

Future Wearables

Advisor: Suraj Kothari

Client: Ted Kepros and Ensoft

Members: Aaron Reyes - Team Leader

Nick Plutt - Webmaster

William Park - Communication Leader

Josh Cline - Key Concept Holder #1

Nick Gonner - Key Concept Holder #2

- [Introduction](#)
- [Feasibility](#)
- [Risks](#)
- [Budget](#)
- [Concept Sketch](#)
- [System Block Diagram](#)
- [System Description](#)
- [Operating Environment](#)
- [User Interface Description](#)
- [Functional and Non-Functional Requirements](#)
- [Validation and Verification](#)
- [Market and Literature Survey](#)
- [Deliverables](#)
- [WBS and Project Plan](#)
- [Conclusion](#)

Introduction

In today's world sensors are becoming a prominent part of everyday life. They have become small, cheap, and powerful enough to put into almost everything. This has led to an increase in wearable devices capable of many different functions. This increase opens up the possibility of wearable devices to help monitor and improve our own health. This also allows medical professionals to better diagnose and treat their patients.

Our goal is to create a wearable device that is able to monitor back posture through sensor data. This data will then be sent to a smartphone that will be able to analyze and interpret the data for users and medical professionals. To achieve this the data needs to be collected using sensors and a microcontroller. For this device to be practical the information needs to be sent wirelessly to a smartphone so that it can be interpreted, to be useful to the user. This means that the smartphone will need an app capable of interpreting data so it can be used in different ways.

Feasibility

This project is very feasible for us, as we have a myriad of different talents on our team. Overall the project requires hardware knowledge for the creation of the device, embedded system knowledge for the programming of the hardware, and software/mobile development knowledge for creation of the app to analyze the data as well as dump it to a web server. While there are many different aspects on this project, our team has the talents and capabilities needed to succeed.

The hardware knowledge is covered by Aaron Reyes and Nick Gonner who are an electrical engineer and computer engineer respectively. The embedded system aspect covered by Aaron Reyes, Nick Gonner, and Josh Cline. This works well as Aaron Reyes is the electrical engineer who will work on the device along with Nick Gonner, who is a computer engineer focused more on embedded systems, and Josh Cline who is a computer engineer with more of a focus on software development. The final aspect of mobile development is well covered by Nick Plutt, Will Park, and Josh Cline. This is covered well as both Nick Plutt and Will Park are software engineers and Josh Cline is a computer engineer that has worked on mobile development before and will have helped with the embedded systems of the device.

As shown above we have a very motivated and capable team to ensure the success of this project. With the team's good makeup and motivation as well as our experience in each aspect, there is a strong likelihood that this project will be very successful.

Major Risks

Risk	Probability	Criticality	Risk Factor (Probability X Criticality)	Mitigation Strategy
Lack of knowledge of device specific hardware	.25	5	1.25	Research parts for initial prototype. Once we start working on the prototype we will gain more knowledge.
Problems in analyzing data(data is not useful)	.50	60	30	Rework the prototype for better data.
Amount of data being processed.	.50	5	2.5	Limit the rate at which we receive data but it is still usable.
Creation of working initial prototype taking longer than expected.	.20	15	3	Work concurrently so that it does not drastically affect the project.
Software development delayed because of difficulty getting useful data to test.	.20	10	2	Do unit testing and other test to help ensure it will work correctly with real data.
Amount of data being transmitted over bluetooth or to	.35	5	1.75	Use the microcontroller to process some data to decrease

web server is too large.				the amount.
--------------------------	--	--	--	-------------

Budget

This device will require six hardware components: two inertial measurement units, one electromyography sensor, an SD card interface, a bluetooth module, and a microcontroller to manage each of these sensors. The following parts have been selected to use in the initial prototype.:

Item Name	Price Per Item	Number of Items	Total Cost
MPU 6000 Inertial Measurement Unit(IMU) from CSGshop.com	\$25.00	4	\$100.00
Muscle Sensor v3 from SparkFun.com	\$26.00	1	\$26.00
BlueSMiRF Silver Bluetooth from SparkFun.com	\$25.00	1	\$25.00
SD/MMC breakout board from SparkFun.com	\$10.00	1	\$10.00
Pro Micro Microcontroller from SparkFun.com	\$20.00	1	\$20.00

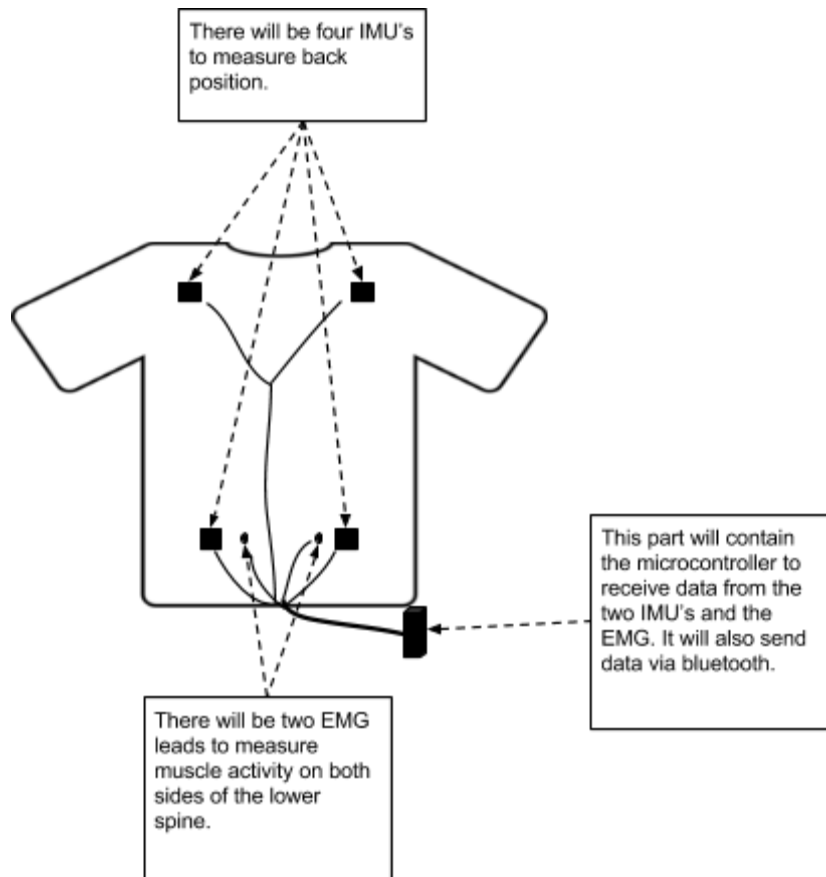
Including soldering, wiring, battery pack, and the harness, we expect that this device will cost no more than \$250 to build, for the initial prototype. Assuming that the initial prototype will not be perfect we can double our costs for the whole project, making the estimated budget \$500 dollars which is within the clients expectations.

Assessment of Design/Alternatives

We believe that the proposed design is a reasonable and cost effective solution to the problem. An accelerometer and a gyroscope are necessary to detect the change in orientation of a person.

We have decided to use an IMU in our design, which combines both of these sensors onto one chip. We prefer this option because it reduces the number of devices we need to interface with, 2 IMU chips versus 2 accelerometers and 2 gyroscopes, which simplifies the design. We determined two viable IMU units, the MPU 6000 and the MPU 6050. Both devices provide 16 bit resolution for each gyroscope and accelerometer axis data, an ample range of measurement, and operate at 3.3V. We chose the MPU 6000 over the MPU 6050 because of the ability to communicate over SPI or I2C, whereas the MPU 6050 only supports I2C communication. SPI is capable of much faster transfer speeds than I2C and the other devices necessary for the project, i.e. the SD card interface and possibly the Bluetooth unit, are also SPI compatible, making SPI the preferred bus protocol. The microcontroller we chose is the ARM Maple microcontroller. Maple is a 32-bit microcontroller that operates at 3.3V, supports I2C and SPI interfaces, runs quickly at 72 Mhz, and has available software development tools and libraries. The alternatives we investigated, Atmel's ATmega32U4, Arduino UNO, and Arduino Pro Mini 328 fell short of the Maple in a few critical areas. The ATmega32U4 is an 8 bit microcontroller, which is not ideal when operating on the 16 bit data received from the IMUs. The UNO operates at 5V, which is out of range for most of our peripherals, and is also an 8-bit microcontroller. The Arduino Pro Mini runs at 8 Mhz, much slower than the Maple, and is still only an 8-bit microcontroller.

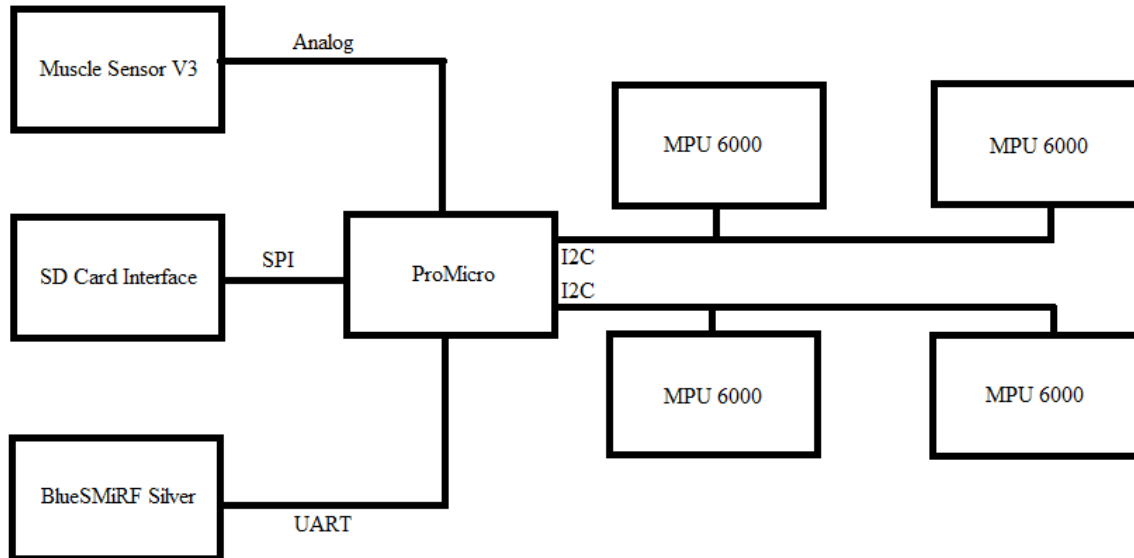
Concept Sketch



System Block Diagram

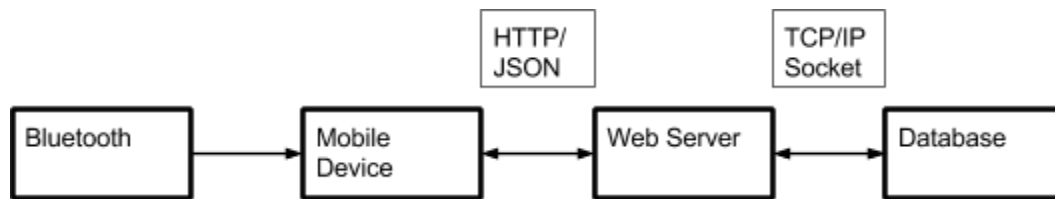
Hardware Diagram

This is a block diagram overview of the system components and the interconnections between them.



Software connections

This is a diagram of the different software components and the connections between them.



System Description

This device measures spinal position and muscle activity to analyze posture. Four inertial measurement units(IMUs) are used to track the orientation and position of the spine. Two IMUs are located on the upper back on both shoulder blades, below the neck. The other two IMUs are located near the base of the back on both sides of the spine above the waistline. An Electromyography(EMG) sensor is used to track muscle activity in the lower back muscles(spinal erectors). The EMG is located on the lower back, above the waistline. This device is designed to be used for monitoring patients who experience chronic back pain, poor posture, repetitive lifting tasks, or looking at back problems in sports.

Operating Environment

Part of the product will be operating on an embedded system. This part will need to be programmed in such a way to minimize processing and memory usage. The other part of the system will be operation on a smart phone. This part will not have much consideration of computing power however we will have to be careful with how much and how often it receives

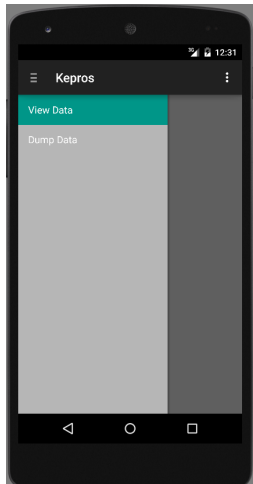
data. The third operating environment will be a webserver that will send and receive from the mobile devices and store data in a database.

User Interface Description



Physician side of application

This screen would be where the main activities would be displayed. Since it is just a blank activity any thing could be here. At the top is the title of the current activity or fragment and an example of a button that could be used in some manner on the page. It also has three dots in the top right that would display a button for the settings page. The other button with three lines in the top left opens the navigation drawer which is explained below.



This image shows the navigation drawer, which is used to navigate around the different part of the application. Right now it only has two placeholder tabs that all open up to an empty activity. The two placeholder tabs are view data and dump data. The name at the top of the app when the navigation bar is open is the name of the app.

With these two images we can see the basics of how the UI will look. It also shows how navigation of the app will be done.

Client Side of application

There will be no UI for client side as it will just collect and send data.

Functional and Non-Functional Requirements

Functional Requirements

- Connect through bluetooth to hardware.
- Receive data from hardware.
- Analyze from hardware.
- Allow viewing of data from hardware.
- Store data from hardware.
- Dump data from hardware to web server.
- The device will measure muscle activity in the lower back region.
- The device will measure back orientation from two positions, the upper back between the shoulder blades just below the neck and the lower back above the waistline.
- The device will be battery operated.
- The device will be embedded in a wearable “harness” to be worn by patients.
- The device will store records of data and the time it was collected.
- The device will communicate with an android device.
- The device will last on a battery for a minimum of 18 hours, between charging periods.

Non-Functional Requirements

- Efficiency in analyzing data.
- Performance in sending data.
- Extensibility for future features that may be implemented.
- Comfortability of harness for device.
- The device will fit a variety of patients.

Validation and Verification

Validation

We plan to keep our client up to date for the duration of the project to ensure that they are happy with the development of the device. Doing this should ensure that the client is happy with the final result and specifications.

Steps for Validation

1. Meet with the client to get preliminary specifications
2. Research parts for project and start designing the hardware and software
3. Meet with the client to go over researched parts and preliminary design ideas
4. Revise part list according clients suggestions
5. Send finalized parts list to the client

6. Finalize design plans and send to client for review
7. Revise plan according to the client suggestions
8. Create prototype and collect data
9. Send data to the client to see if it will be useable
10. Revise device if needed for better data
11. Work on sending data to the mobile application
12. Develop basic mobile functionality
13. Go over current prototype and mobile functionality with the client
14. Revise prototype and mobile application to client suggestions
15. Work on improving prototype and mobile application
16. Design and build harness
17. Go over prototype, harness, and mobile application with the client
18. Create final prototype for final testing and presentation
19. Go over final prototype with the client
20. Revise final prototype if needed
21. Put finishing touches on prototype

Verification

Coding

For the software and embedded system verification we will have various measures to ensure the quality of the code. We have four people that are experienced with dealing with software. Each person is going to be somewhat familiar with and working on all aspects of the coding. This will ensure that if we have one person gone or busy the others can pick up the slack. This also means that the people are able to double check others work and spot problems early on. We also will have various reviews, checks, and testing of the code so that we can find and document problems early on. We will also be using github for version control and reporting bug issues.

Milestones

These are important parts of the development of our product and as the parts are developed they need to be tested before moving on to the next step.

Software:

1. General Software Outline
2. Android UI Framework
3. SQLite Android Database Setup
4. Shared preference database
5. Able to store data in remote database

6. UI to view data
7. Integration and receive with bluetooth
8. Web server creation for testing data dump
9. Data dump to web server.

Hardware:

1. Identify necessary parts
2. Generate initial hardware solution to be approved
3. Parts ordered first week of March
4. Circuit design, connect hardware
5. Configure the microcontroller, read data from peripherals, store data on SD card, transmit data via bluetooth
6. Write preliminary signal processing
7. Hardware re-evaluation
8. Research and develop battery solution
9. Develop harness for device

Market and Literature Survey

The Lumo Back is a wearable device that monitors the lower back posture of the user. The data is collected by a cloud device and is available for viewing by the user through an app. The device can be set to vibrate when the wearer slouches. Lumo's new product, the Lumo Lift, is a small wearable device designed to measure upper body and shoulder posture. It will vibrate if the wearer slouches. Our device combines the capabilities of these two devices and also tracks the muscle activity in the lower back. Our device is not meant to be corrective, but rather collect and process data on orientation and muscle activity for a user or medical professional to analyze. Our device measures muscle activity and uses this data to evaluate posture and lifting technique, which the Lumo Back is not equipped to do.

Deliverables

PHASE I

Complete a hardware design including list of components necessary to achieve the job, e.g., accelerometer, gyroscope, electromyography, and how they will be interconnected both physically in hardware and logically in software.

PHASE II

Build a prototype for preliminary trials.

12/15

Creation Date 2/11/2015

Updated Date 4/26/2015

Version 3.0

PHASE III

Build final prototype for final review.

Work Breakdown Structure and Project Plan

Task Name	Start	End Date	Assigned To	Duration (Days)
Parts for Delivery	03/02/15	03/31/15	FedEx	30
Configure Microcontroller	03/31/15	04/05/15	Hardware Team	6
Circuit Design/Connect Hardware	03/31/15	04/05/15	Hardware Team	6
Programming the Embedded Systems	04/05/15	04/15/15	Hardware/Software Tear	11
Read Data from Peripherals	04/15/15	05/01/15	Nick G	17
Store Data on SD Card	04/15/15	05/01/15	Aaron R	17
Write Preliminary Signal Processing	04/15/15	05/01/15	Nick G/Aaron R.	17
SQLite Android Database Setup	04/15/15	05/01/15	Software Team	17
Shared Preference Database	04/15/15	05/01/15	Software Team	17
Transmit Data via Bluetooth	08/24/15	08/30/15	Aaron R	7
Connecting and Receiving from Bluetooth	08/24/15	08/30/15	Software Team	7
Hardware Re-Evaluation	08/24/15	09/06/15	Hardware Team	14
Research and Develop Battery Solution	08/24/15	09/27/15	Aaron R.	35
Send Data to Web Database	08/30/15	09/03/15	Will P.	5
Store Data in Database	08/30/15	09/08/15	Josh C./Nick P.	10
Retreive Data from Web Database	09/03/15	09/09/15	Will P.	7
View Data in Graph	09/09/15	09/22/15	Josh C.	14
Develop Harness for Device	09/25/15	10/15/15	All Team	21
Refine and Optimize Device	10/01/15	11/25/15	All Team	56

*Visual representation in appendix A.

Conclusion

In conclusion our team feels very confident in the success of this project. Although risks have been identified our team is confident that by using the mitigation strategies listed above, that these issues should not prevent us from completing this project. Also through the use of our detailed plans and outlines we will create several iterations of the device, to ensure its success. And by including the client throughout the creation process, we ensure that the client will be happy with the design and capabilities of the final product.

Appendix A

Task Name	Jan 25					Feb 1					Feb 8					Feb 15					Feb 22						
	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F
General Software Outline																											
Identify Necessary Parts																											
Android UI Framework																											
Parts List to be Approved																											
Parts for Delivery																											
Configure Microcontroller																											
Circuit Design/Connect Hardware																											
Programming the Embedded Systems																											
Read Data from Peripherals																											
Store Data on SD Card																											
Write Preliminary Signal Processing																											
SQLite Android Database Setup																											
Shared Preference Database																											
Transmit Data via Bluetooth																											
Connecting and Receiving from Bluetooth																											
Hardware Re-Evaluation																											
Research and Develop Battery Solution																											
Send Data to Web Database																											
Store Data in Database																											
Retrieve Data from Web Database																											
View Data in Graph																											
Develop Harness for Device																											
Refine and Optimize Device																											

Task Name	Mar 1					Mar 8					Mar 15					Mar 22					Mar 29					Apr 5					Apr 12					Apr 19					Apr 26				
	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S			
General Software Outline																																													
Identify Necessary Parts																																													
Android UI Framework																																													
Parts List to be Approved																																													
Parts for Delivery																																													
Configure Microcontroller																																													
Circuit Design/Connect Hardware																																													
Programming the Embedded Systems																																													
Read Data from Peripherals																																													
Store Data on SD Card																																													
Write Preliminary Signal Processing																																													
SQLite Android Database Setup																																													
Shared Preference Database																																													
Transmit Data via Bluetooth																																													
Connecting and Receiving from Bluetooth																																													
Hardware Re-Evaluation																																													
Research and Develop Battery Solution																																													
Send Data to Web Database																																													
Store Data in Database																																													
Retrieve Data from Web Database																																													
View Data in Graph																																													
Develop Harness for Device																																													
Refine and Optimize Device																																													

Task Name	Aug 23					Aug 30					Sep 6					Sep 13					Sep 20						
	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F
General Software Outline																											
Identify Necessary Parts																											
Android UI Framework																											
Parts List to be Approved																											
Parts for Delivery																											
Configure Microcontroller																											
Circuit Design/Connect Hardware																											
Programming the Embedded Systems																											
Read Data from Peripherals																											
Store Data on SD Card																											
Write Preliminary Signal Processing																											
SQLite Android Database Setup																											
Shared Preference Database																											
Transmit Data via Bluetooth																											
Connecting and Receiving from Bluetooth																											
Hardware Re-Evaluation																											
Research and Develop Battery Solution																											
Send Data to Web Database																											
Store Data in Database																											
Retrieve Data from Web Database																											
View Data in Graph																											
Develop Harness for Device																											
Refine and Optimize Device																											

Task Name	Sep 20				Sep 27				Oct 4				Oct 11				Oct 18				Oct 25				Nov 1				Nov 8				Nov 15				Nov 22			
	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	
General Software Outline																																								
Identify Necessary Parts																																								
Android UI Framework																																								
Parts List to be Approved																																								
Parts for Delivery																																								
Configure Microcontroller																																								
Circuit Design/Connect Hardware																																								
Programming the Embedded Systems																																								
Read Data from Peripherals																																								
Store Data on SD Card																																								
Write Preliminary Signal Processing																																								
SQLite Android Database Setup																																								
Shared Preference Database																																								
Transmit Data via Bluetooth																																								
Connecting and Receiving from Bluetooth																																								
Hardware Re-Evaluation																																								
Research and Develop Battery Solution																																								
Send Data to Web Database																																								
Store Data in Database																																								
Retrieve Data from Web Database																																								
View Data in Graph																																								
Develop Harness for Device																																								
Refine and Optimize Device																																								