Health**Moni**

Your personal healthcare companion

Date: 2022/11/29

Table of contents

Section 1: Summary

Healthmoni is a hand ring that helps users to manage their health data automatically. It constantly records the user step count to generate calorie consumption. It also can report heartbeat and blood pressure by using a plugin sensor. All the data uploaded will further process to show the tendency of the user performance. The target users are people with diabetes mellitus. By using Healthmoni, they can save time in recording their activity level, calories, etc. In addition, they can have their own health profile to understand their body performance. The measurement results from Health are reliable and satisfactory.

Section 2: Introduction

According to statistics, there is 10% of the population with diabetes mellitus in Hong Kong. And the number keeps increasing steadily. Diabetes mellitus can cause insufficient insulin secretion, resulting in a glucose level rise. In the long run, high glucose could damage patients' organs including the cardiovascular, retina, nerves, and kidneys. In order to control or prevent diabetes mellitus, patients need to record their health information such as glucose level, calorie consumption regularly for weight control, since obesity is one of the primary risk factors of Diabetes mellitus. And they need to perform a body check to detect early symptoms of Diabetes Mellitus. All these recordings have to be done manually and periodically. It is time-consuming to manage these records and analyze the results from these records. It is because the amount of data grows as time grows. Eventually, handling these health records becomes another problem. Here, we provide our product Healthmoni with these problems. It is a lightweight hand ring that can automatically track users' activity levels and convert them back to calorie consumption. Also, with a plugin pressure sensor, it can collect users' heart pressure data and upload it to the cloud. The users can upload their glucose levels through a personal health profile. All the collected data will be analyzed and plotted in different graphs to show the tendency of the users' performance. The users can access these results by login into the web server and reading from any device platform from anywhere. To sum up, Healthmoni provides users with a centralized health profile that automatically tracks and manages their health records and analyzes their health performance. In addition, our product has a satisfactory measurement result in step counting, heartbeat, and blood pressure.

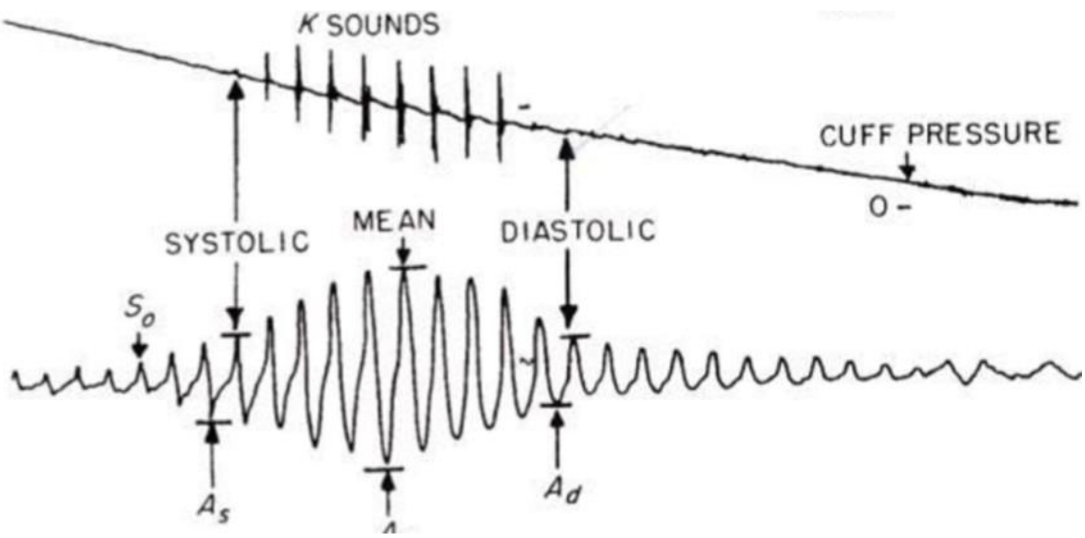
Section 3: Objectives

- Automatic data management
- Result displaying
- Accurate measurement
- Access from anywhere & device

Section 5: Technical background

Blood Pressure
The diastolic blood pressure and systolic blood pressure were measured by vibration measurement.

The oscillograph method, also known as the vibration measurement method, determines blood pressure by establishing the relationship between the systolic blood pressure Ps, diastolic blood pressure Pd, mean pressure Pm, and cuff pressure wave. The principle is that the pressure sensor in the detection device senses the initial and maximum pulsation of the brachial artery under the cuff, converts the pressure signal into an electrical signal through amplification and filtering processing, extracts the static DC component and pulsating AC component, and gives the corresponding acquisition processing to obtain the required measured. The principle diagram of blood pressure measurement by the oscilloscopic method based on the venting process is shown below.



IMG 5.1 Principle diagram of blood pressure measurement

When the cuff pressure (static pressure) is higher than the systolic pressure Ps, the artery is compressed closed, at this time, due to the impact of the proximal pulse, presents a small shock wave; When the cuff pressure is less than the systolic pressure Ps, the amplitude increases. When the cuff pressure was equal to the average pressure Pm, the arterial wall was under load, and the amplitude reached the maximum Om. When the cuff pressure was lower than the diastolic pressure (Pd), the artery lumen had fully expanded during the diastolic period, and the wall rigidity increased, so the amplitude of the wave was maintained at a small level.

Therefore, the cuff pressure corresponding to the envelope of the oscillating wave indirectly reflects the arterial blood pressure as long as the oscillating wave is continuously measured during the deflating process of the air sleeve (the oscillating wave generally presents an approximate parabolic envelope). In oscilloscopic measurement, blood pressure is obtained by identifying the feature points in the bell-shaped envelope formed by the pulse wave. At present, there are two main methods: Method 1, the fixed ratio algorithm proposed by Geddes.

Firstly, the vertex Am of the bell-shaped envelope of the pulse wave is found, and its corresponding cuff pressure Pm is the average pressure. In addition, there is a point of As on the rising edge of the envelope and a point of Ad on the falling edge of the envelope, corresponding to the systolic blood pressure Ps and diastolic blood pressure Pd, respectively.
The sizes of Os and Od can be obtained according to the following empirical formula: As/Am=0.55 (1.1) Ad/Am=0.82 (1.2).
In the actual clinical measurement, the values in the above empirical formula vary widely, and equation (1.1) is: 0.45-0.75; Formula (1.2) is: 0.69~0.89.

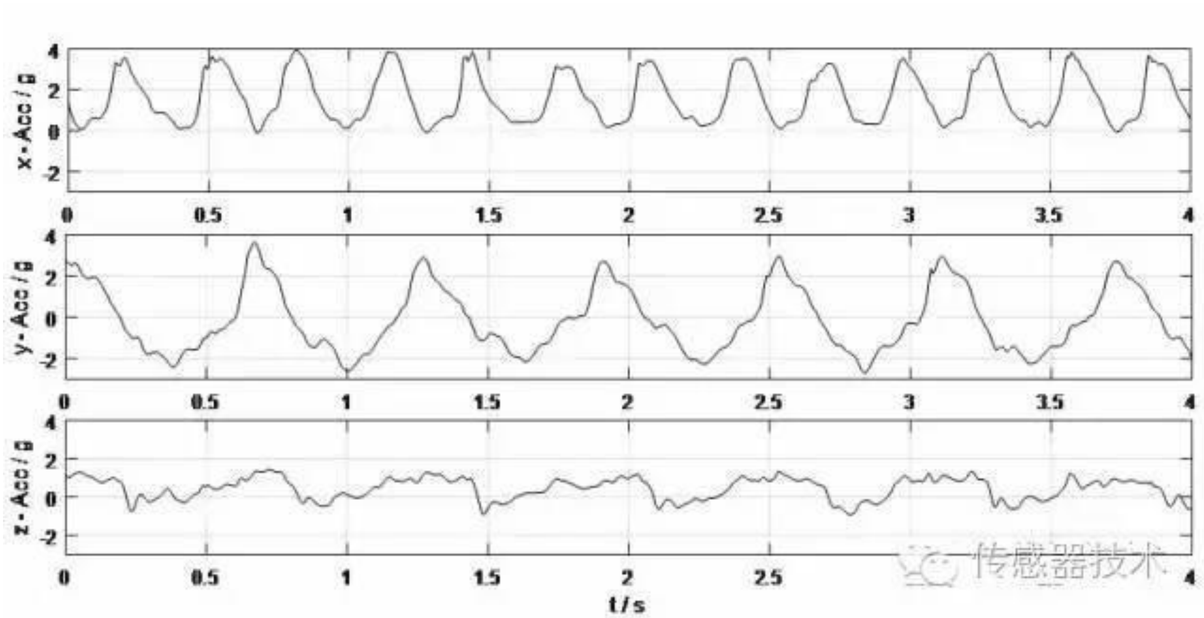
Method 2: According to the characteristics that the change steepness of Os and Od points of the envelope is large, while the change of Om is small, the pulse envelope is differentiated to obtain the corresponding systolic blood pressure Ps, diastolic blood pressure Pd and mean blood pressure Pm.
At present, method one is adopted in our design, which is to obtain systolic and diastolic blood pressure from the average pressure through the empirical formula (1.1) and (1.2).

2. steps-counter

The electronic pedometer usually has a built-in acceleration sensor (Accelerometer) and an operation unit (MCU). Through the acceleration sensor, the user's acceleration changes, and then through the MCU to estimate the number of steps taken. Take the example of an acceleration sensor placed at the wrist. When the user is walking horizontally, the acceleration at the wrist will be affected by both the gravity acceleration and the hand-throwing acceleration. In the course of walking, the acceleration caused by the hand-shaking motion changes periodically.

Peak detection class algorithm:
Here our group applied Peak detection class algorithm. First of all, we can get a sinusoidal trajectory of walking motion by calculating the vector length of the three accelerations. The second step is peak detection. We record the last vector length and direction of motion. Through the change of the vector length, we can judge the current acceleration direction and compare it with the previously saved acceleration direction. If it is the opposite, that is, it is just past the peak state, then enter the step counting logic for step counting. Otherwise, this section will be abandoned. By adding up the number of peaks, we can calculate the number of steps the user has walked.

Variation of acceleration data in the x,y,z directions:



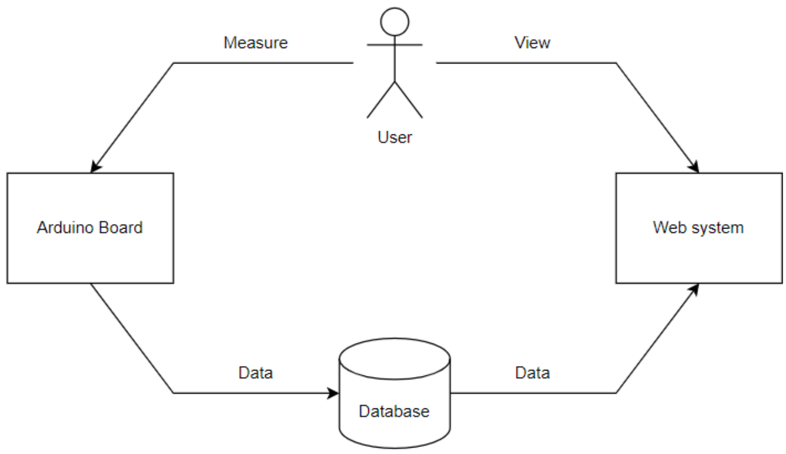
IMG 5.2 Variation of acceleration data

3. Information storage(mysql and thingspeak)
MySQL stores basic user information, such as name, age, etc.
thingspeak stores activity data, such as blood pressure and heart rate. thingspeak provides different user profiles based on other ids and can do data calculations.

Section 6: Hardware and Software Architecture

<u>6.1 Introduction of Project Design</u>

6.1.1 User-system interaction model
In the system design, our team aims to not only measure user data, also provide some backup functions to support the target user. Therefore, the system contains two important parts for the user, which are the Bady Health Measurement part (Arduino Board) and Data Review & Doctor Consultation part (Web-system).

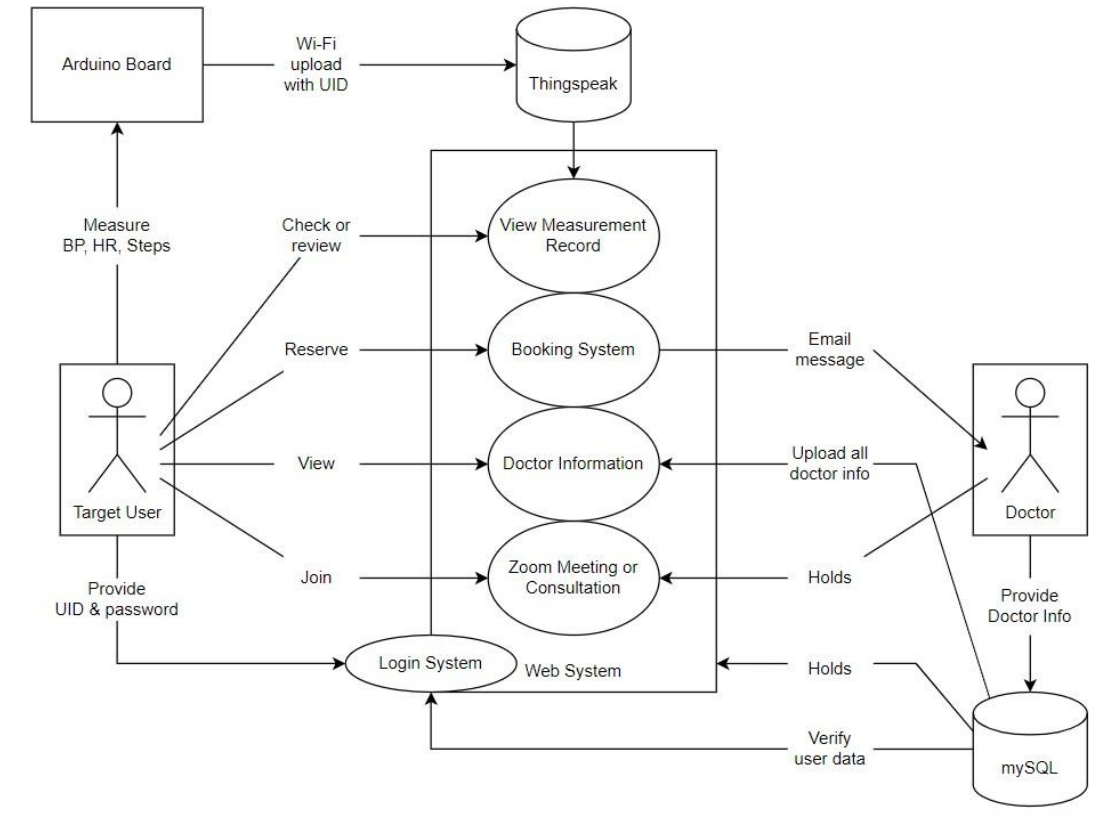


IMG 6.1: Simple user-system interaction graph

The graph of IMG 6.1 briefly shows our user-system interaction model. The model design is that user can measure their blood pressure (systolic & diastolic), heart rate, and steps in their daily life. All the measurement data will store in the Arduino board automatically. When the user pressed the Wi-Fi button, the measurement datastream <UID, Datetime, Blood Pressure systolic, Blood Pressure diastolic, Heart Rate, Steps> will be uploaded to the ThingSpeak cloud database. Our web system can request user measurement data by sending an HTML request with the user channel, when ThingSpeak catches the HTML request, it will send back the user data in a JSON file. By decoding the JSON file, we can post all data back on the user webpage and the user can review all the measurement data in our web system, and also utilize our online backup functions, like the booking system, to reserve a doctor consultation.

6.1.1 Use Cases Analysis

In the section, our team have to declare the function use case. The following image shows all architecturally-significant use cases.



IMG 6.2 Architecturally-Significant Use Cases design

6.1.1.1 Login System
Brief Description: This use case describes how a user logs into the HealthMoni System. The actors starting this use case are target users only. Doctor communicates with the user by zoom, not in our system.

6.1.1.2 Doctor Information
Brief Description: This use case allows the doctor to submit their doctor information to the system registrar, also allows users to view all doctor information on the doctor info page, and the system registrar maintains all doctor information in the database system. This includes adding, modifying, and deleting doctors from the system. The actor of this use case is the system registrar, doctors, and users.

6.1.1.3 Booking System
Brief Description: This use case allows the user to book their doctor to make a health consultation on the booking page, and the system reserve time slot to the user and target doctor in our database system. The actor of this use case is doctors, and users.

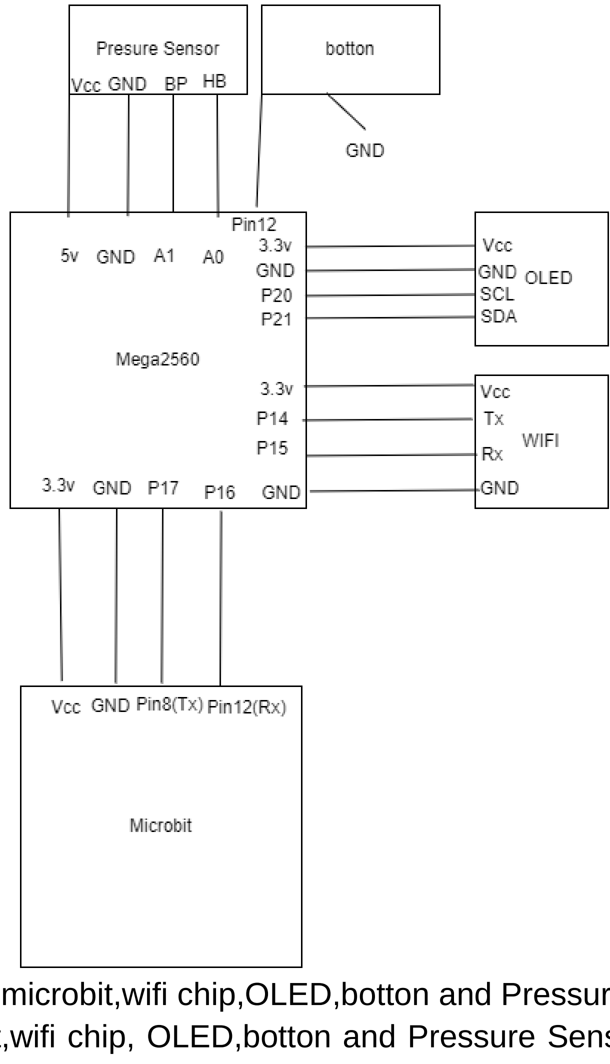
6.1.1.4 View Measurement Record
Brief Description: This use case allows the user to review or check their measurement information, also allows users to view their data is BP data, and the health state on the home and dataview page, and the system download all user

record and information from the ThingSpeak and mySQL database system. The actor of this use case is users only.

6.1.1.5 Zoom Meeting
Brief Description: This use case allows the doctor to hold their consultation room to the system, also allows users to join the doctor zoom link on the doctor info page. This includes online video consultation or voice consultation from the system. The actor of this use case is the system doctors, and users.

6.2 Give clear figures to describe the hardware design, eg. how to connect different modules or communicate

6.2.1 Hardware Connection & block diagram



We have mega2560,microbit,wifi chip,OLED,botton and Pressure Sensor.
We connect microbit,wifi chip, OLED,botton and Pressure Sensor to the mega2560 using different pins.

1. We connect Pressure Sensor to Mega2560 by A1—-BP.O/P,A0—--HB.O/P. Because we only need to read analog data from Pressure Sensor,we use analog in A0 and A1.

2. We connect Microbit to Mega2560 by Pin8—-P17,Pin12—--P16.Because we set Pin8 and Pin12 in microbit to be tx and rx, then tx should be connect to rx, rx shoule be connect to tx.As we use Serial2, we use pin16 and pin17.

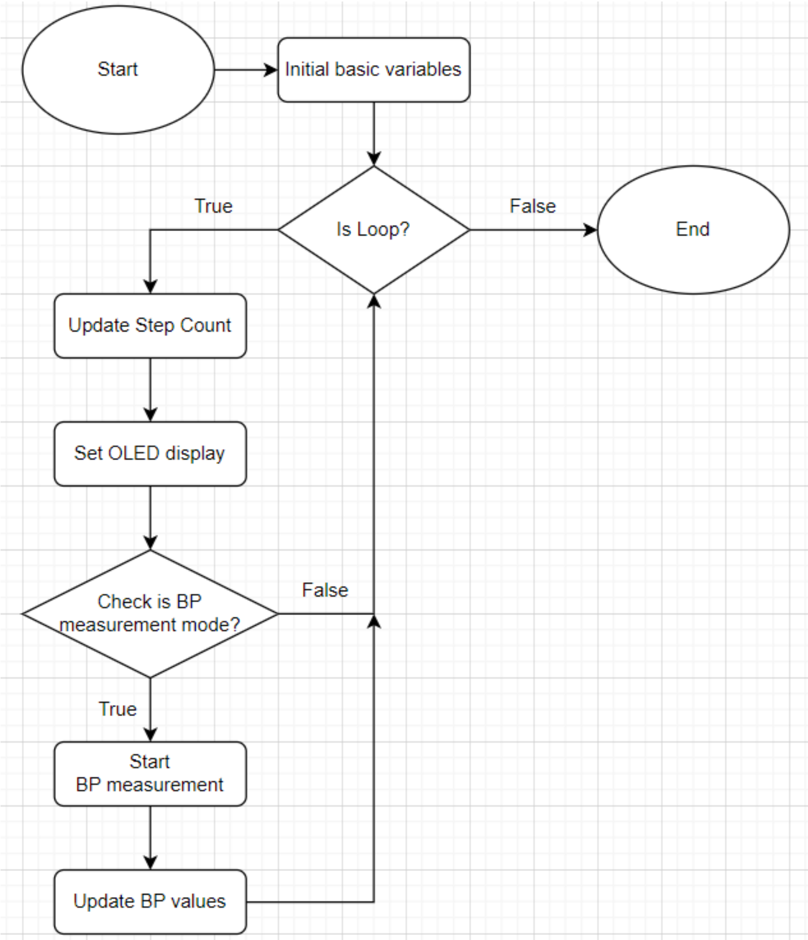
3. We connect Wifi chip to Mega2560 by tx—P14, rx—-P15 while P14 is rx and P15 is tx in Mega2560.

4. We connect OLED to Mega2560 by SCL—-P20,SDA—-P21 because P20 is SCL and P21 is SDA in Mega2560.

5. We connect button to Mega2560 by pin12.Then we can use button to connect the potential of pin12.

6.2.2 Flow Chart

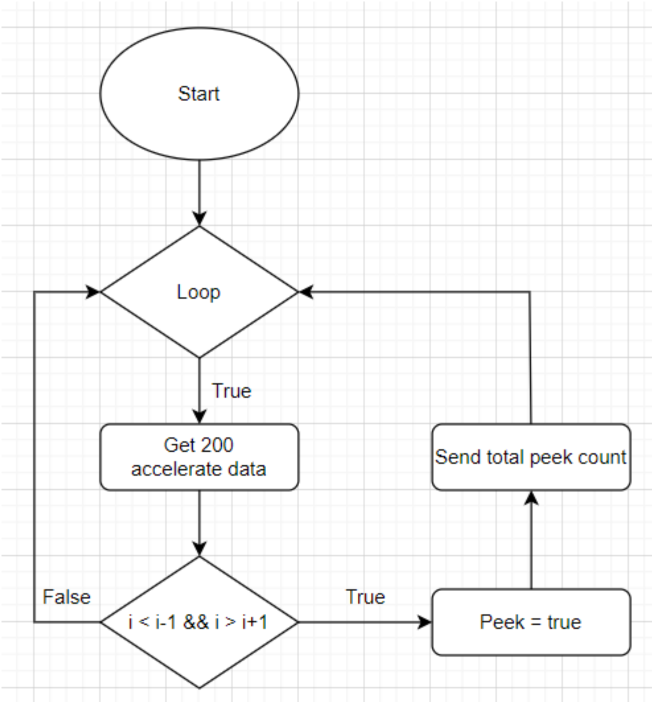
Arduino Mega 2560 Internal Program Flow:



IMG 6.3 Flow Chart of Arduino Mega 2560 internal program design

The image of IMG 6.3 shows the internal program flow of Arduino Mega 2560. This paragraph will provide a detail explanation of the program flow. First, set the global variable check=0, inflate the cuff, enter the function only when the cuff pressure is more significant than 150mmhg, and set check=1. This way, the process can continue to execute even if the cuff pressure is less than 150mmhg after the function has been run once. This way, the function, and data are not recorded until the cuff pressure exceeds 150 mmhg. This avoids recording data when the cuff pressure is too low, and the pressure is not high enough.

Microbit Internal Program Flow:



IMG 6.4: Flow Chart of Microbit internal program design

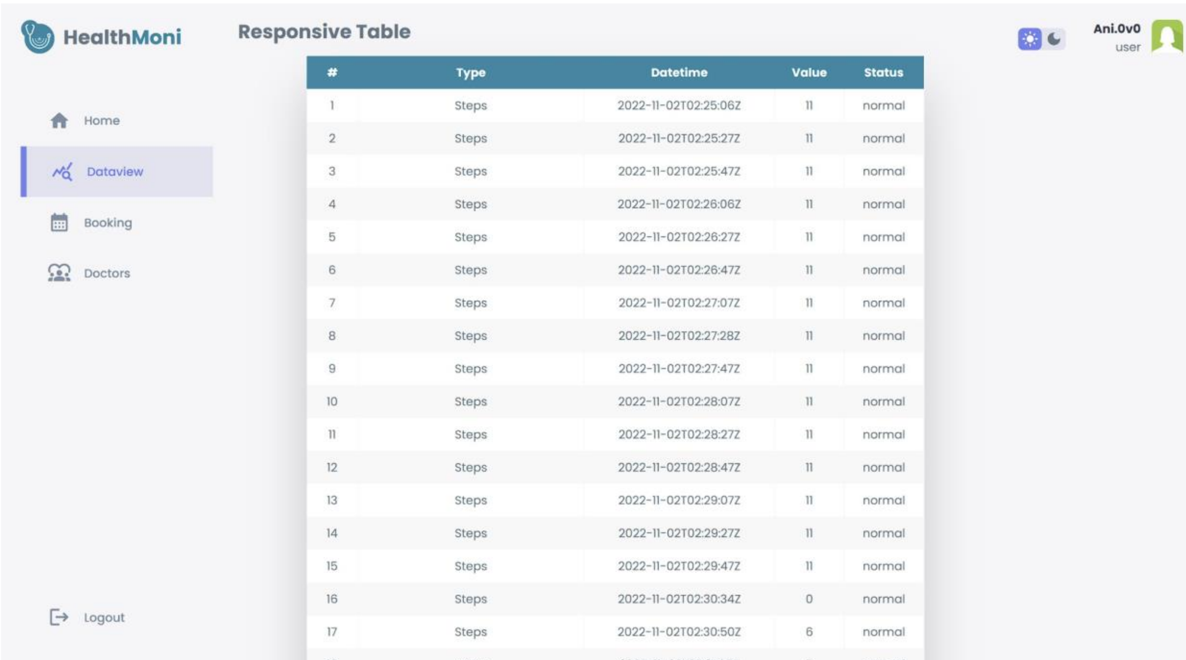
The image of IMG 6.4 shows the internal program flow of Microbit. This paragraph will provide a detail explanation of the program flow. First the accelerometer collects 200 samples each between 0.1 sec. Then traverse the sample array to find the peek index. A peak is the data that is greater than its left and right. Then traverse the peek array to check the distance between the peeks. If the distance between two peeks is greater than 2, then step = step +1. Finally, send the step by UART to the arduino.

6.3 User Interface and User Experience design (UI/UX design)

6.3.1.1 Home page

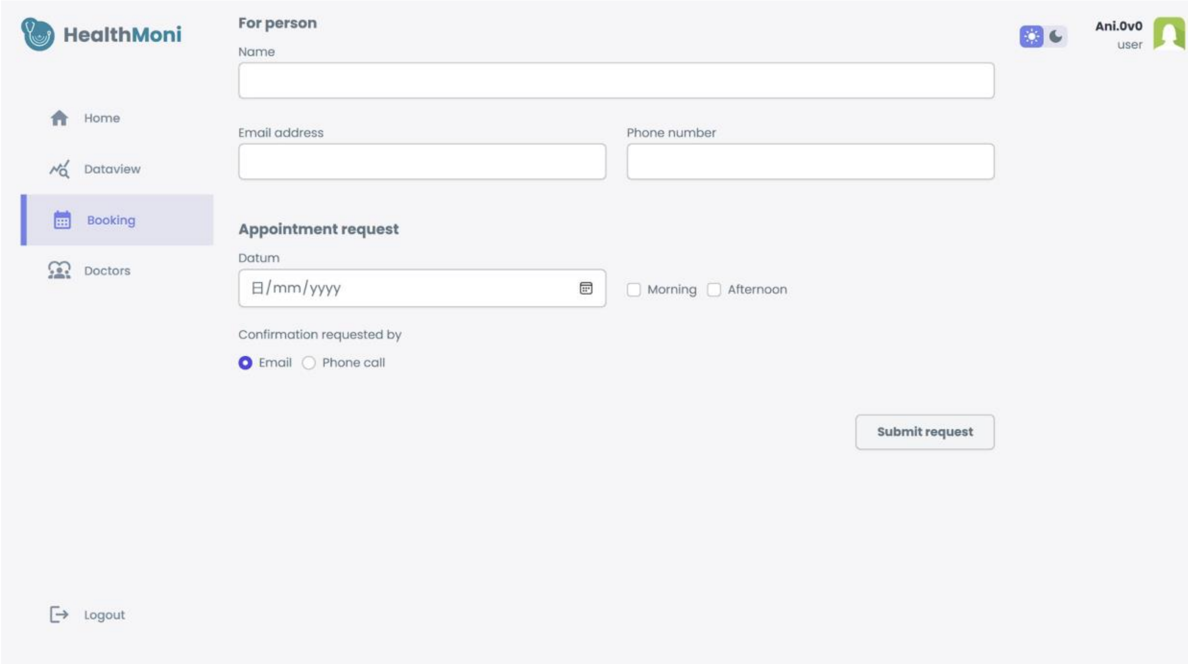
This page is designed to show all user measurement data, like Blood Pressure, Heart Rate, and Steps. The step count chart will help user to convert their step count to kcal consumption. And the user can jump to another page by clicking the sidebar menu.

6.3.1.2 Dataview page



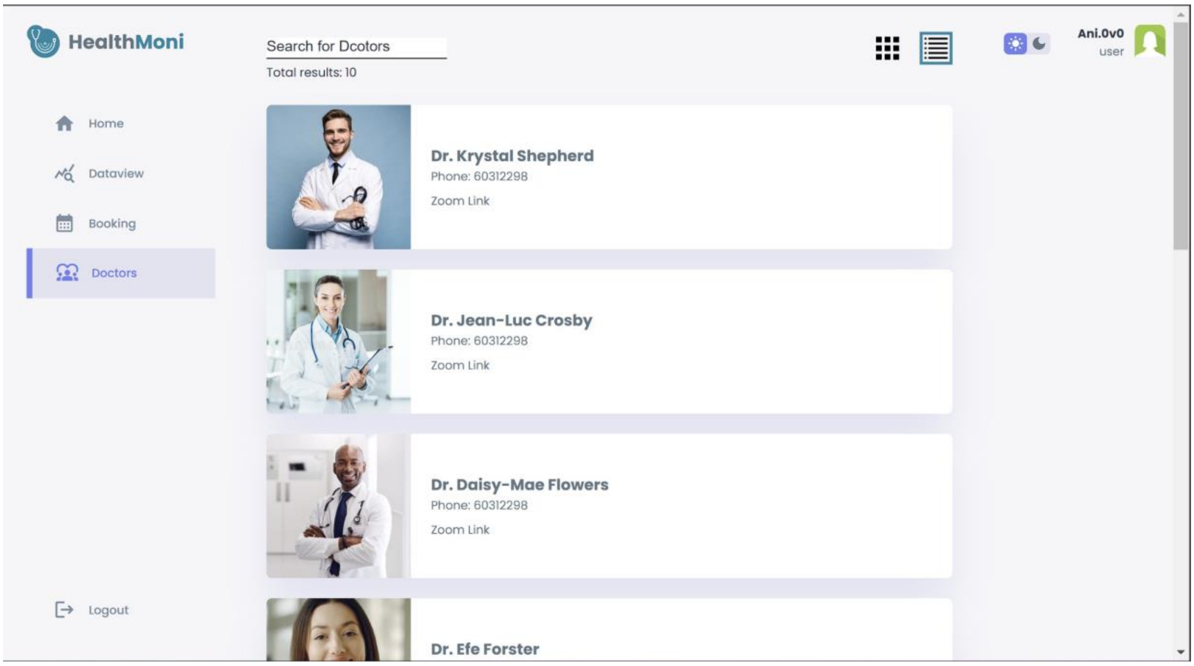
This page is designed to show all user measurement data, like Blood Pressure, Heart Rate, and Steps. The responsive table will help users identify whether their data is normal state or a dangerous state in the status column. And the user can jump to other pages by clicking the sidebar menu.

6.3.1.3 Booking page



This page is designed to let users make doctor consultations, like Blood Pressure issue, or Heart Rate issue. In the responsive form, it will take the user's name, email, and phone number. In the appointment request part, the user has to select what time slot their what to book. And the submission button helps them to submit their information to our database for doctor reservations. If the timeslot is valid, the system will book the time slot and send a successful email back to the user's email box. Also, the user can jump to another page by clicking the sidebar menu.

6.3.1.4 Doctor page



This page is designed to show all doctor information. Users can use the search bar to find the target doctor or watch the doctor's information on the doctor list. This page also provides all doctor zoom links, so the user can go into their reserved doctor’s zoom room. And the user can jump to another page by clicking the sidebar menu.

Section 8: Testing, Results and Discussion

8.1 Function Test List:

- step counter static & dynamic measure
- heartbeat counting
- blood pressure compare with existing product
- burst data upload to ThinkSpeak & passing data to arduino
- web login, display, graphing result checking
- OLED checking

8.2 Step counter static & dynamic measure

The results are highly accurate. When we first tested the step counter, we walked about 10 steps and ran about 5 times. The results are either 8 or 9 steps. These results are not bad. Then we change from 10 steps to 20 sec to fit in the working model of our product. We run the test about 10 times, each time we walk 40 steps in 20 sec.The results are 36 - 40 steps.

However, these tests are all dynamic. When we are static, the step counter still generates results. These behaviors are not expected. After investigating, we found the issues are due to the hardware problem. The accelerator records the acceleration from the gravity when it is not moving. But the number from the static recording shows that they are not the same, which means even though the accelerator is not moving, it still generates small waves. This kind of behavior corrupts our results.

Therefore, after we collected the gravity acceleration range, we set the range 1.1g to 0.9g as static range. Any data within this range will not be treated as a peek. Although this setting may affect the performance, the chance of moving acceleration fall in this range is quite small by comparison with the chance of static, which means the effect of this change should not be huge. After testing, the result shows 0 steps when the user is not moving. Then we tried mixed mode, which the users will change from static to dynamic. The performance is slightly degraded from 36 - 40 steps to 34 - 40 steps.

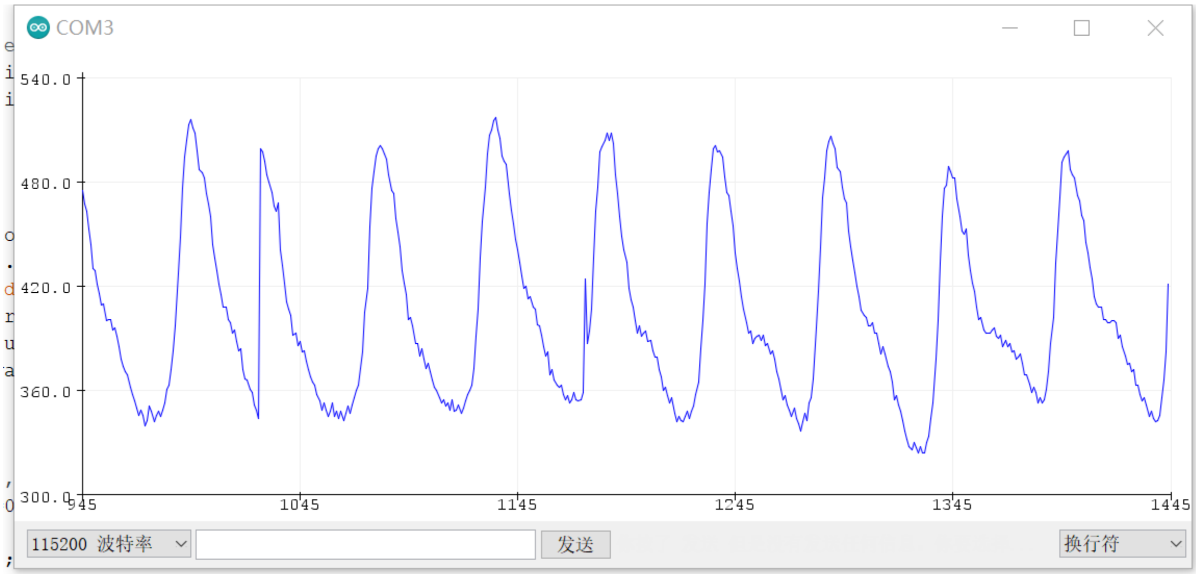
- The dynamic accuracy is about 90% after running the final test for 10 times.
- The static accuracy is 100%.
- The static & dynamic is about 87.5% after running the final test for 10 times.

Overall the results are satisfactory.

8.3 Heartbeat counting

We did a lot of testing with the cuff.
In the first test, the data was too large. Figures of 110,120 bpm are often found. We found that the reason was that the device recorded high pressure during and at the beginning of the cuff inflation, so we restricted the heartbeat data to a more normal

range. In the second test, We get a variety of things: a heartbeat of 85,90,83,93 BMP, and then we do a heartbeat test with the smartwatch, and we get 90,91,88,93 BMP. It can be seen that although the data are close, there may be some data smaller than actual data . We checked and found that it was a limitation of the code. The principle of the code is to collect the crest number n in a short period, t, to calculate the heartbeat. In some cases, however, the crest wasn't recorded, as seen in the figure below.



So we changed the code to record the maximum value that frequently occurs over some time so that multiple peaks were recorded.

8.4 blood pressure compare with existing product

We conducted several tests and compared the results with Watsons, a commercial blood pressure monitor provided by the school.
In the first test, we found that the amount of data collected was too large, and there were duplicates. The reason is that pointer used to judge the wave crest moved too close after the judgment, resulting in the repeated recording of the same wave crest. The code was then improved to move back 20 data volumes for every ridge collected to avoid repeated collection. On the second test, we found it occasionally challenging to manage systolic or diastolic blood pressure. The reason is: we used the coefficient method, and we may not see the amplitude of data exactly 0.85Am or 0.45 Am, so we modified the search function to find the data within the range of ±5. In the third test, we found that the result was significantly different from that of the commercial sphygmomanometer. After checking all the amplitudes of the peaks collected, it was found that the reason might be an inaccurate coefficient, so the original coefficient of 0.85 and 0.45 was modified to adapt to the blood pressure of different people of different ages, genders, and so on. In testing, we use the tester (ourselves) as a template to make changes. Due to the limited performance of the hardware, we cannot automatically calculate the different parameters of each person and then

bring them into the calculation. If the dynamic memory of hardware devices can be improved in the future, we can first collect data for some time to get the general parameters and then measure blood pressure.

8.5 Burst data upload to ThinkSpeak & passing data to arduino

Burst data upload to ThinkSpeak :

When we test the ThingSpeak channel, we upload some random data to it. We want to upload 4 data fields at different times. Since the step counter updates its data constantly but the heartbeat and blood pressure doesn't. The upload fails if the time between the last upload is less than 20 sec. We found this limitation after we failed to upload data to the channel. Then we ran to ThingSpeak to check about the specification. Based on this limitation, the step counter will collect the result every 20 sec to avoid update failures. And the heart pressurement pauses for 20 sec after uploading the results in order to ensure that after it will not collide with the step counter.
After changing the time period, the update to ThingSpeak worked fine.

8.6 Passing data from micro bit to arduino:

We want to pass the step counter result to the arduino.
We first try to use the analog signal to pass the data. But we fail due to the step counter result being too low and easily inference by nature noise. Then we swap to URAT. In this way we can be sure the data is not inference by others. We set up the tx , rx pin and braute for both. Then begin the transfer.
The results are working as expected.

8.7 Web login, display, graphing result, zoom checking

We test the jumping between the pages, the graph plotted and the format of the UI. Everything worked as expected. The layout of the UI is corrected. And the analyzed results on the graph are corrected. The login function is working and correctly jumping between pages. Once logged out, the page can go back to the login page. The doctor booking systems can open the zoom link for the doctor booked.
Overall, the performance of the web page is satisfactory.

8.8 OLED

We connected power to the OLED on the board and tested it.
Test one, the OLED, occasionally displays garbled codes. Testing the code revealed that it was probably because there was an error in the data that needed to be said, and the data in front of it needed to be modified.
Test 2, the OLED occasionally only partially displayed. The reason could be that more code was written or the running time was too long. The solution, just re-plug the UBS cable.

Section 9: Conclusion and Suggested Improvement

Our system is divided into a hardware part and a UI part.
The hardware part measures the user's blood pressure and heart rate and displays it on OLED. And at the same time can test the user's step number and be used as a pedometer. Then, you can use buttons to transfer the data over wifi to the databases Thingspeak and MySQL, and from thingspeak, you can transfer the data to the UI page we designed for users and doctors to view.

On the UI page, users can view their blood pressure, heartbeat, step count, and other data measured in the past period, as well as the information of multiple doctors, and can make an appointment and communicate online through zoom. Performance: After testing, the system's performance is good, the measured blood pressure and heartbeat data are more accurate, and the accuracy of commercial equipment is less different. The web design is reasonable; the picture is simple, the function is comprehensive, and the guidance is appropriate, in line with user-friendliness.

The impact of our design on users:
Our method can help users quickly detect their blood pressure and heartbeat and enable users to communicate with doctors timely and conveniently so that they can get professional help in the early stage of possible physical abnormalities, or monitor their physical conditions, to avoid the sudden deterioration of existing conditions. Second, different users can access the system using their ids. Our system will automatically match the corresponding data.

Improvement:
Our approach is based on an Arduino board, Mega board, and other hardware, which is large and not easy to carry. In the future, it can be upgraded to use more minor chips to provide users with a more convenient and fast experience. Our equipment is underperforming, and the code is not of high quality. In the future, the code can be optimized to first process the data for some time to obtain different parameters required by other people and then put it into the function for the calculation to reduce the error.

Self-reflection
Our system only has the role of a sphygmomanometer and heartbeat measurement. If we have the opportunity, we should add more sensors to provide more functions, such as temperature, electrocardiogram, and so on, to better monitor human health.

Section 10: References

1. Blood Pressure measurement method (Section5)
S. Daochai, W. Sroykham, Y. Kajornpredanon and C. Apaiwongse, "Non-invasive blood pressure measurement: Auscultatory method versus oscillometric method," The 4th 2011 Biomedical Engineering International Conference, 2012, pp. 221-224, doi: 10.1109/BMEiCon.2012.6172056.

2. Accelerometer step principle
https://www.sohu.com/a/115643406_468626#:~:text=%E7%94%B5%E5%AD%90%E8%AE%A1%E6%AD%A5%E5%99%A8%E9%80%9A%E5%B8%B8,%E6%8A%97%E5%B9%B2%E6%89%B0%E8%83%BD%E5%8A%9B%E6%9B%B4%E5%BC%BA%E3%80%82（online popular science Webpage）