Healthcare Monitoring Assistance using Internet of Things (IoT)

Mini Project

Yash Nayak

Healthcare Monitoring Assistance using Internet of Things (IoT)

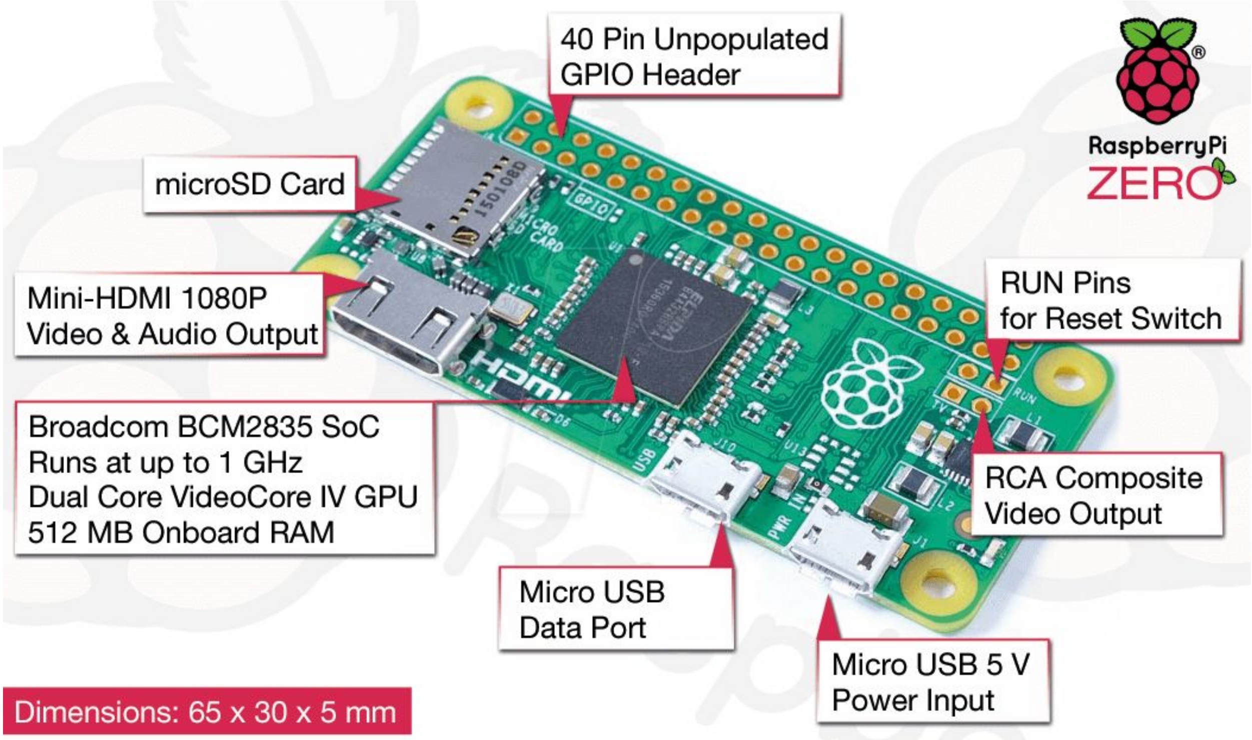
Introduction

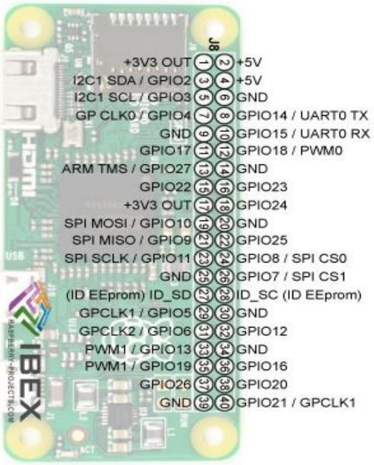
In a hospital health care monitoring system it is necessary to constantly monitor the patient’s physiological parameters. For example a pregnant woman parameters such as blood pressure (BP) and heart rate of the woman and heart rate and movements of fetal to control their health condition. This paper presents a monitoring system that has the capability to monitor physiological parameters from multiple patient bodies. In the proposed system, a coordinator node has attached on patient body to collect all the signals from the wireless sensors and sends them to the base station. The attached sensors on patient’s body form a wireless body sensor network (WBSN) and they are able to sense the heart rate, blood pressure and so on. This system can detect the abnormal conditions, issue an alarm to the patient and send a SMS/E-mail to the physician. Also, the proposed system consists of several wireless relay nodes which are responsible for relaying the data sent by the coordinator node and forward them to the base station. The main advantage of this system in comparison to previous systems is to reduce the energy consumption to prolong the network lifetime, speed up and extend the communication coverage to increase the freedom for enhance patient quality of life. We have developed this system in multi-patient architecture for hospital healthcare and compared it with the other existing networks based on multi-hop relay node in terms of coverage, energy consumption and speed.

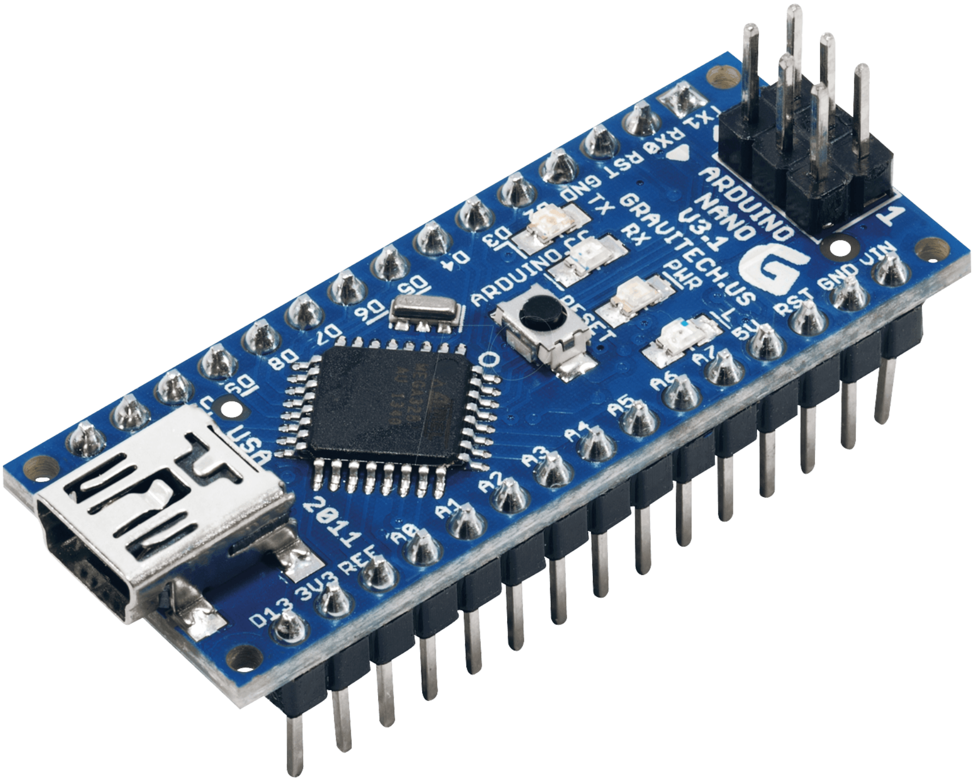
IMPLEMENTATION OF PROJECT:-

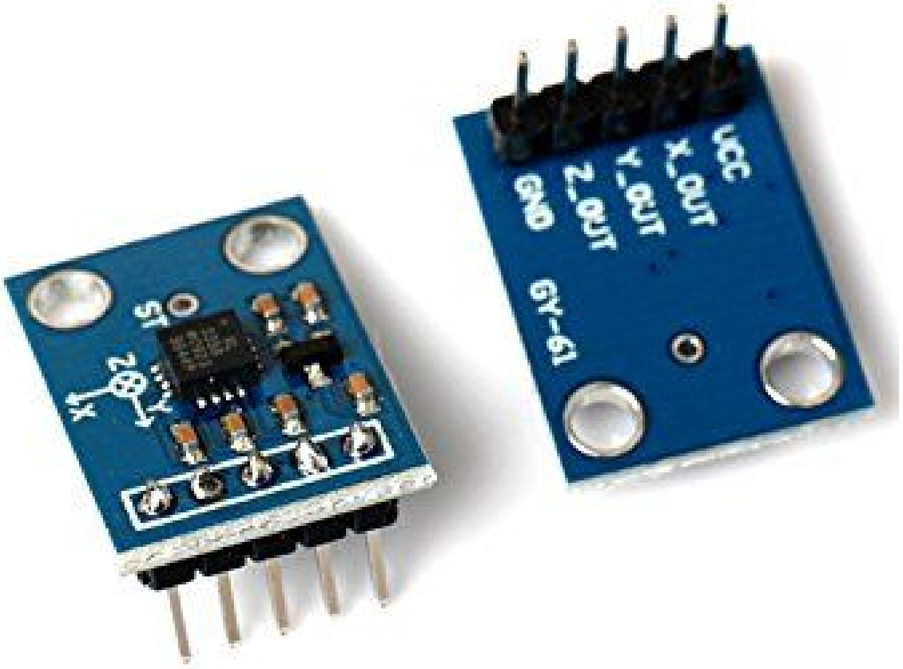
Major Components Used :-

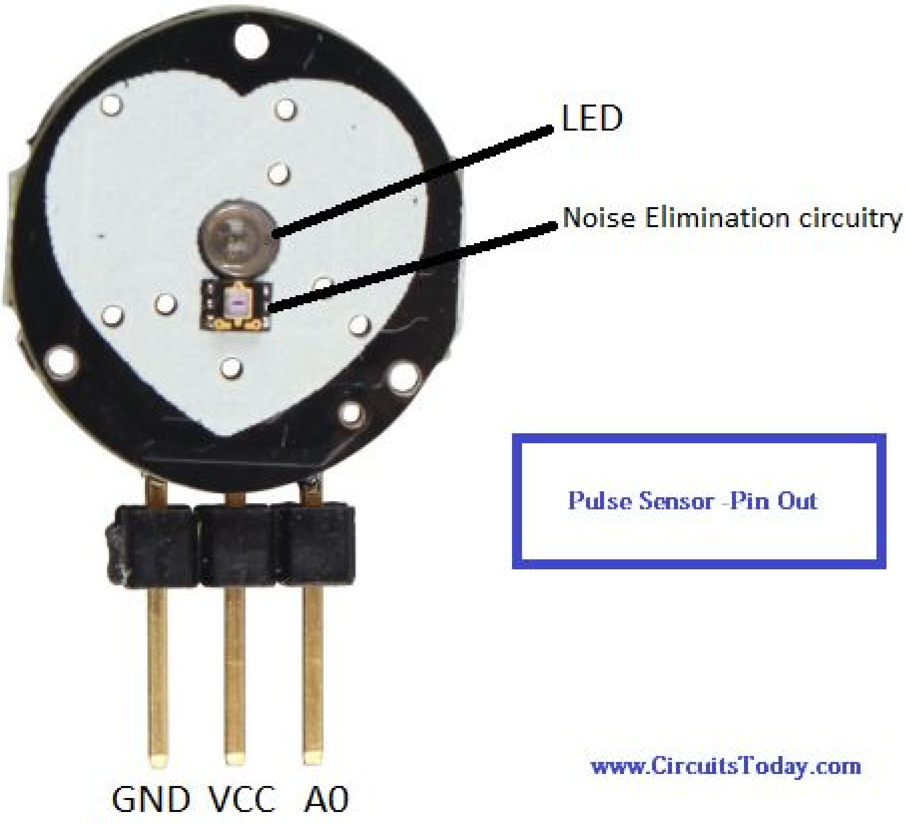
- Raspberry pi zero w
- Arduino Nano
- Accelerometer Adxl335
- Heart Rate Sensor by pulsesensor.co
- Mcp3008 ADC IC

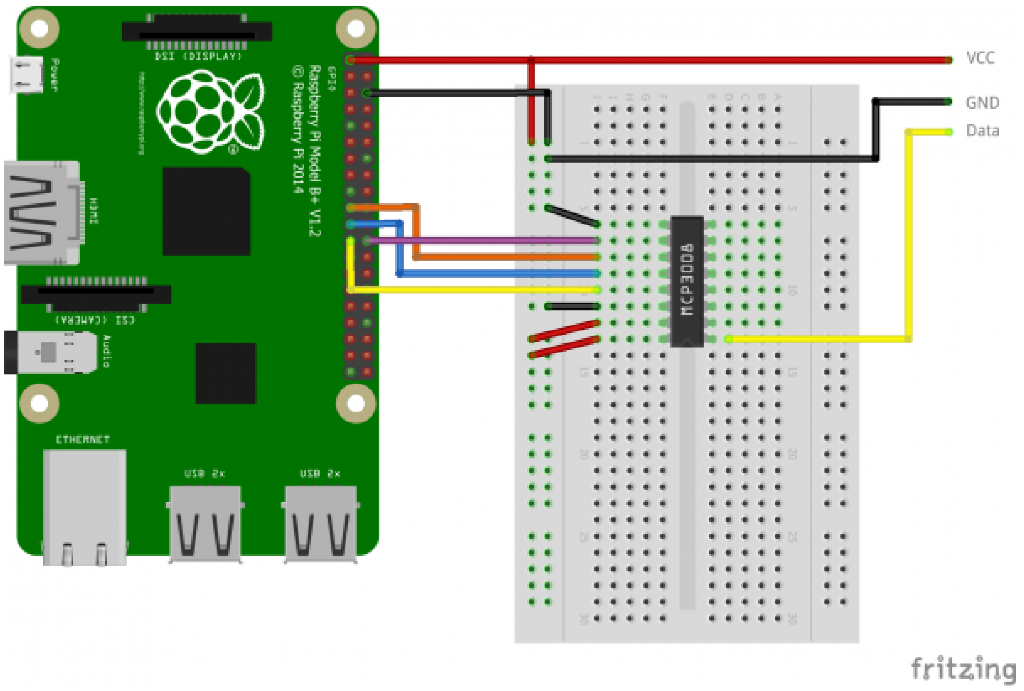



J8
+3V3 OUT 1 2 +5V
I2C1 SDA / GPIO2 3 4 +5V
I2C1 SCL / GPIO3 5 6 GND
GP CLK0 / GPIO4 7 8 GPIO14 / UART0 TX
GND 9 10 GPIO15 / UART0 RX
GPIO17 11 12 GPIO18 / PWM0
ARM TMS / GPIO27 13 14 GND
GPIO22 15 16 GPIO23
+3V3 OUT 17 18 GPIO24
SPI MOSI / GPIO10 19 20 GND
SPI MISO / GPIO9 21 22 GPIO25
SPI SCLK / GPIO11 23 24 GPIO8 / SPI CS0
GND 25 26 GPIO7 / SPI CS1
(ID EEprom) ID_SD 27 28 ID_SC (ID EEprom)
GPCLK1 / GPIO5 29 30 GND
GPCLK2 / GPIO6 31 32 GPIO12
PWM1 / GPIO13 33 34 GND
PWM1 / GPIO19 35 36 GPIO16
GPIO26 37 38 GPIO20
GND 39 40 GPIO21 / GPCLK1
IBEX
RASPBERRY-PROJECTS.COM


ARDUINO
NANO
V3.1
GRAVITECH.US
ARDUINO.CC
RESET
2011
USA


VCC
X_OUT
Y_OUT
Z_OUT
GND
GY-61
ST


LED
Noise Elimination circuitry
Pulse Sensor -Pin Out
www.CircuitsToday.com
GND
VCC
A0


VCC
GND
Data
DSI (DISPLAY)
Power
HDMI
CSI (CAMERA)
Audio
ETHERNET
USB 2x
USB 2x
GPIO
Raspberry Pi Model B+ V1.2
© Raspberry Pi 2014
MCP3008
fritzing

Program

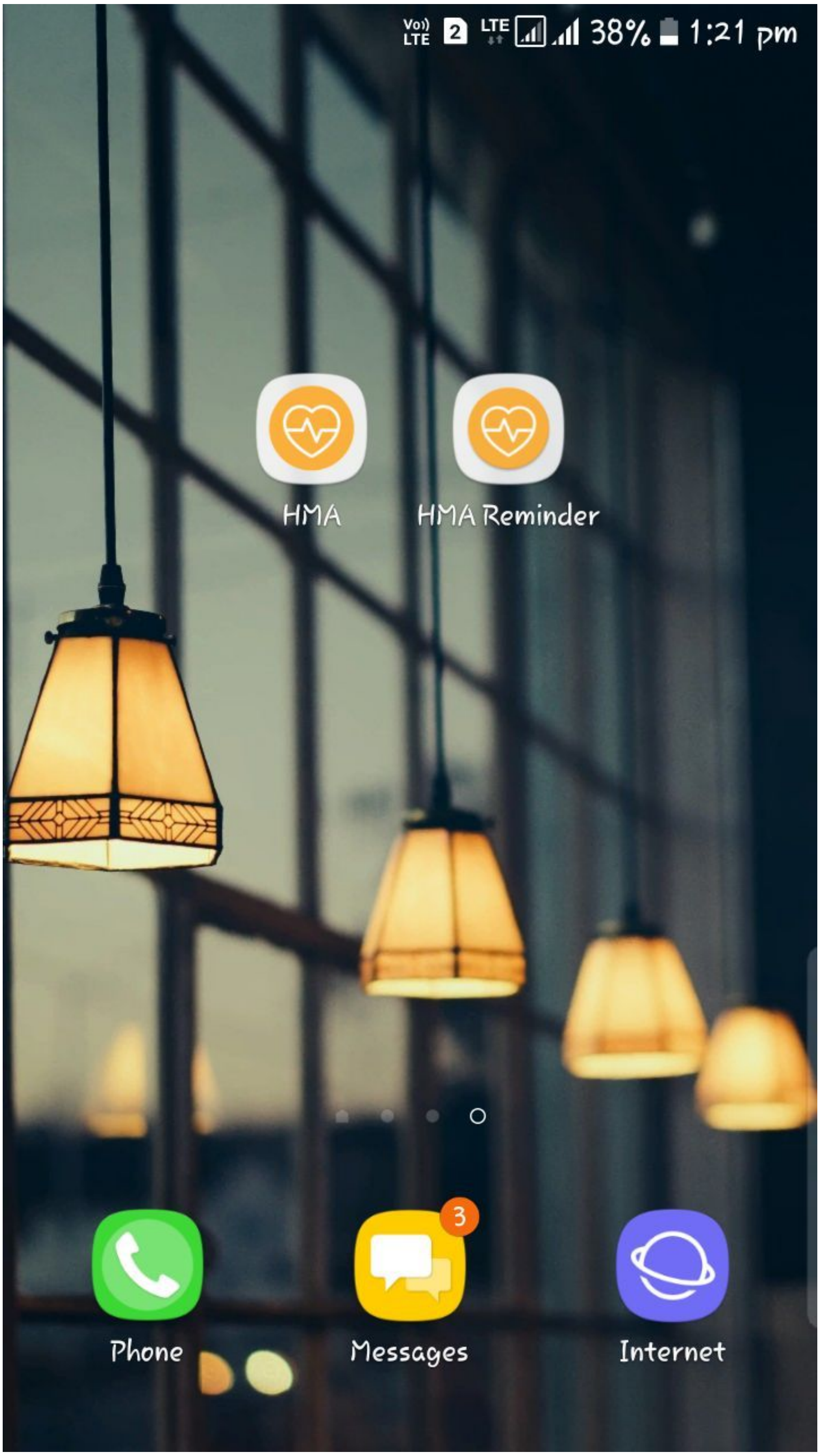
Counting steps using accelerometer

```
const int xpin=A2;
int ypin=A3;
int zpin=A4;
float threshhold=40.0;
float xval[100]={0};
float yval[100]={0};
float zval[100]={0};
float xavg;
float yavg;
float zavg;

int steps,flag=0;
void setup()
{
Serial.begin(9600);
pinMode(13,OUTPUT);
calibrate();
}
void loop()
{int acc=0;
  float totvect[100]={0};
float totave[100]={0};
  float xaccl[100]={0};
  float yaccl[100]={0};
  float zaccl[100]={0};
for (int i=0;i<100;i++)
{
xaccl[i]=float(analogRead(xpin));
delay(1);
yaccl[i]=float(analogRead(ypin));
delay(1);
zaccl[i]=float(analogRead(zpin));
delay(1);
 totvect[i] = sqrt(((xaccl[i]-xavg)* (xaccl[i]-xavg))+ ((yaccl[i] - yavg)*(yaccl[i] - yavg)) + ((zval[i] - zavg)*(zval[i] - zavg)));
 totave[i] = (totvect[i] + totvect[i-1]) / 2 ;
 delay(200);
if (totave[i]>threshhold && flag==0)
{
  steps=steps+2;
  flag=1;
}
 else if (totave[i] > threshhold && flag==1)
{
}
  if (totave[i] <threshhold  && flag==1)
  {flag=0;}
  Serial.println(steps);
}
 delay(1000);
}
void calibrate()
{
  digitalWrite(13,HIGH);
  float sum=0;
  float sum1=0;
  float sum2=0;
for (int i=0;i<100;i++)
{
xval[i]=float(analogRead(xpin));
sum=xval[i]+sum;
}
delay(100);
xavg=sum/100.0;
for (int j=0;j<100;j++)
{
xval[j]=float(analogRead(xpin));
sum1=xval[j]+sum1;
}
yavg=sum1/100.0;
delay(100);
for (int i=0;i<100;i++)
{
zval[i]=float(analogRead(zpin));
sum2=zval[i]+sum2;
}
zavg=sum2/100.0;
delay(100);
digitalWrite(13,LOW);
}
```

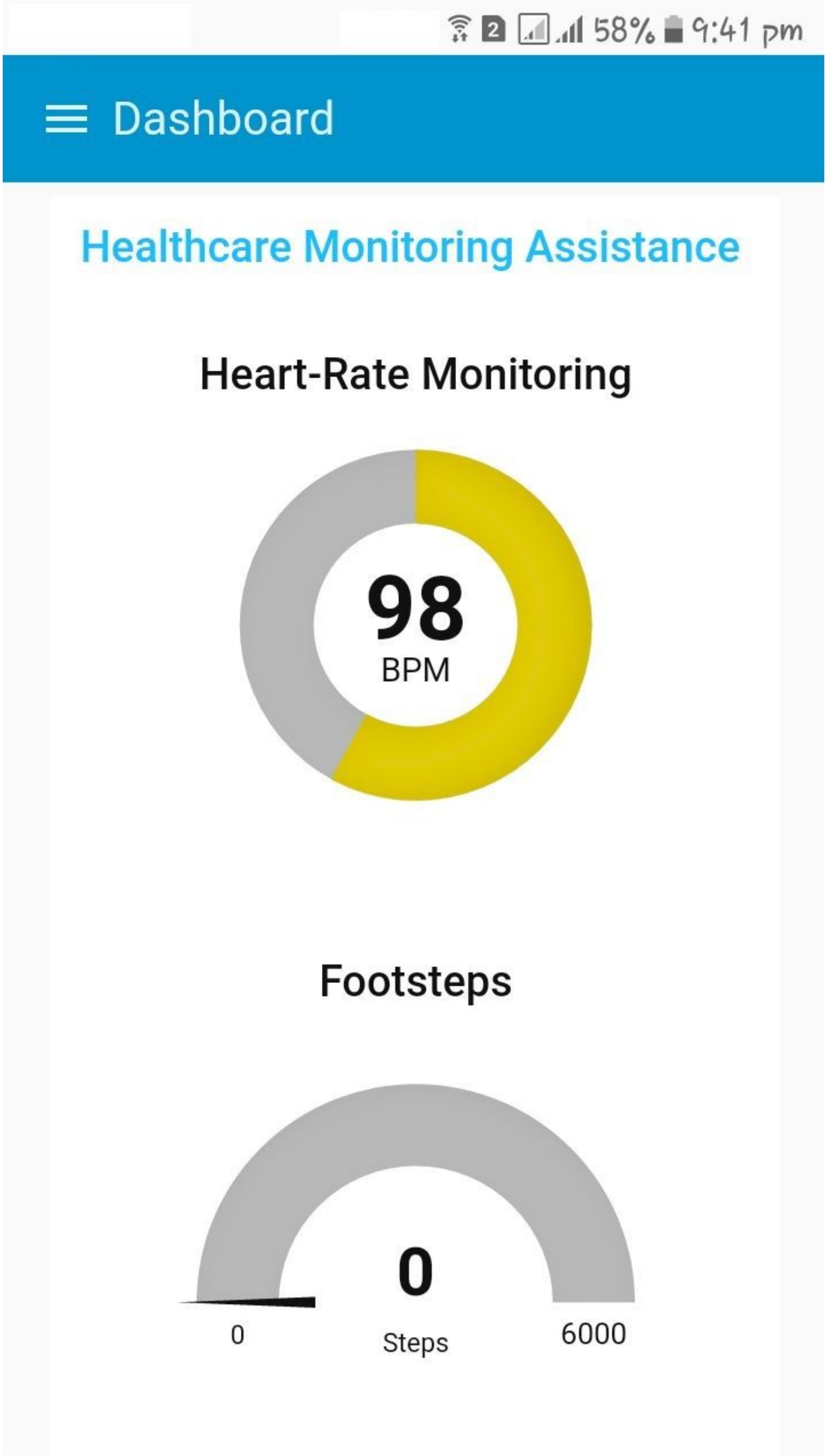
Processing pulse rate from Pulsesensor

```
#define USE_ARDUINO_INTERRUPTS true
#include <PulseSensorPlayground.h>
const int PulseWire = 0;
const int LED13 = 13;          // The on-board Arduino LED, close to PIN 13.
int Threshold = 550;           // Determine which Signal to "count as a beat" and which to ignore.
                               // Use the "Gettting Started Project" to fine-tune Threshold Value beyond default setting.
                               // Otherwise leave the default "550" value.

PulseSensorPlayground pulseSensor;  // Creates an instance of the PulseSensorPlayground object called "pulseSensor"

void setup() {
  Serial.begin(9600);          // For Serial Monitor
  // Configure the PulseSensor object, by assigning our variables to it.
  pulseSensor.analogInput(PulseWire);
  pulseSensor.blinkOnPulse(LED13);       //auto-magically blink Arduino's LED with heartbeat.
  pulseSensor.setThreshold(Threshold);
  // Double-check the "pulseSensor" object was created and "began" seeing a signal.
   if (pulseSensor.begin()) {
      //Serial.println("We created a pulseSensor Object !");  //This prints one time at Arduino power-up,  or on Arduino reset.
  }
}
void loop() {
 int myBPM = pulseSensor.getBeatsPerMinute();  // Calls function on our pulseSensor object that returns BPM as an "int".
                                  // "myBPM" hold this BPM value now.
if (pulseSensor.sawStartOfBeat()) {        // Constantly test to see if "a beat happened".
 //Serial.println("♥  A HeartBeat Happened ! "); // If test is "true", print a message "a heartbeat happened".
 //Serial.print("BPM: ");                // Print phrase "BPM: "
 Serial.println(myBPM);                        // Print the value inside of myBPM.
}
  delay(20);                   // considered best practice in a simple sketch.
}
```

Android App



HMA



58% 9:41 pm

Dashboard

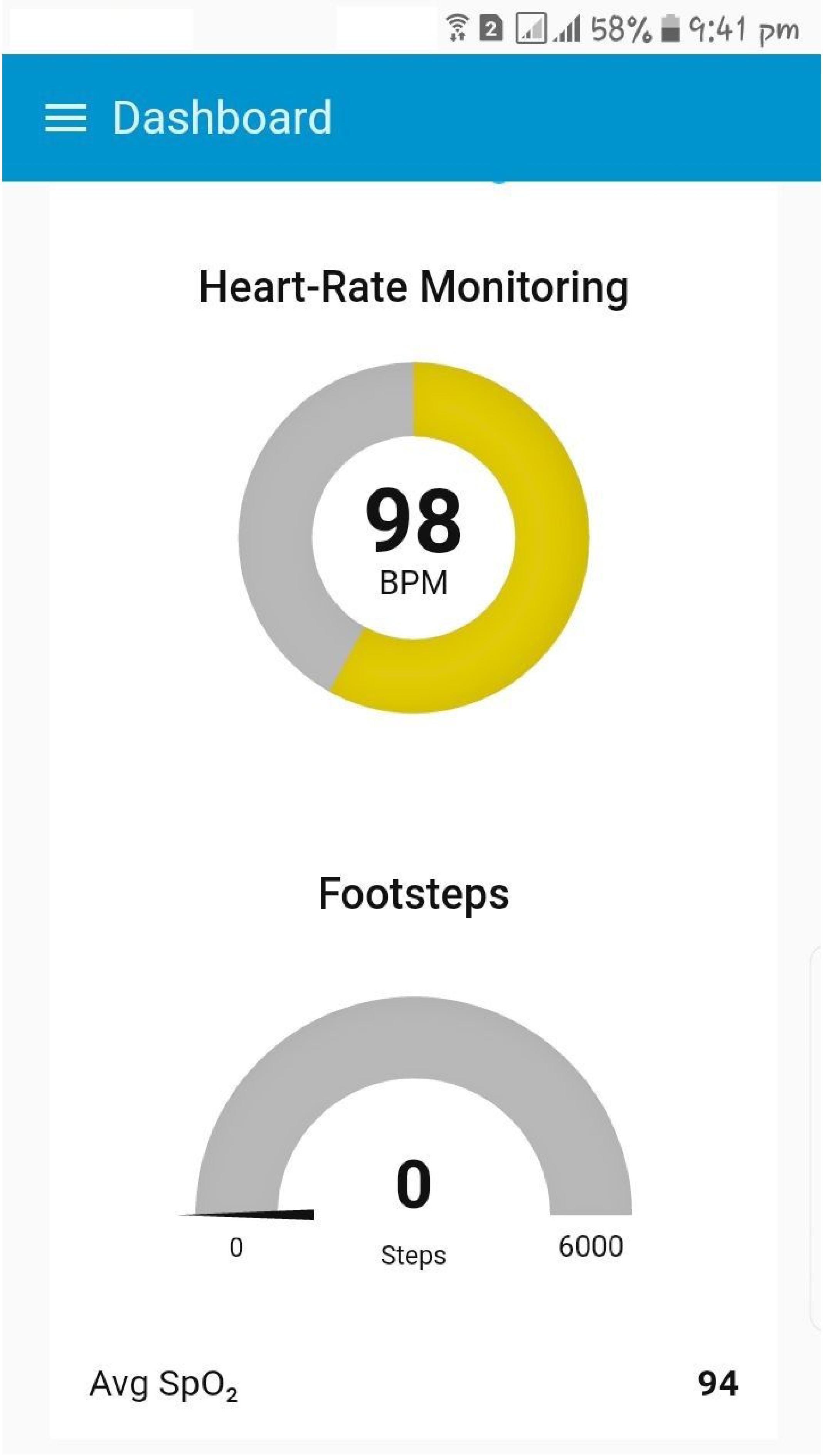
Healthcare Monitoring Assistance

Heart-Rate Monitoring

98 BPM

Footsteps

0 Steps



58% 9:41 pm

Dashboard

Heart-Rate Monitoring

98

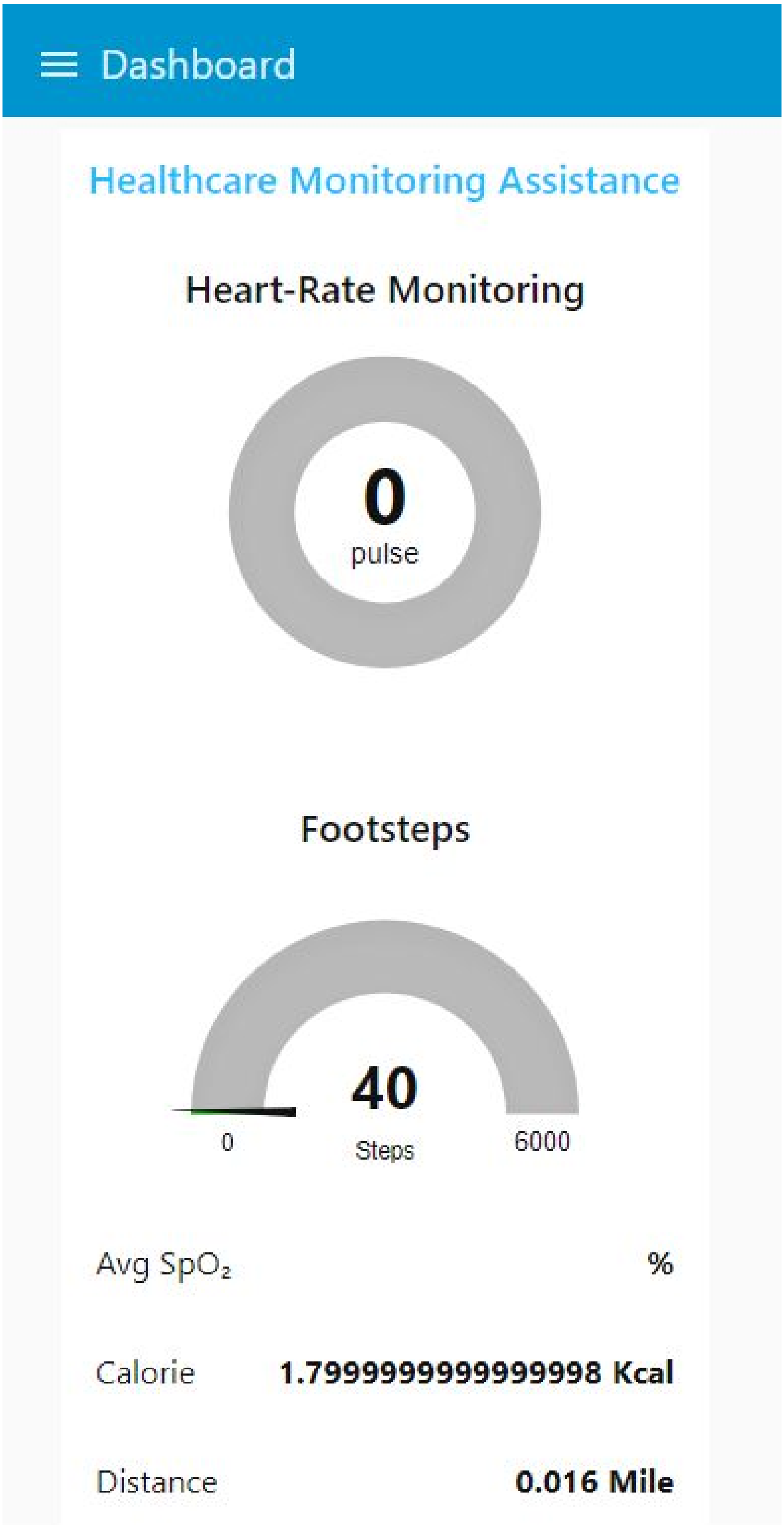
BPM

Footsteps

0

Steps

Avg SpO_2 94



Healthcare Monitoring Assistance

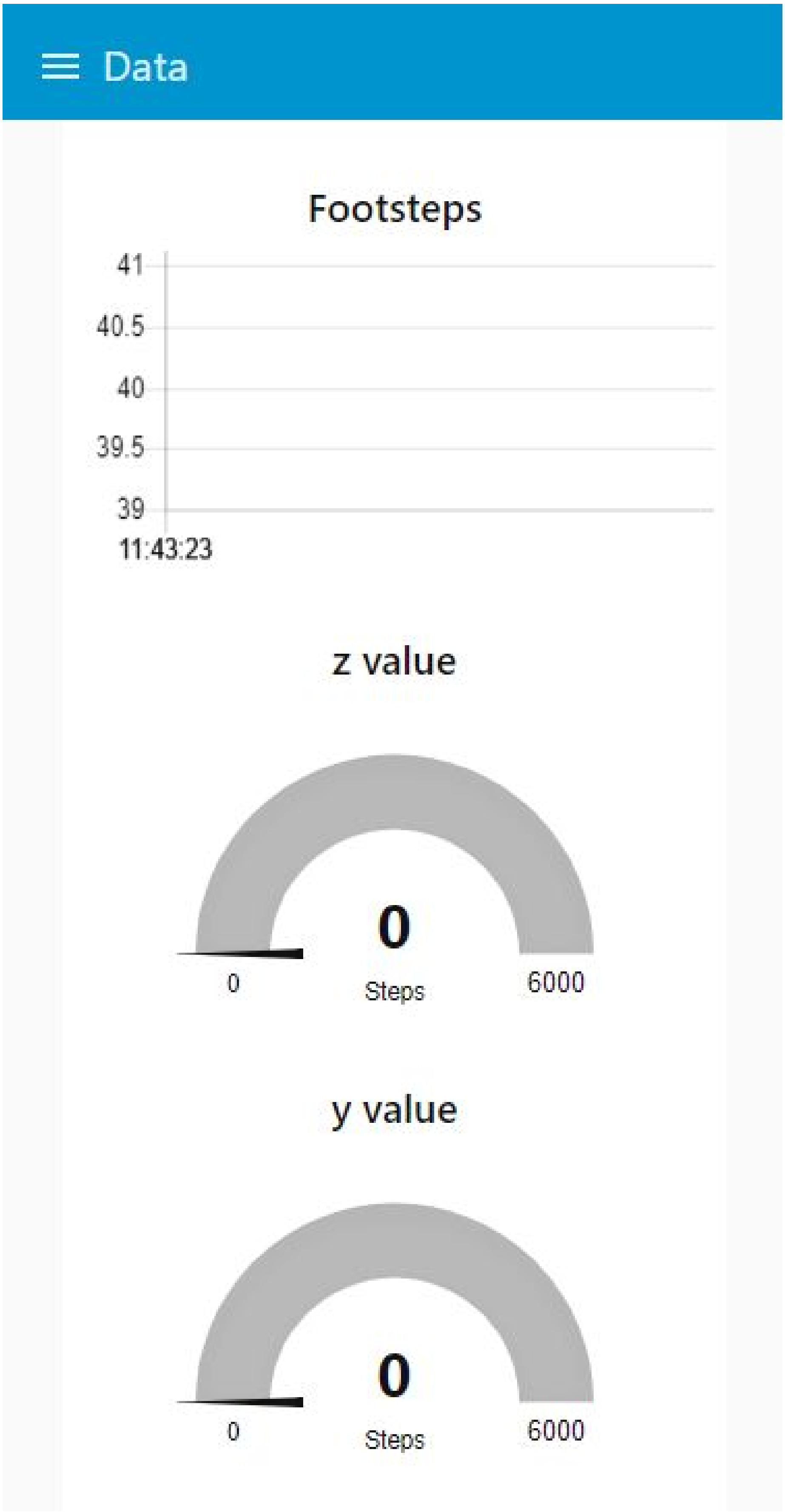
Heart-Rate Monitoring

Footsteps

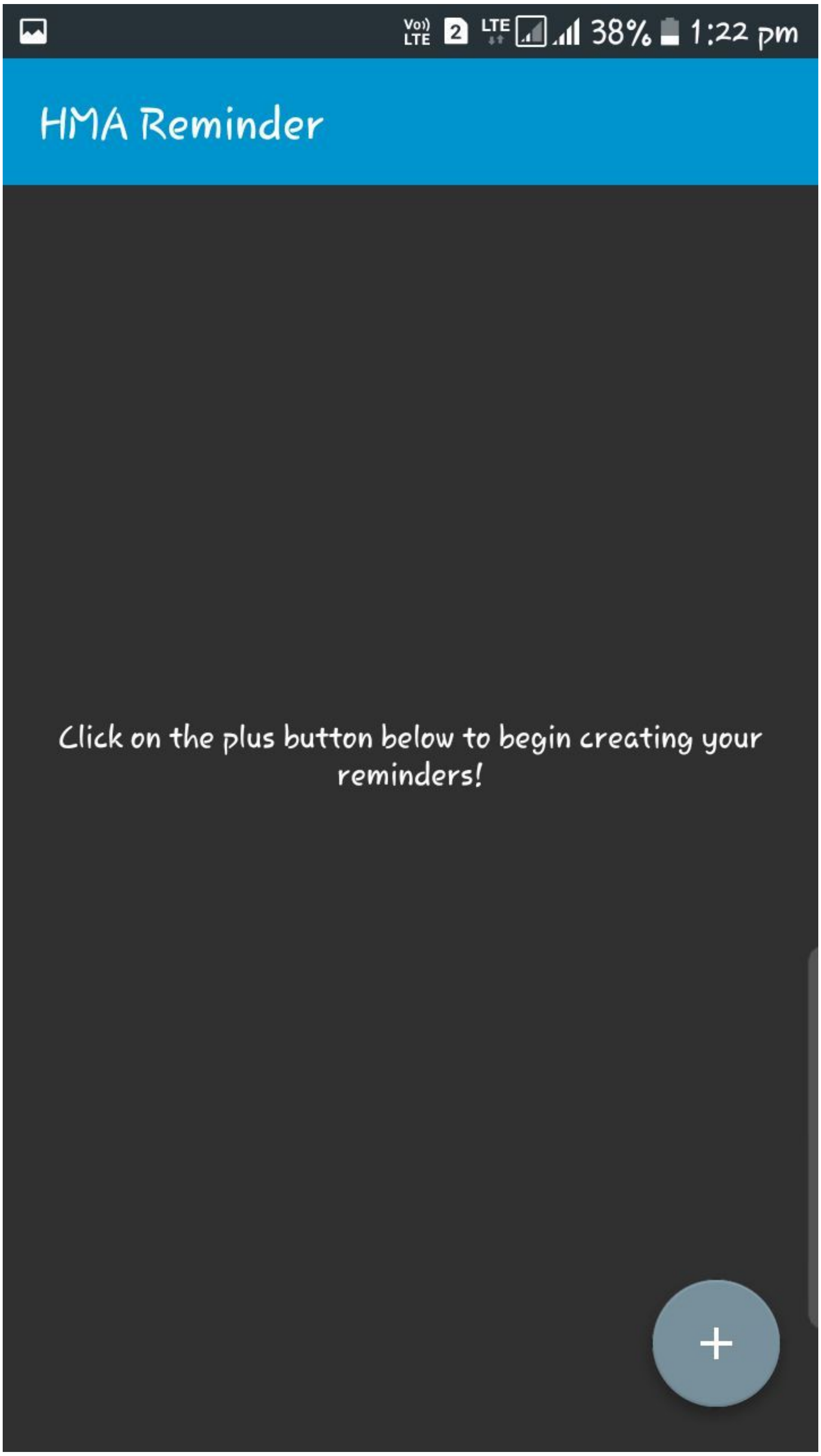
Avg SpO₂ %

Calorie 1.7999999999999998 Kcal

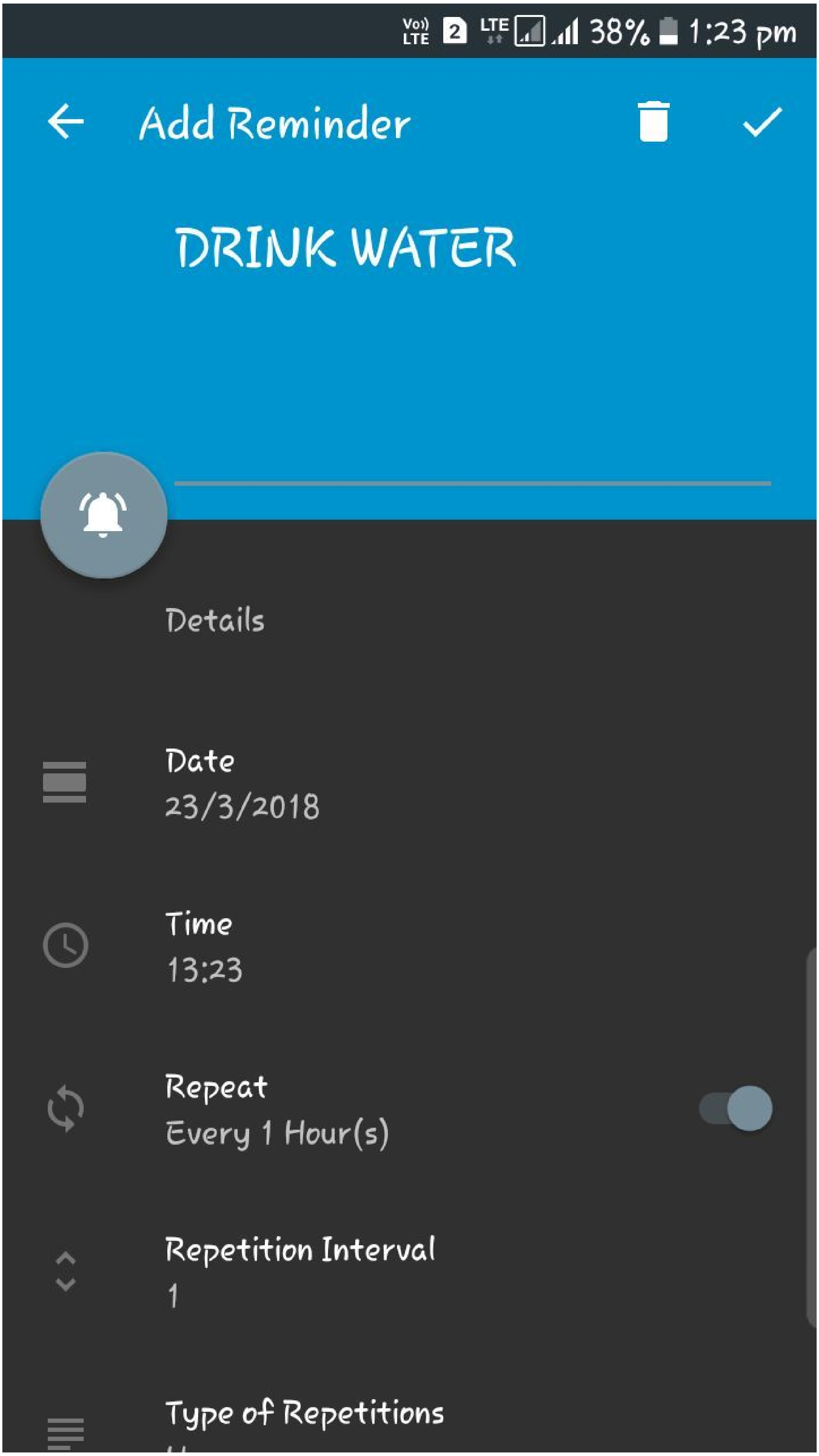
Distance 0.016 Mile


Data
Footsteps
41
40.5
40
39.5
39
11:43:23
z value
0
0
Steps
6000
y value
0
0
Steps
6000

HMA Reminder



Click on the plus button below to begin creating your reminders!



Add Reminder

DRINK WATER

Details

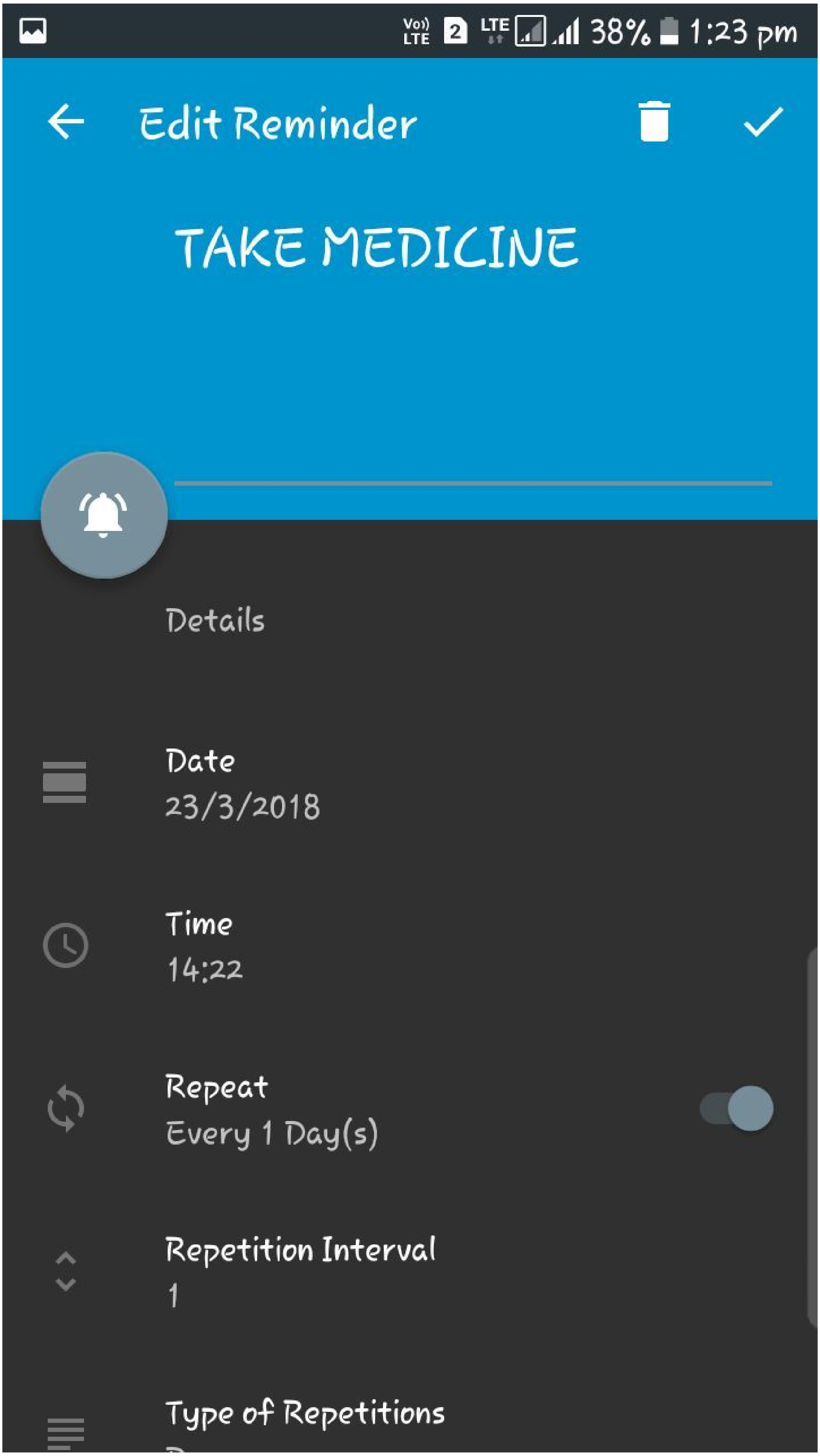
Date
23/3/2018

Time
13:23

Repeat
Every 1 Hour(s)

Repetition Interval
1

Type of Repetitions



Edit Reminder

TAKE MEDICINE

Details

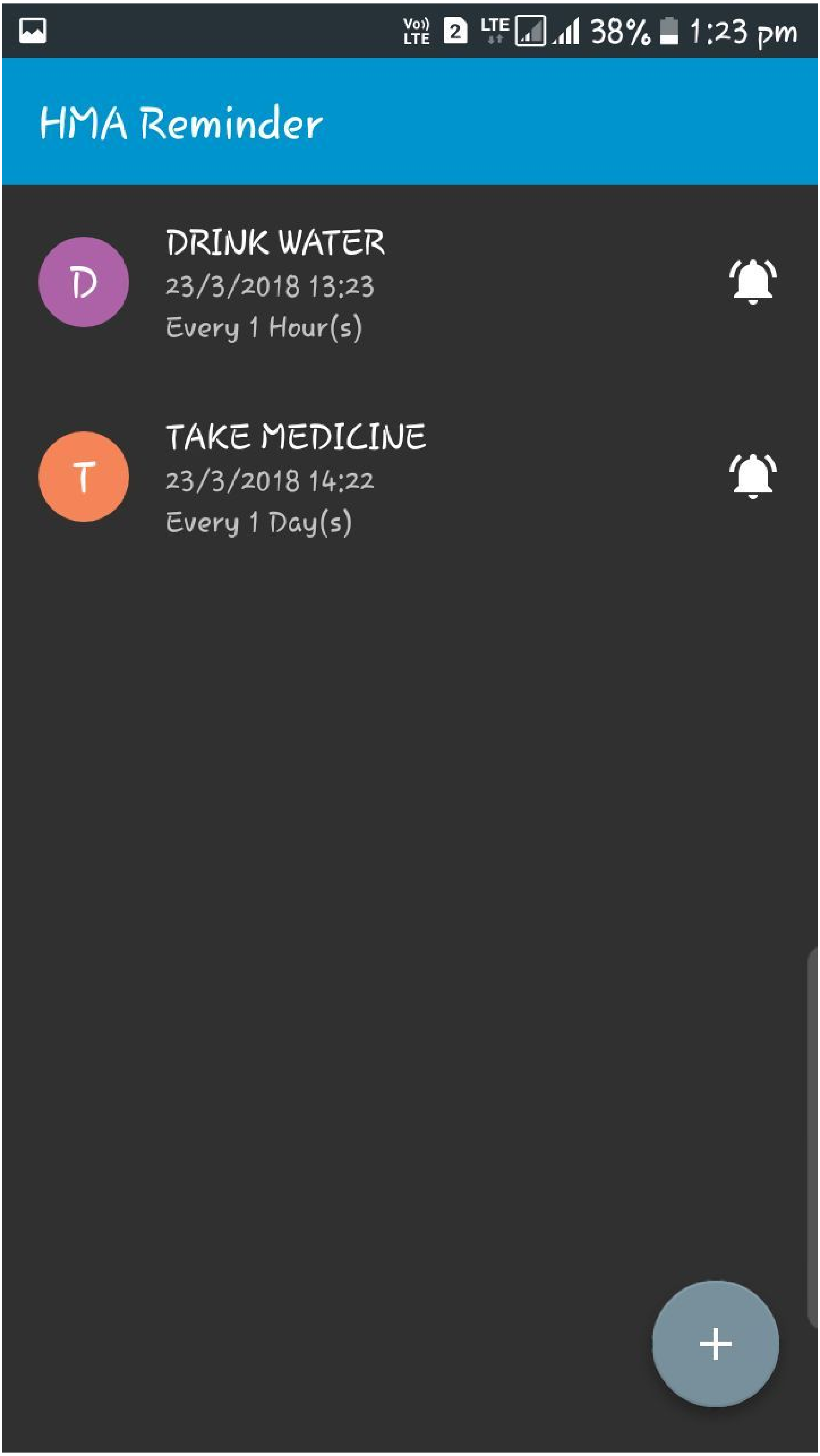
Date
23/3/2018

Time
14:22

Repeat
Every 1 Day(s)

Repetition Interval
1

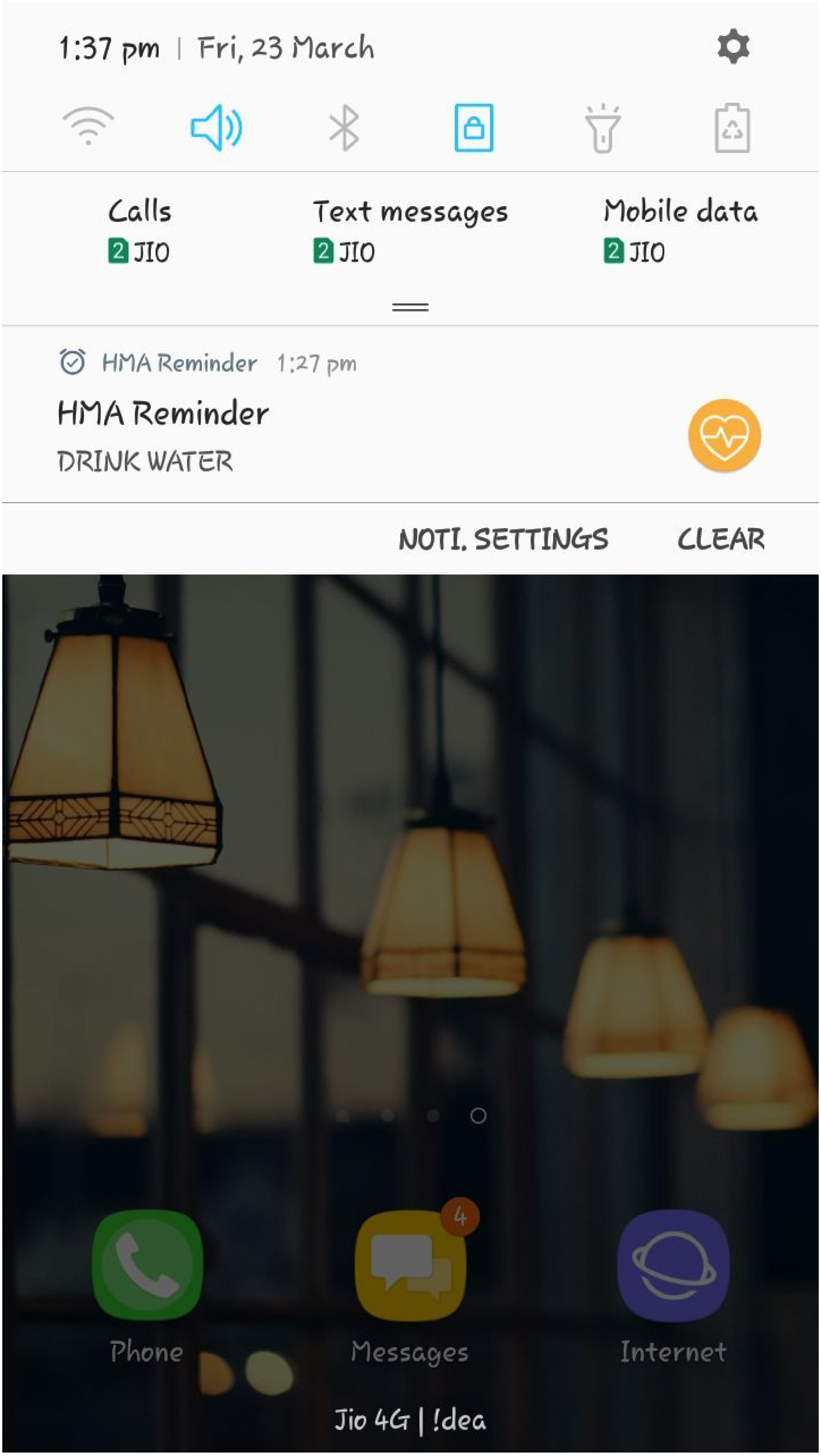
Type of Repetitions



HMA Reminder

DRINK WATER
23/3/2018 13:23
Every 1 Hour(s)

TAKE MEDICINE
23/3/2018 14:22
Every 1 Day(s)



1:37 pm | Fri, 23 March

Calls 2 JIO

Text messages 2 JIO

Mobile data 2 JIO

HMA Reminder 1:27 pm

HMA Reminder

DRINK WATER