Future Wearables

Advisor: Suraj Kothari

Project: Future Wearables

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

In todays world sensors are becoming a prominent part of everyday life. They have become small enough and cheap 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 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.

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.

PHASE III
Build final prototype for final review.

System Level Design

## Functional Decomposition
- Connect through bluetooth to hardware.
    - Have hardware transmit data over bluetooth
    - Set up bluetooth to work for the app on the phone.
    - Set up a way to pair the hardware to the phone.
- Receive data from hardware using bluetooth
- Analyze data 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.

Block Diagram of the Concept

Hardware Diagram

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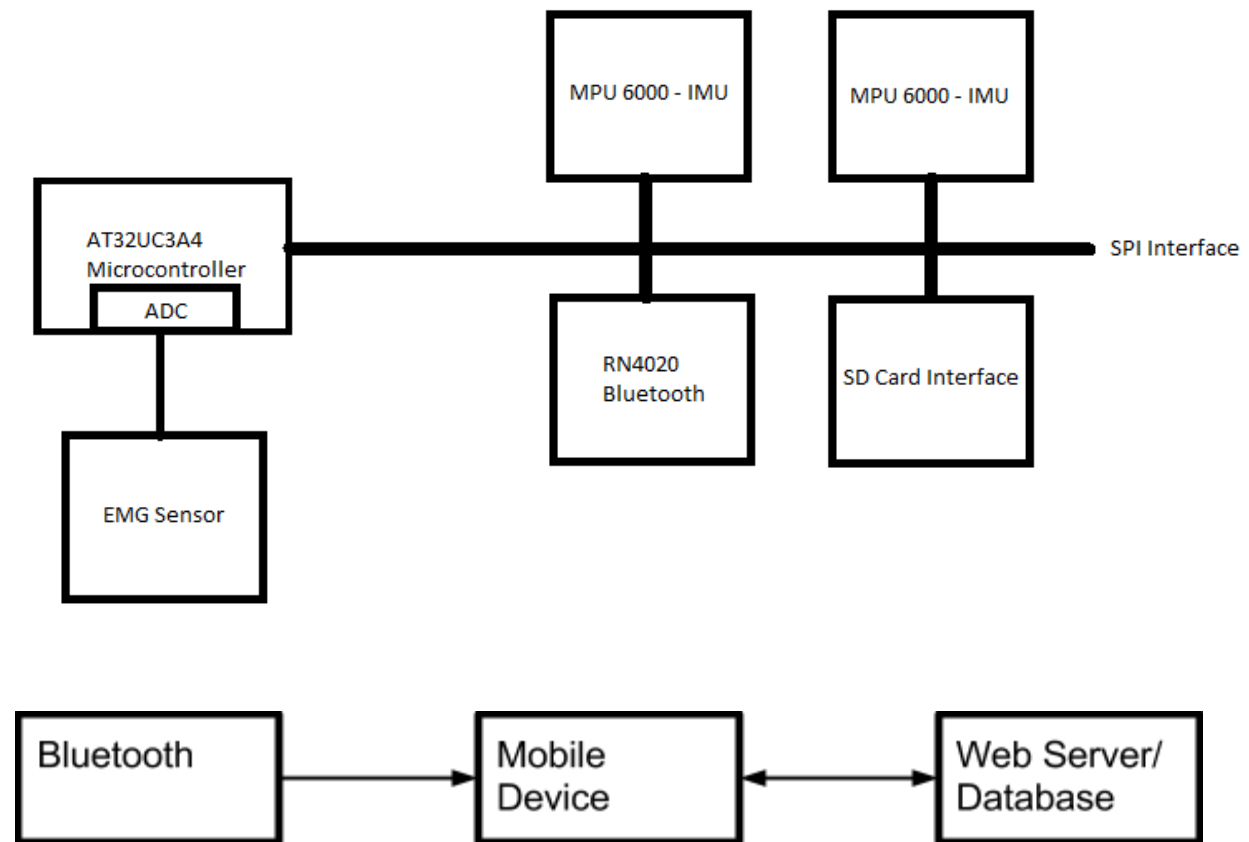
Detail Description

I/O Specification

Hardware

    Input

- IMU from the upper left, upper right, lower left, and lower right of the back.
- EMG from the left and right of the lower spine.

Output
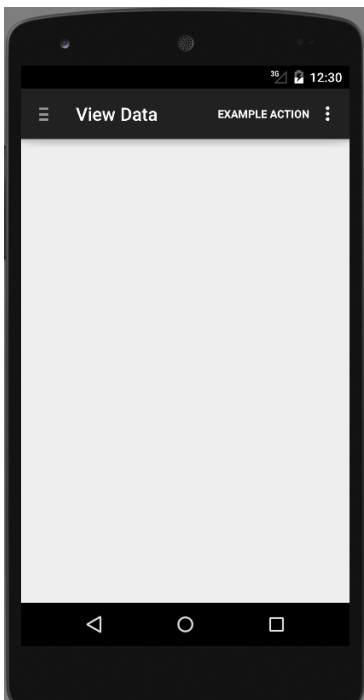- Filtered data over bluetooth

Software
    Input
- Data points received from hardware

    Output
- Either raw data from hardware or possibly further filtered down data.

Interface Specification

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 left that would display a button for the settings page. The other button with three lines in the top right opens the navigation drawer which is explained below.
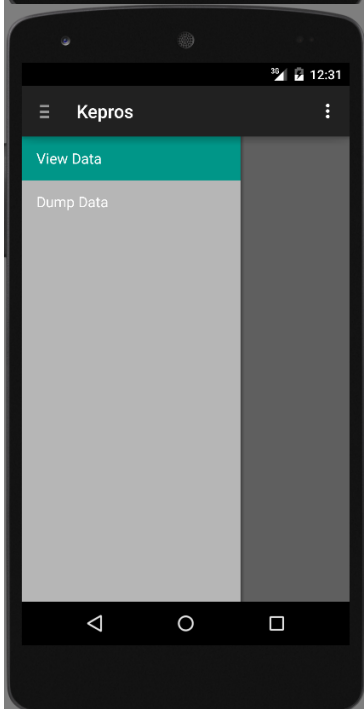
This image shows the navigation drawer, which is used to navigate around the different part of the application. Right now it only has two placeholder tabs that all open up to an empty activity. The two placeholder tabs are view data and dump data. The name at the top of the app when the navigation bar 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.

Hardware/Software Specification

Software Specifications
- Mobile application
  - Retrieve data from bluetooth
  - Take retrieve data and possibly further refine/filter it
  - Send collected data to web database/server
  - Have way to view graph of data real time as well

Implementation Issues/Challenges

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	.20	15	3	Work

working initial prototype taking longer than expected.				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 web server is too large.	.35	5	1.75	Use the microcontroller to process some data to decrease the amount.

Testing, Procedures, and Specifications

Software testing will be based on code coverage. We will try to maintain a fairly high level of code coverage throughout the app to ensure that the app has minimal bugs to the end of launch. It is also in works that we will be creating test cases based on regression testing so that our application will still work the way it should even with new changes and things being added in. Throughout our sprints with the code development. We will dedicate time to make sure that we are testing our code through the timeline of the project. Since we are developing two differents levels of the application (client and core), we will need to have two different test suites to provide coverage for both applications.

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 database
6. Maybe 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

Other Documentation

Software/Firmware Design Documents

For the software aspect of the application. We have decided to all use Android Development Studio (ADS) as our main integrated development environment. There was some talks about using Eclipse as the IDE, however, we decided that ADS would be the most beneficial IDE. Although it is rather new as an IDE, we are fairly confident that it will suffice for this project.

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.