Published data Successfully: %s {'body_temperature': 97.5, 'pulse_rate': 55, 'systolic_blood_pressure': 126, 'diastolic_blood_pressure': 53}
Published data Successfully: %s {'body_temperature': 100.3, 'pulse_rate': 53, 'systolic_blood_pressure': 110, 'diastolic_blood_pressure': 95}
Published data Successfully: %s {'body_temperature': 99.9, 'pulse_rate': 95, 'systolic_blood_pressure': 139, 'diastolic_blood_pressure': 72}
Published data Successfully: %s {'body_temperature': 95.8, 'pulse_rate': 106, 'systolic_blood_pressure': 74, 'diastolic_blood_pressure': 49}
Published data Successfully: %s {'body_temperature': 99.9, 'pulse_rate': 56, 'systolic_blood_pressure': 140, 'diastolic_blood_pressure': 41}
Published data Successfully: %s {'body_temperature': 95.8, 'pulse_rate': 64, 'systolic_blood_pressure': 134, 'diastolic_blood_pressure': 83}
Published data Successfully: %s {'body_temperature': 97.0, 'pulse_rate': 43, 'systolic_blood_pressure': 82, 'diastolic_blood_pressure': 59}
Published data Successfully: %s {'body_temperature': 95.5, 'pulse_rate': 49, 'systolic_blood_pressure': 91, 'diastolic_blood_pressure': 53}
Published data Successfully: %s {'body_temperature': 101.6, 'pulse_rate': 117, 'systolic_blood_pressure': 84, 'diastolic_blood_pressure': 66}
```

b. UI output Screenshot.

Node-red UI Output:



Smart patient monitoring Application

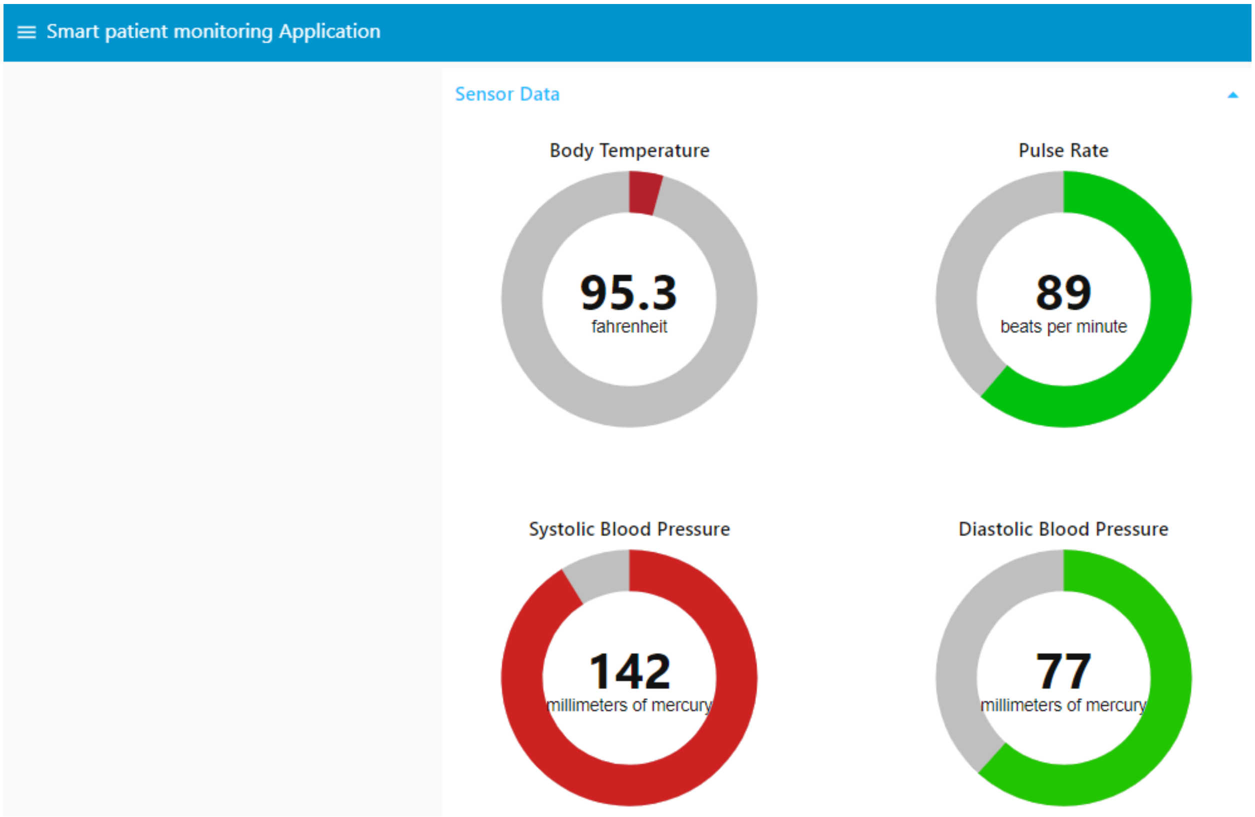
Sensor Data

Body Temperature
96.5
fahrenheit

Pulse Rate
55
beats per minute

Systolic Blood Pressure
126
millimeters of mercury

Diastolic Blood Pressure
86
millimeters of mercury



Smart patient monitoring Application

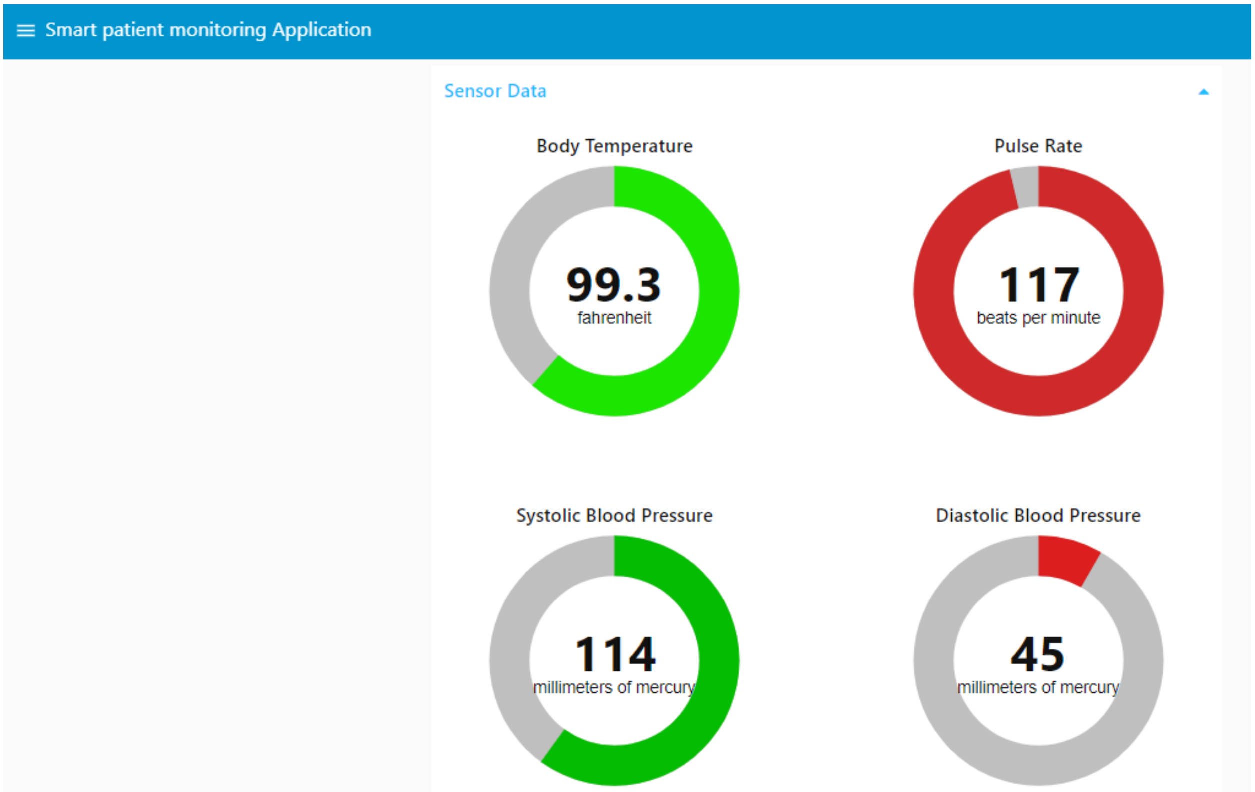
Sensor Data

Body Temperature
95.3
fahrenheit

Pulse Rate
89
beats per minute

Systolic Blood Pressure
142
millimeters of mercury

Diastolic Blood Pressure
77
millimeters of mercury



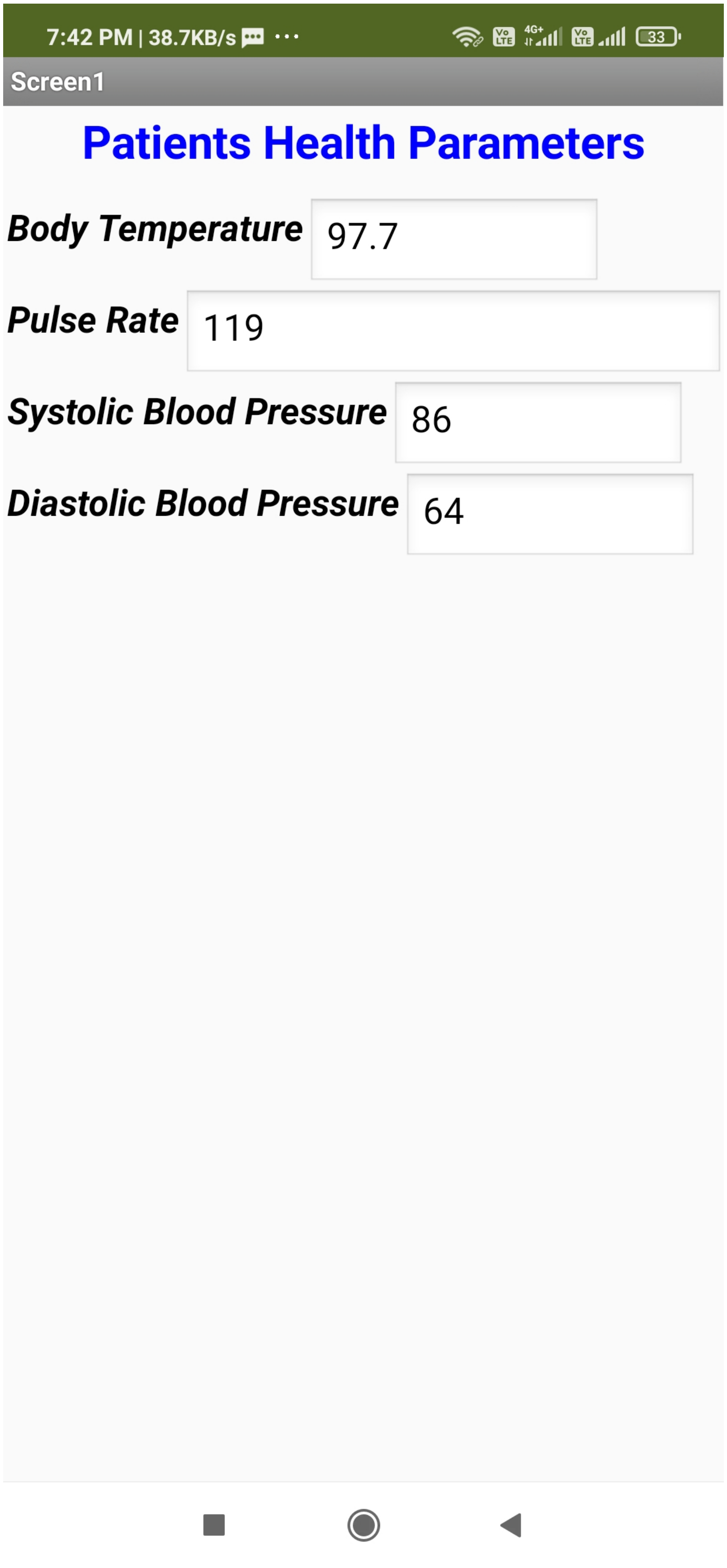
Body Temperature
99.3
fahrenheit

Pulse Rate
117
beats per minute

Systolic Blood Pressure
114
millimeters of mercury

Diastolic Blood Pressure
45
millimeters of mercury

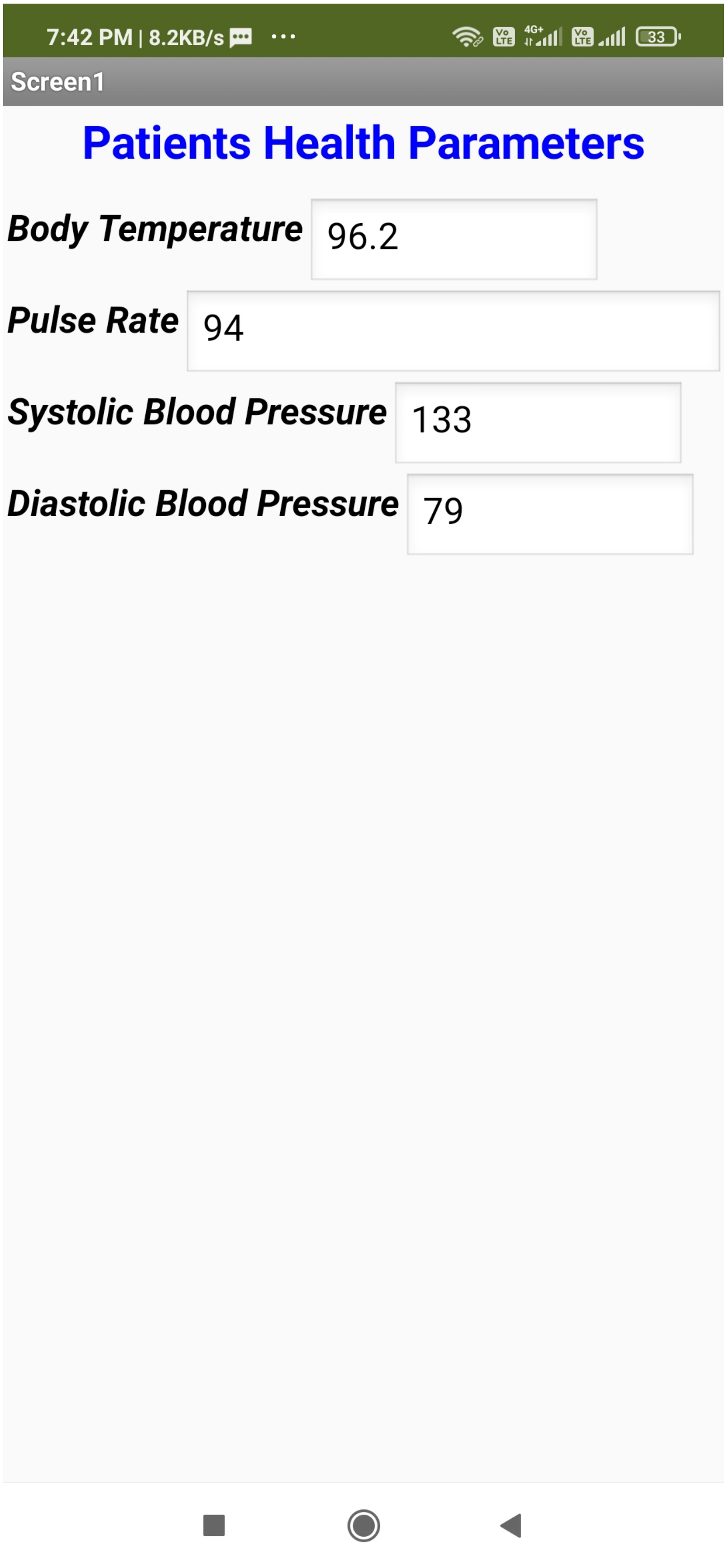
MIT Mobile App output:



Screen1

Patients Health Parameters

Body Temperature	97.7
Pulse Rate	119
Systolic Blood Pressure	86
Diastolic Blood Pressure	64



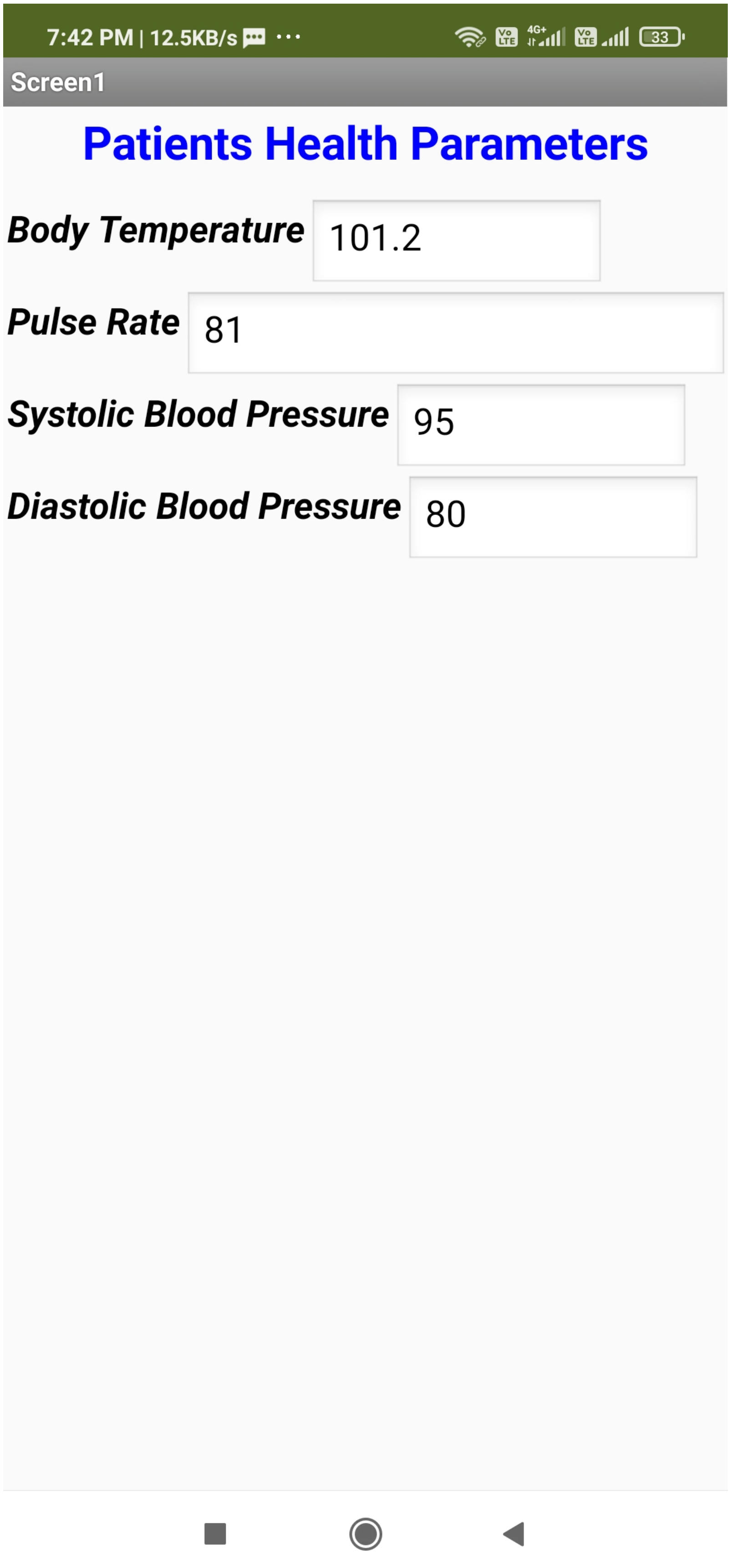
Patients Health Parameters

Body Temperature 96.2

Pulse Rate 94

Systolic Blood Pressure 133

Diastolic Blood Pressure 79



Patients Health Parameters

Body Temperature	101.2
Pulse Rate	81
Systolic Blood Pressure	95
Diastolic Blood Pressure	80