Svasth healthcare
always better

November3, 2014

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
V5A 1S6

Re: ENSC 305/440 Project Proposal for *Comfort Mat System (CMS)*

Dear Dr. Rawicz,

In regards to the course requirements of ENSC 305W/440W, enclosed to this letter is *Svasth Healthcare's* design specification for the product - *Comfort Mat System (CMS)*. We are designing and implementing a seat mattress, which can detect unfavorable conditions leading up to, advent of pressure sores and prevent them from developing.

The following documentation provides a detailed description of all technical aspects of the product to achieve the functional requirements. This document also provides the design justifications of *CMS f*or its various stages of development. To affirm the functionality of our product, test plans are also enclosed in the appendix.

Svasth Healthcare consists of four talented engineers: Wei Lu, Di Luo, Henson Truong, and JabarJung Sandhu. If you have any questions, or concerns regarding the functional specification or product, then please feel free to contact JabarJung Sandhu atjss19@sfu.ca or 604.351.4927.

Sincerely,

Jabarjung Sandhu

JabarJung Sandhu
Svasth Healthcare CEO

Enclosure: Design Specifications for Comfort Mat System

Svasth healthcare

always better

Design Specification for Comfort Mat System (CMS)

Project team	**JabarJung Sandhu (CEO)** **Wei Lu (CFO)** **Di Luo (COO)** **Henson Truong (CTO)**
Contact	JabarJung Sandhu
Submitted to	Dr. Andrew Rawicz - ENSC 440W Dr. Steve Whitmore - ENSC 305W School of Engineering Science Simon Fraser University
Issue date	November3, 2014
Revision	1.0

Executive Summary

This document provides the details for the design specifications for the entire Comfort Mat System. The purpose of this document is to outline and describe in detail the parts and functionalities of Svasth Healthcare's CMS. We have also emphasized and added more detailed on testing part.

The proof-of-concept model of the CMS is divided into five components: seat cushion with fabric cover, input signal unit, output signal unit, data processing and user interface unit. Each of these components has its own section describing the technical details which fulfill the functional requirements of each component. Justifications are provided for design approaches and the motives for choosing specific hardware components. Our design choices are also included for some of the components.

The final section of this document explains the testing that will be done for each component of the design separately. A detailed testing check list is also included in the appendix section for CMS's testing reference and record. This testing will affirm that each subsystem is designed and implemented to be compatible with the other components onboard.

The schedule of design process is pretty tight at this point, but we believe we are able to finalize our product on time.

Table of contents

Design Specification for Comfort Mat System (CMS) 2
Executive Summary 2
Table of contents 3
List of figures 4
List of tables 5
Glossary 5
1. Introduction 6
  1.1 Scope 6
  1.2 Intended Audience 6
  1.3 Background 7
2. System Specifications 7
  2.1 High Level System Design 8
3. Seat Enclosure 8
  3.1 Cushion Cover 8
  3.2 Pressure relief mat 8
4. Input Signal Unit 10
  4.1 Temperature and Humidity Sensor 10
  4.2 Force Sensitive Resistor (FSR) 11
  4.3 Sampling Frequency 13
  4.4 Battery Power 13
5. Output Signal Unit 14
  5.1 Vibration Motor 14
  5.2 Bluetooth Signal (BLE mini) 15
6. Data Processing and User Interface Unit 16
  6.1 Computer (Matlab) 16
  6.2 Smartphone 17

6.2.1 Introduction to the app 17
6.2.2 Communication with the app 18
6.2.2.1 Android 18
6.2.2.2 iOS 19
6.2.3 Presentation of the data 20
7. Test Plan 20
9 Bibliographies 21
10. Appendix: Detailed Test Plans 23

List of figures

Figure 1: System overview of Comfort Mat System 7
Figure 2.Force that acts perpendicular to the surface causes pressure and force that acts parallel to the surface causes shear[17] 9
Figure 3. Egg-tray shape cushion[15] 10
Figure 4.Pin connection of DHT22[14] 10
Figure 5. Communication between sensor and microcontroller[14] 11
Figure 6.Mechanical data of FSR[12] 11
Figure 7.Vout curves of using different resistors [12] 12
Figure 8. The structure of FSR matrix of CMS 12
Figure 9.Configuration of combining dc motor, L293D and Arduino board 14
Figure 10.Pin configuration of L293D motor controller[9] 14
Figure 11.Circuit diagram of BLE mini board [6] 15
Figure 12. Pressure mapping on Matlab[8] 16
Figure 13. Process of introducing phone application 17

List of tables

Table 1. Voltage usage of different components 13
Table 2.Comparison of different types of AA batteries [3] 13
Table 3. Testing plan for hardware........ 23
Table 4. Testing plan for software........ 24
Table 5. Testing plan for Matlab........ 25

Glossary

API - Application programming interface
APP - Application
BLE - Bluetooth 4.0 Low Energy
CMS - Comfort Mat System
DC - Direct current
DPI - Dynamic Pressure Images
FSR - Force Sensitive Resistor
IC - Integrated circuit
iOS - iPhone OS
LCD - Liquid-crystal display
PIN - Personal identification number
SDK - Software Development Kit
USB - Universal Serial Bus
COM - Communication Port
PC - Personal Computer
TX/RX - Transmit/Receive

1. Introduction

Svasth Healthcare is designing and developing Comfort Mat System (CMS), which will be able to monitor the circumstances responsible for instigating bedsores by collecting and analyzing data from different sensors (pressure, temperature, and humidity sensors) with the help of a smartphone application which will display the results in a graphical user interface in real time. Another feature that CMS exhibits is that once the analyzed data meets a preset threshold value, the system can automatically turn on different vibration motors to enhance the blood circulation in the affected area. It will then activate a reminder alerting the user to reposition their body or the person who is caretaker for that person. Comfort Mat System will be integrated with Android and iOS apps, which will allow user to monitor and control the system with a smartphone device.

1.1 Scope

This document illustrates the design approach leading to the selection of the various hardware and software components for Svasth Healthcare's Comfort Mat System. The functionalities of the proposed system are justified with their explicit components, their design and their operation details. This document also describes in detail about the technical aspects of the system, which includes the Arduino board, BLE mini Bluetooth shield, Android smartphone application, force sensitive resistors, vibration motor, temperature and humidity sensor, as well as the battery that supplies the power. Also this document includes but is not limited to analysis of individual component performance, hardware design justification, communication protocols, and integration of subsystems. Lastly, the document proposes a system test plan for the Svasth Healthcare's Comfort Mat System.

1.2 Intended Audience

The aim of this document is to highlight the design approach to fulfill the proposed features and requirements of Svasth Healthcare's Comfort Mat System. It will be used as user documentation and for test planning purposes. This document will also be used to justify any design decisions as well as to serve as a template for future modifications, if any issues are encountered during the testing and validation phases. All of the team members will use this document as a guide to progress on the CMS project and its relevant tasks.

1.3 Background

Bedsores are injuries to the skin and underlying tissues near bones when exposed to unfavorable conditions. Nowadays, an increasing number of hospital patients or disabled persons who are bound to wheelchairs or bedsores suffer from it. According to Agency for Healthcare Research and Quality report, the numbers of patients who suffer from bedsores have increased by more than 80% over the last two decades [5]. This number is expected to expand in the future. Our product *CMS* mainly focus on helping those who have limited mobility to reduce their chances of developing bedsores. People who sit on chairs for long periods of time like those who work in offices will be also benefited from CMS.

2. System Specifications

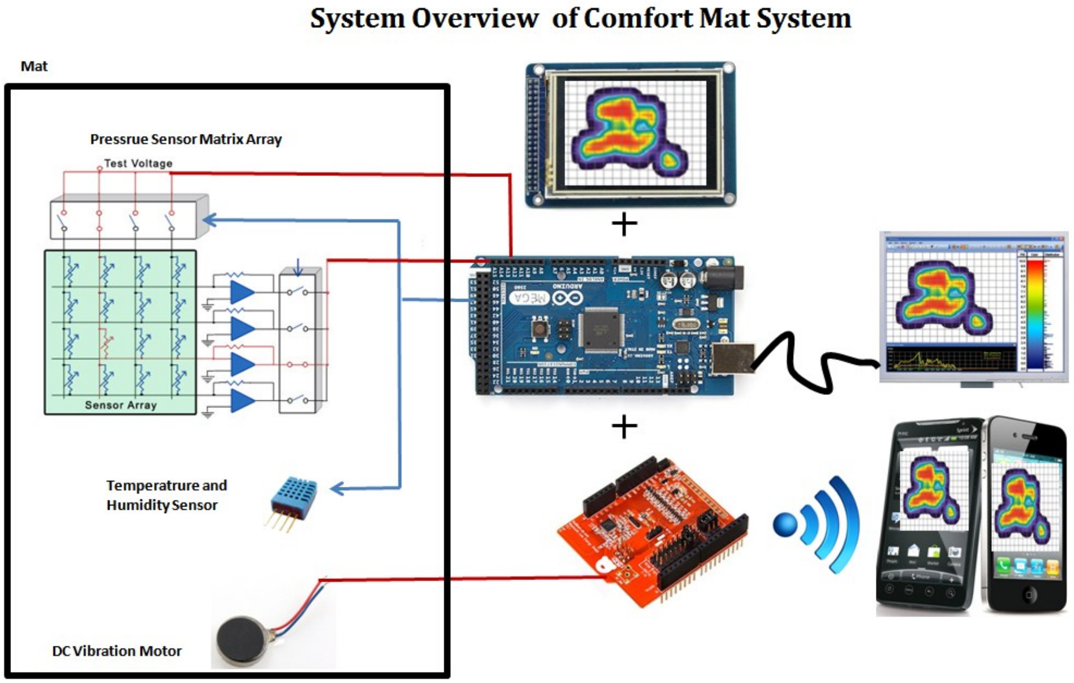

System Overview of Comfort Mat System
Mat
Pressrue Sensor Matrix Array
Test Voltage
Sensor Array
+
+
Temperatrure and Humidity Sensor
DC Vibration Motor

Figure 1: System overview of Comfort Mat System

CMS consists of a pressure sensitive mat, a smartphone, and a microcontroller (Fig.1). When user is sitting on the mat, the mat outputs the pressure, temperature, and humidity sensors' data to the microcontroller. The microcontroller processes the data and sends the data to the attached device for the information to be displayed. There are three different devices we can use to display the data – a computer, a small LCD screen, or a smartphone. We will start by getting the computer first; move on to the small LCD display, then finally to the smartphone once we know everything is functionally working. The vibration motors can be set to vibrate whenever the reminder goes off or whenever the user sets it off.

2.1 High Level System Design

The primal function of CMS is to detect long lasting high pressure points and inform the user in real time when it is detected. An 8 by 8 force sensitive resistor (FSR) matrix is used in order to collect a detailed data set from user and access pressure distribution for further analysis. A temperature and humidity sensor is installed under bamboo fabric topper to monitor environmental condition of user’s current location, because bedsores are more likely to develop under a humid and warm environment. After receiving all sensors' data, the CMS will send the data to either a computer or a wireless device based on user's preference: via a USB cable or a Bluetooth signal. A BLE mini board is the key component for transmitting and receiving the Bluetooth signal. The purpose of transferring that data is to turn them from numeric results in to a more direct graphical result: pressure mapping. If prolonged pressure is confirmed by the CMS, the colors on the pressure mapping will change gradually on GUI and an alarm will also be triggered to inform user or caregiver immediately to reposition themselves on the seat. Besides pressure mapping on the GUI, the user can also control several vibration motors at will in order to improve his/her blood circulation on a desired area.

3. Seat Enclosure

The seat enclosure for the CMS encloses all the sensors (including the temperature and humidity sensor, as well as the FSR array) and is the part where the user sits on. This is an important part of the system because it needs to be able to protect all the sensors it encloses, while being able to withstand pressure from being sat on. The seat enclosure is composed of two parts: the cushion cover and the pressure relief mat.

3.1 Cushion Cover

The cushion cover will be a breathable fabric cover wrapping over the pressure relief mat. For proof of concept, we will use a durable type of fabric which can stretch to a reasonable amount (to withstand the shearing force being applied to the cover when being sat on).

3.2 Pressure relief mat

The pressure relief mat is a customized mat that is placed on top of the pressure sensor matrix array. The mat is designed for comfortable sitting along with pressure relief, temperature & humidity reduction, and improved air circulation. As a result, the pressure relief mat can decrease the likelihood of pressure sores developing.

Pressure is formed by two individual forces: Normal force and Shear force. The Normal force is the compression between bone and skin when there is an external force applied. The Shear force is when the skin and its underlying tissue move in different directions. Pressure has a proportional relationship with the normal force and shear force individually, which is shown in Figure 2. The friction between skin and mat, temperature, and humidity play important roles in pressure sore formation.

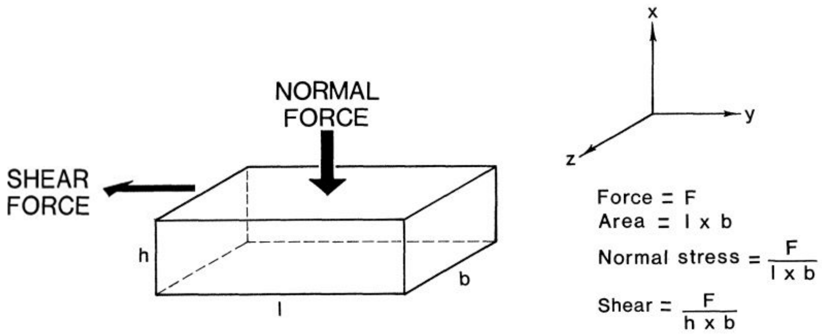


Figure 2.Force that acts perpendicular to the surface causes pressure and force that acts parallel to the surface causes shear[17]

To conclude, the material used for the pressure relief mat should be one of the most critical parts for the CMS. Sweating is more likely to occur as the temperature rises. When someone is sweating, moisture accumulation will cause the deformation of soft tissues [2]. Therefore we picked a material which is porous so it is able to dry quickly when the sweat appears and to distribute the pressure evenly. Some materials such as gel have good heat conductivity and heat capacity which can dissipate heat on skin quickly but it is not comfortable and easy to customize.

Because of those factors we have decided to use foam to make the mat and bamboo fiber as the mat's exterior cover. The advantages of using foam are that it is lightweight, easy to customize, flexible, and waterproof. The reason why we use bamboo fiber as cover is because it has a higher thermal conductivity (0.121W/m-K) rate than foam (0.03W/m-K) at room temperature [13]. We plan to customize shape of the cushion to egg-tray shape (Figure 3) because this shape can ease pressure by reducing the area that is in touch with the skin while at the same time having circulation of air.

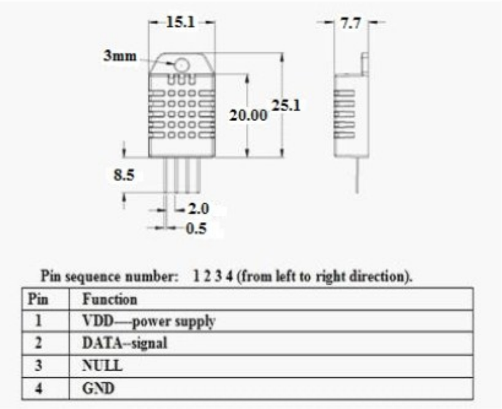
Figure 3. Egg-tray shape cushion[15]

4. Input Signal Unit

4.1 Temperature and Humidity Sensor

The temperature and humidity sensor can continuously monitor temperature and humidity of the environment. Many types of sensors are available in the market and RHT22 is one of them. An advantage of this sensor is that it comes in a plastic enclosure which can protect the sensor in case of damage by body pressure. It is good for 0-100% humidity readings with 2-5% accuracy and -40° to 80°C temperature readings with ±0.5°C accuracy.

The RHT22 has 4 pins as depicted below in figure 4. The communication process between the sensor and the microcontroller is shown in figure 5. We will be using a library that is already present for the Arduino to communicate with this sensor.



Pin sequence number: 1 2 3 4 (from left to right direction).

Pin	Function
1	VDD—power supply
2	DATA--signal
3	NULL
4	GND

Figure 4.Pin connection of DHT22[14]

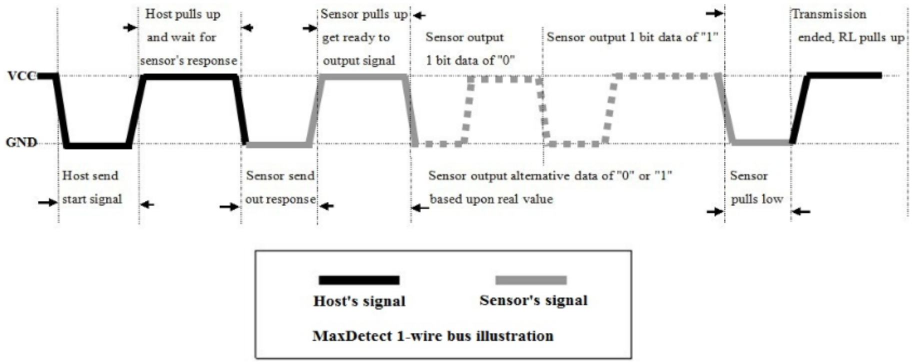


Figure 5. Communication between sensor and microcontroller[14]

4.2 Force Sensitive Resistor (FSR)

A FSR is a variable resistor that changes resistance when pressure is applied to it. We will be using the FSR 402 manufactured by Interlink for our FSR array to sense body pressure applied to these sensors because of its price advantage and excellent performance (i.e. force sensitivity range and force resolution)[12]. Some mechanical data of FSR is shown figure.6. The FSR device is tied to a measuring resistor in a voltage divider and the output is described by the following equation:

$$V_{\text{out}} = \frac{R_m}{R_m + R_{FSR}} V_+$$

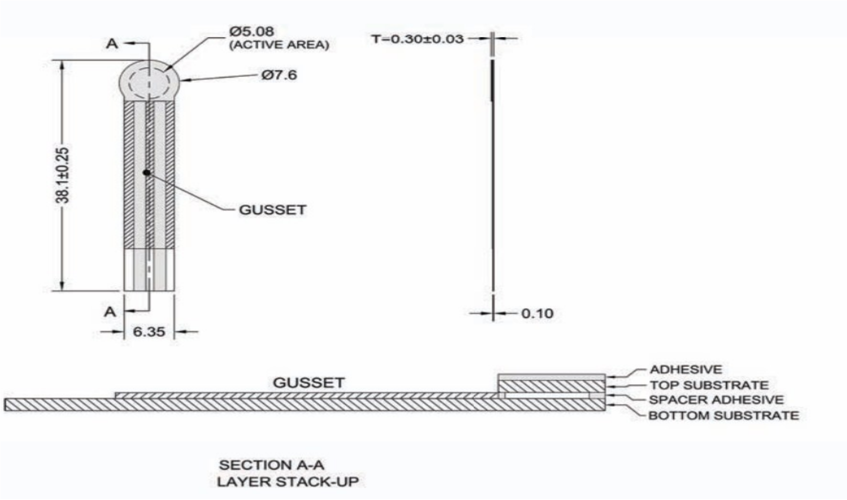


Figure 6.Mechanical data of FSR[12]

The $\mathrm{V_{out}}$ curves are shown on the graph show in figure.7
for the FSR in voltage divider configuration. As force increases, resistance decreases in the FSR. $\mathrm{R_m}$ is the resistor that adjusts the sensitivity of our FSR. The value of $\mathrm{R_m}$that we will be using is 10k. The $\mathrm{V_{out}}$ signal will then be measured by an analog input signal on the Arduino board.

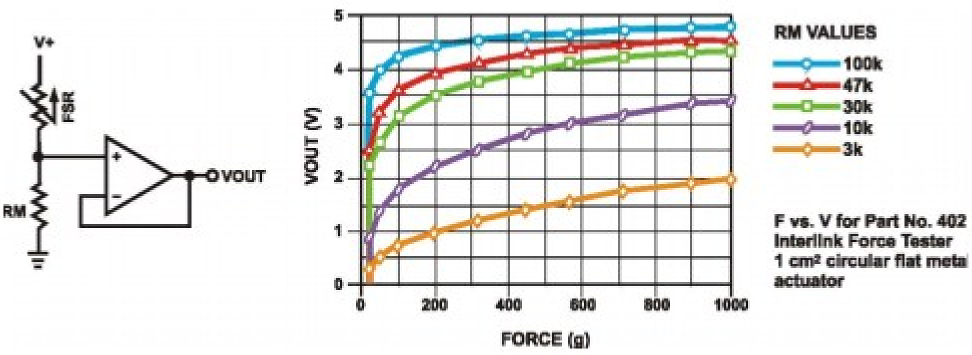


Figure 7.V_{out} curves of using different resistors [12]

A pressure sensitive mat can be built with out of these FSRs using a matrix of pressure sensors (8 x 8 in our case). The circuit diagram of the FSR matrix is shown in the figure below (Fig.8). A FSR reading can be obtained by enabling a single row and column, thus singling out a single FSR.

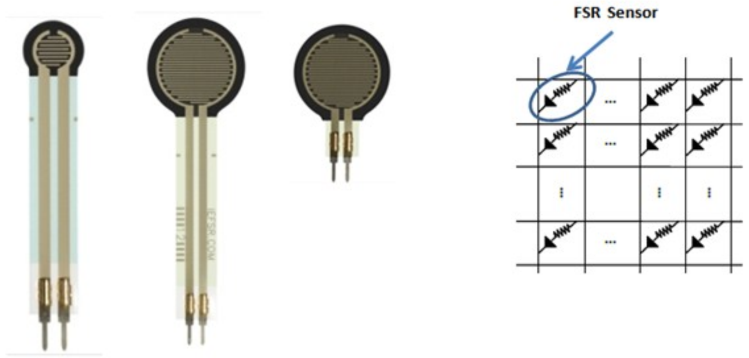


Figure 8. The structure of FSR matrix of CMS

4.3 Sampling Frequency

The sampling frequency of the CMS we will be using is 100 Hz after doing some stability testing on both the phone application and on the Arduino board. There are some reasons for our choice. First, we have to use the "millis" and "delay" functions from Arduino board to build a timer for our CMS. The function of "millis" is to return the number of milliseconds that has passed since the Arduino board has begun running the current program. The delay function pauses the program for a certain amount of time (in milliseconds) specified as parameter. The time unit for both functions is in millisecond. Second, in order to provide an explicit data set for further processing, we need to guarantee that the CMS collects a persistent data, while the speed of the sampling rate should not freeze the data processing of phone app. We have tried several different sampling rates and see their performance to determine the most appropriate rate. When the frequencies are less than 100 Hz, some delay or time lag happen frequently on our wireless device (IPad mini 2 with Apple A7 chipset). That is the reason why we choose 100 Hz as our sampling rate.

4.4 Battery Power

8 AA Batteries will be enough to supply power to our major components: CMS and DC motors (Table.1).The average voltage of each AA battery is 1.5 Volts (Table.2), so the total voltage should be 12 Volts. The reason why we chose to use batteries as our power source instead of power from outlet is because of safety, power consumption and portability advantage. We strongly recommend users to use rechargeable batteries to avoid having to keep purchasing batteries over and over again.

Component name	Voltage (Volts)
Arduino mega 2560	5
BLE mini	5
DC motors	12

Table 1. Voltage usage of different components

Type	Capacity under 50 mA constant drain (mAh)	Nominal Voltage (Volts)	Max. energy at nominal voltage and 50 mA drain (Wh)	Rechargeable
Zinc-carbon	400-1700	1.5	2.55	No
Alkaline	1800-2600	1.5	3.90	Some
Li-FeS_2	2700-3400	1.5	5.10	No
NiCd	600-1000	1.2	1.20	Yes
NiMH	600-2850	1.2	3.42	Yes
NiZn	1500-1800	1.65	2.97	Yes

Table 2.Comparison of different types of AA batteries [3]

5. Output Signal Unit

5.1 Vibration Motor

Vibration motors will be placed inside the CMS. The DC motors only improve user's blood circulation, but can also notify the user that his/her current pressure value at a specific point on the mat has exceeded a preset threshold. Several motors are located under fabric topper in order to contact skin more directly. There are two ways of operating the motors: turning the motors on by a phone application or triggered automatically whenever a preset limit is reached on the Arduino board. To fully control the motors, a few L293D Integrated circuits will need to work together with Arduino board to supply to correct amount of power to drive those motors (Fig.9).

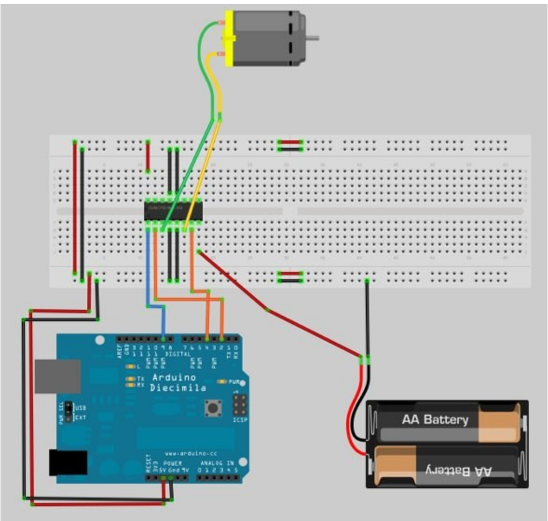


Figure 9.Configuration of combining dc motor, L293D and Arduino board

An explicit pin configuration of an IC is shown below (Fig.10). After programming outputs of digital pins on Arduino board, motor should now change accordingly.

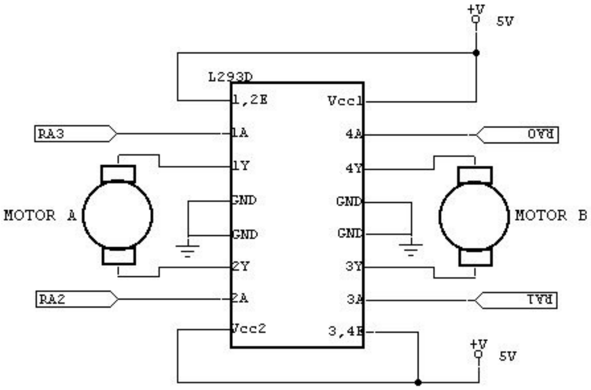


Figure 10.Pin configuration of L293D motor controller[9]

5.2 Bluetooth Signal (BLE mini)

The Bluetooth 4.0 Low Energy (BLE) development board requires only a serial port for communication. It is very easy to integrate this Bluetooth development board with Arduino mega board, but some preparations are needed. Let us first take a look at BLE mini's circuit (Fig.11). The power of the BLE mini can be supplied by a micro USB connection or the Arduino board itself. In order to transmit and receive the Bluetooth signal successfully, the TX and RX pins from BLE must connect to PIN18 and PIN19 on Arduino board as BLE library defines. These two pins can also be referred to as serial1. Besides the hardware setup, some codes that define what types of data are sending should be uploaded on the Arduino board. "Serial1.print" is the key function here, as it stores the key data to the serial port. Next, the saved data is then ready for retrieval from a wireless device. The reason why we are using "print" instead of "write" function is because the write function can only send one byte character to the serial port, but we need an integer in our case.

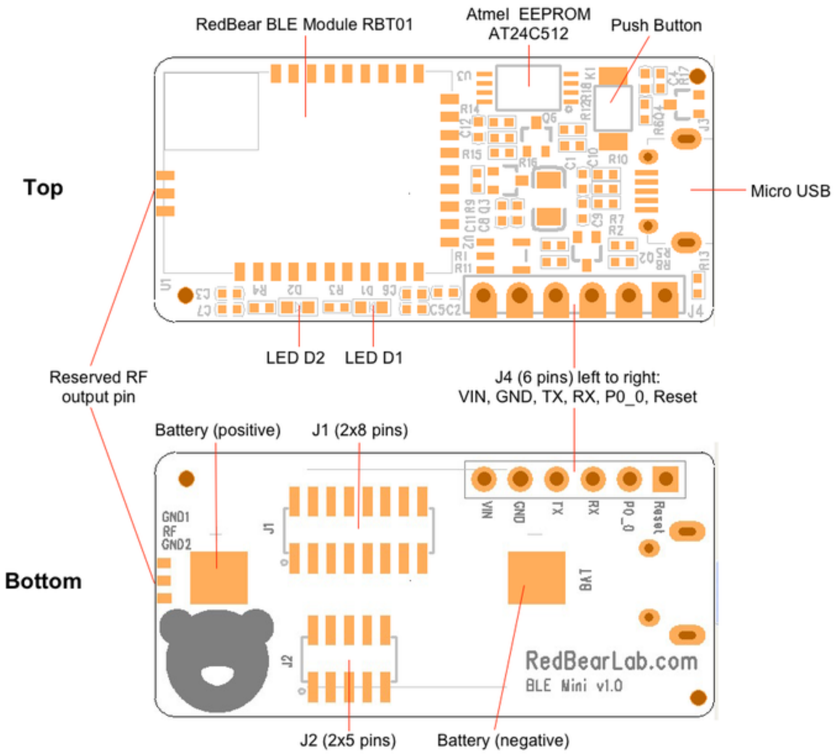


Figure 11.Circuit diagram of BLE mini board [6]

6. Data Processing and User Interface Unit

6.1 Computer (Matlab)

A Matlab code is ready to use at this point in case users prefer to use a computer as their GUI. Matlab is a very powerful tool that is able to read the sensors' data directly from Arduino board and convert them into a visualized result. The way to achieve this is to first create a Matlab function that defines the serial COM port that Arduino board is connecting, and then the function will listen to any commands arriving at the serial port and execute them. On the other hand, we need to upload a correspond program on Arduino board to make sure all data are sent and stored at a correct location that matches M-function's serial port. After those preparation steps, we shall be able to communicate between Matlab and Arduino.

A new M-function is needed to process our numerical results and turn them into graphic result. "imagesc" and "colormap" are two key functions to achieve this goal. "imagesc" function scales image data to the full range of the current colormap and displays the image. A "colormap" function is a matrix of real numbers. Each row of the matrix represents one color. A graphic result (Fig.12) should be display on a computer.

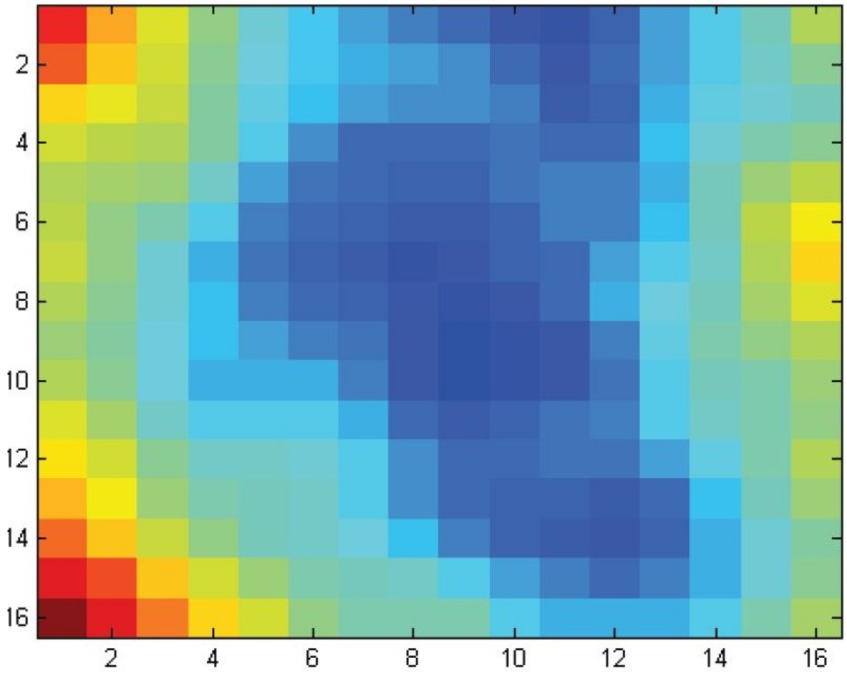


Figure 12. Pressure mapping on Matlab[8]

6.2 Smartphone

In this section, we will focus on one of the most important parts of the CMS: the phone application (Fig.13).The application testing part will be included in appendix section.

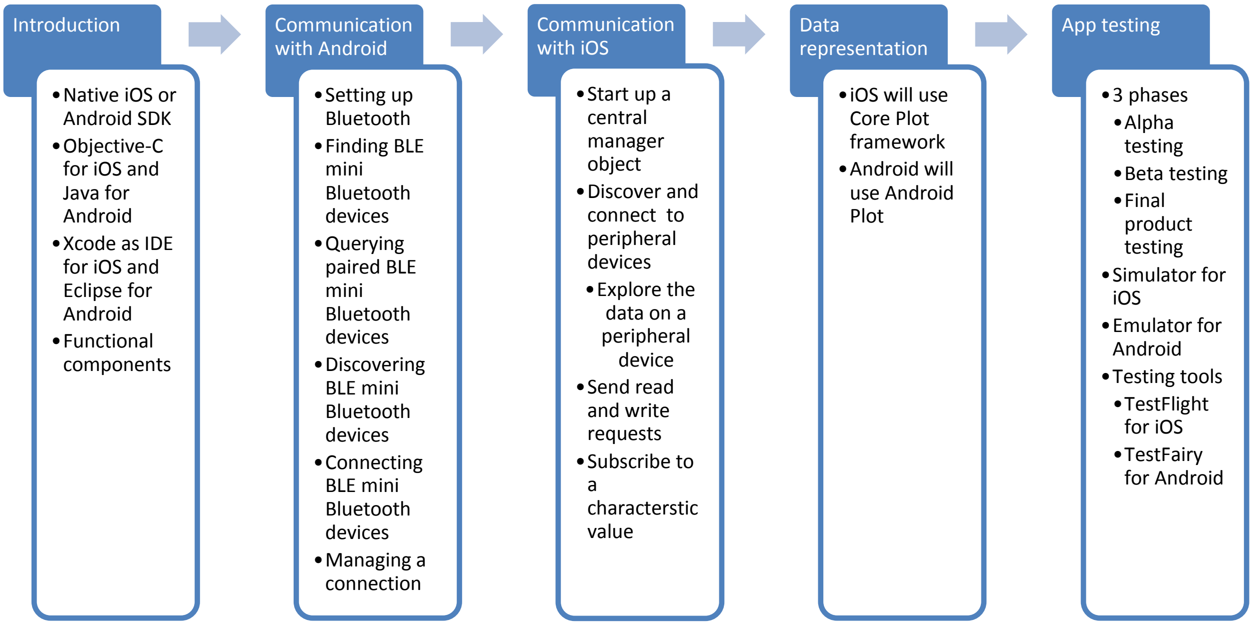

Introduction
• Native iOS or Android SDK
• Objective-C for iOS and Java for Android
• Xcode as IDE for iOS and Eclipse for Android
• Functional components
Communication with Android
• Setting up Bluetooth
• Finding BLE mini Bluetooth devices
• Querying paired BLE mini Bluetooth devices
• Discovering BLE mini Bluetooth devices
• Connecting BLE mini Bluetooth devices
• Managing a connection
Communication with iOS
• Start up a central manager object
• Discover and connect to peripheral devices
• Explore the data on a peripheral device
• Send read and write requests
• Subscribe to a characterstic value
Data representation
• iOS will use Core Plot framework
• Android will use Android Plot
App testing
• 3 phases
• Alpha testing
• Beta testing
• Final product testing
• Simulator for iOS
• Emulator for Android
• Testing tools
• TestFlight for iOS
• TestFairy for Android

Figure 13. Process of introducing phone application

6.2.1 Introduction to the app

Svasth Healthcare’s CMS App will be built using the native iOS and Android SDKs which uses Objective-C/Swift for iOS and Java for Android as the programming languages. iOS development is performed using Xcode as its IDE and Android development is performed using the Eclipse IDE. We will be using ReadBearLab’s own library for the communication with the BLE mini Bluetooth device [16].The app has the following functional components:

1. Connectivity of the app with the BLE mini Bluetooth device.
2. Data transfer between BLE mini and app.
3. FSR matrix data display in the form of Dynamic Pressure Images (DPIs).
4. Temperature readings in Celsius or Fahrenheit.
5. Humidity readings in terms of absolute humidity.
6. Reminder pop-up window to alert the user.
7. Control of the vibration motors.
8. Data transfer between app and the PC via USB port or Bluetooth.
9. Logo design
10. GUI, menus and transition within app.

6.2.2 Communication with the app

6.2.2.1 Android

The Android platform includes support for the Bluetooth network stack, which allows the App to wirelessly exchange data with *CMS*. Using the Bluetooth APIs, an Android application can perform the following:

- Scan for BLE mini Bluetooth device.
- Query the local Bluetooth adapter for paired BLE mini Bluetooth device.
- Establish RFCOMM channels.
- Connect to other BLE mini Bluetooth devices through service discovery.
- Transfer data to and from other BLE mini Bluetooth devices.
- Manage multiple connections.

Here is the sequence which has to be followed for pairing with BLE mini Bluetooth devices:

a. Setting up Bluetooth
   - Get the Bluetooth Adapter
   - Enable Bluetooth

b. Finding BLE mini Bluetooth devices
   - Using the Bluetooth Adapter, you can find remote BLE mini Bluetooth devices either through device discovery or by querying the list of paired (bonded) devices.

c. Querying paired BLE mini Bluetooth devices
   - Before performing device discovery, it’s worth querying the set of paired devices to see if the desired device is already known. To do so, call getBondedDevices().

d. Discovering BLE mini Bluetooth devices
   - To start discovering devices, simply call startDiscovery(). The process is asynchronous and the method will immediately return with a Boolean indicating whether discovery has successfully started.

e. Connecting BLE mini Bluetooth devices
   - The server and client are considered connected to each other when they each have a connected Bluetooth Socket on the same RFCOMM channel. At this point, each device can obtain input and output streams and data transfer can begin.

f. Managing a connection

- When you have successfully connected two (or more) BLE mini Bluetooth devices, each one will have a connected Bluetooth Socket.
- Using the Bluetooth Socket, the general procedure to transfer arbitrary data is simple:
    - Get the InputStream and OutputStream that handle transmissions through the socket, via getInputStream() and getOutputStream(), respectively.
    - Read and write data to the streams with read(byte[]) and write(byte[]).

6.2.2.2 iOS

In iOS, we use the Core Bluetooth framework for the communication with the BLE mini Bluetooth devices. It provides the classes needed for your iOS apps to communicate with devices that are equipped with Bluetooth low energy wireless technology.
There are two major players involved in all Bluetooth low energy communication: the central and the peripheral. Based on somewhat traditional client-server architecture, a peripheral typically has data that is needed by other devices. A central typically uses the information served up by peripherals to accomplish some particular task.

Here is the sequence which has to be followed for pairing with BLE mini Bluetooth devices:

1. Start up a central manager object – CBCentralManager: is the class we use.
2. Discover and connect to peripheral devices that are advertising - scanForPeripheralsWithServices:options: is the function we use for finding nearby devices.
3. Explore the data on a peripheral device after you've connected to it - connectPeripheral:options: is the function we use for exploring the device connected.
4. Send read and write requests to a characteristic value of a peripheral's service - readValueForCharacteristic: is used to read the values from the device and peripheral:didWriteValueForCharacteristic:error: is used for writing values.
5. Subscribe to a characteristic value to be notified when it is updated. This is how the Bluetooth framework works in both iOS and Android.

6.2.3 Presentation of the data

iOS will use CorePlot framework [7] for the data presentation and Android will use AndroidPlot [10]. Core Plot is a 2D plotting framework for Mac OS X and iOS. It is highly customizable and capable of drawing many types of plots. Because Core Plot is intended to be used in scientific, financial, and other domains where correctness is paramount, unit testing is integrated into the framework.

AndroidPlot is an API for creating dynamic and static charts within Android application. It's designed from the ground up for Android, compatible with all versions of Android from 1.6 onward and is used by over 500 apps on Google Play

7. Test Plan

The system test plan consists of multiple tests outlined in Appendix attached to this document to ensure that the hardware and software components function properly as standalone functional units, as well as the integrated product as a whole.

8. Conclusion

The Comfort Mat System as designed by the team Svasth Healthcare is intended to help people prevent pressure sores from developing by creating a Comfort Mat System which informs the user about how much pressure is exerted in a specific area. Also by incorporating a temperature and humidity sensor, we can detect unfavorable environmental conditions where pressure sores develop more quickly. To combat against pressure sores, we incorporated vibration motors into our mat, which helps circulate blood in the affected region when activated.

The design specifications in this document present how the functional specifications of the CMS are to be fulfilled in detail. These specifications will be adhered to as much as possible during the development and implementation to meet the functional specifications. The test plan included in this document will help ensure that the design specifications are met and work as designed.

9. Bibliographies

[1](n.d.). Retrieved November 1, 2014, from TestFlight: https://www.testflightapp.com/

[2] (n.d.). Retrieved November 2, 2014, from TestFairy: http://testfairy.com/

[3]*AA battery*. (2014, October 26). Retrieved October 30, 2014, from Wikipedia: http://en.wikipedia.org/wiki/AA_battery

[4]*About iOS Simulator*. (n.d.). Retrieved November 2, 2014, from iOS Developer Library: (https://developer.apple.com/library/ios/documentation/IDEs/Conceptual/iOS_Simulator_Guide/Introduction/Introduction.html#//apple_ref/doc/uid/TP40012848-CH1-SW1)

[5]*Android Emulator*. (n.d.). Retrieved November 1, 2014, from Android Developer: http://developer.android.com/tools/help/emulator.html

[6]*BLE Mini*. (n.d.). Retrieved October 21, 2014, from redbearlab: http://redbearlab.com/blemini/

[7]*core-plot*. (n.d.). Retrieved November 1, 2014, from GitHub: https://github.com/core-plot/core-plot

[8]*Data Reader Toolkit API*. (n.d.). Retrieved October 25, 2014, from Tekscan: http://www.tekscan.com/data-reader-toolkit

[9]*DC motor direction control using L293D*. (n.d.). Retrieved October 30, 2014, from Electroons: http://www.electroons.com/8051/electroons/dc_motor_control.html

[10]*Docs*. (n.d.). Retrieved November 1, 2014, from androidplot: http://androidplot.com/docs/

[11]Griffin, P. (2008, December 10). *How many hospital patients suffer from bed sores?* Retrieved September 20, 2014, from www.bedsorefaq.com: http://www.bedsorefaq.com/how-many-hospital-patients-suffer-from-bed-sores/

[12]Interlink Electronics. (2012, January 6). *FSR 400 Series Data Sheet.* Retrieved October 3, 2014, from Digikey: http://media.digikey.com/pdf/Data%20Sheets/Interlink%20Electronics.PDF/FSR400_Series.pdf

[13]Kira, M., Nandanwar, A., Naidu, M., & Rajulu, K. V. (2012, January 1). *Effect of Density on Thermal Conductivity of Bamboo Mat Board.* Retrieved October 25, 2014, from International Journal of Agriculture and Forestry: http://article.sapub.org/pdf/10.5923.j.ijaf.20120205.09.pdf

[14]Liu, T. (n.d.). *Digital relative humidity & temperature sensor AM2302/DHT22.* Retrieved October 20, 2014, from Your specialist in innovating humidity & temperature sensors: http://www.adafruit.com/datasheets/Digital%20humidity%20and%20temperature%20sensor%20AM2302.pdf

[15]OBUS Forme Convoluted Foam Mattress Topper. (n.d.). Retrieved October 24, 2014, from Serars: http://www.sears.ca/product/obus-forme-convoluted-foam-mattress-topper/696-000042168-MP-C1515-15T

[16]RedBearLAB/BLEShield. (n.d.). Retrieved November 1, 2014, from GitHub: https://github.com/RedBearLab/BLEShield

[17]W.Ferguson-Pell, M. (n.d.). *Seat Cushion Selection.* Retrieved October 28, 2014, from US Department of Veterans Affairs: http://www.rehab.research.va.gov/mono/wheelchair/ferguson-pell.pdf

10. Appendix: Detailed Test Plans

Hardware:

Functional units	Function to be tested	Test to be conducted
FSR array	Full array of FSRs works with single input	Apply pressure to single FSR while observing change in resistance
	Full array of FSRs work with multiple inputs simultaneously, while outputting correct pressure	Apply pressure to multiple FSRs while observing the change in resistance
	Calibration of FSRs	Value of output is 0 when no pressure is applied
	Impact from pressure relief cushion	Compare the value of full array of FSRs with and without pressure relive cushion. The result should be consistent
Temperature and Humidity Sensor	Outputting correct temperature	Bring the sensor to different temperatures and check if it gives the right temperature
	Outputting correct humidity	Bring the sensor to different humidity environments and check if it gives the right humidity
Vibration Motor	Vibrates with reasonable force when power is supplied	Reasonable vibration force when holding the motor in the hand after powering it.
	Comfort level	Adjust motor's speed level while tester is sitting on it
Bamboo fabric cover	Water absorption	Gently pour some water on the cover and check whether water can be absorbed
	Breathability	Heat the cover quickly for a short period of time and check the temperature to confirm heat dissipation
BLE mini	Continuity of transmitted signal	Transmit a large set of data to wireless device frequently

Table 3. Testing plan for hardware

Software:

The phone application will be tested in 3 phases

Testing phase	Function to be tested	Test to be conducted
Alpha testing	Performance on iOS	*Alpha tests* will be performed on the simulator that comes with iOS [4].
Beta testing	Performance on Android	*Beta tests* will be performed on the emulator that comes with Android [5].
Final product testing	Overall performance on actual device	Applications will be tested on latest OS versions and development will be performed in such a manner that apps can be easily ported to future OS versions.
	Compatibility test	It will be compatible to all iOS devices with iOS 6.0, iOS 7.0 and iOS 8.0. Android compatibility - Android 4.2 and above. Latest Android 5.0.
	Performance of iOS application over the air	For iOS, we use TestFlight tool [1], which allows Beta testing on the fly. It is a free platform used to distribute beta and internal iOS applications to team members over-the-air. Our team can then manage beta testing campaigns and receive feedback with TestFlight's dashboard
	Performance of Android application over the air	For android, it's called TestFairy [2]. In order to test an app with TestFairy, all we need to do is upload APK file (Android app) to TestFairy account and invite beta users to test it.

Table 4. Testing plan for software

Testing unit	Function to be tested	Test to be conducted
Matlab	Data consistency	Apply different forces on the FSR matrix for a short period of time, all sensors' data can be transmitted and received simultaneously by Matlab
	Pressure mapping result	Apply pressure on the FSR matrix; a pressure mapping result will be displayed. The color of different pressure levels will be changed accordingly

Table 5. Testing plan for Matlab