1 Introduction

Information is a vital resource, espcially in the context of Healthcare. When a patient's health is at risk, their medical history becomes the most important piece of information in providing them with appropriate care.

At present, in Ontario, there exists no secure, liftime record of your health history. When faced with a problem, healthcare providers are left without the big picture, and often fumble for generic solutions, instead of catering care to each individual.

One situation where this information gap is the most noticiable is when patients recieve care from Emergency First Responders (EFRs). Today, EFRs respond to emergencies knowing little or nothing about the people in their care. This is unforunate, considering how beneficial information is in improving patient outcomes.

The objective of this capstone is to therefore develop an information management system that stores health records for access by patients, and their healthcare providers. Specifically, the goal of the capstone is to develop software that can be used by Emergency First Responders to access patient health records while responding to a call. This will provide EFRs with information vital to providing patients with the best care possible. In addition to software, hardware will be developed that provides EFRs with a means to quickly identify patients and access there information on the system. This hardware will also include a real-time vitals tracker, which has the potential to track vitals (i.e. Heart Rate) deemed useful by EFRs.

2 Software Design Component

The software system we are designing will function as a portal to Electronic Health Records in Ontario. The main user of the system will be paramedics. In the future, we hope users will extend to all levels of the medical profession as well as the general public. We will design an application targeted at paramedics that works on both iOS and Android phones.This application will allow paramedics to view patient information that they do not normally have access too. The app will have a loading screen, a login page, a home page, patient pages accessed by scanning health cards or inputting the health card number.

2.1 Technology

In order to create the Android and iOS app we will require certain technologies. Firstly, the app will be written using React Native a javascript framework that allows apps to be written for Android and iOS concurrently. The app will require a database to store the patient information. For this, we will use Node.js however, we are assuming that in the future if our app is put to use we will easily adapt to how the EHR are stored by Ontario.

2.2 Functional Design

The app will have basic functional requirements as well as features we would like to implement once the basic requirements are completed. The basic functional requirements are that the device follows the standards outlined on the ehealth blueprint documents put out by eHealth Ontario. This document guides us through the appropriate exchange and access of information within the app. Functionally, it will need to have a secure log-in screen for paramedics. A home screen with buttons leading to history, settings, option to connect to WiFi hardware device, and the health card scanner. Once a hardware device or a health card are connected it will then need to bring up a patient page and show the sensor data and/or the patients EHR. The history screen for the basic requirements will simply show what patients the paramedic has treated but will not allow the paramedic more access to their information once the paramedic has closed the patient's EHR. In future versions the paramedic will be able to record information about the particular visit into the EHR, and will be able to view these additions on the history screen. In the future additions, the sensed data will also be stored and be visible in the EHR so medical personell at a hospital or later at a doctor's visit will be able to view the patients vitals at the time of the paramedics visit.

2.3 Non-Functional Design

The app must have secure information storage and transmission by the standards outlined in the eHealth Blueprint. The app must have a cohesive look, including size and colour scheme in order to be easy to navigate. The app must be available in english and french. The app will be made so that the average paramedic is able to use it (assuming paramedics are males and females adults with above secondary education) and will use terms that are familiar to them. The app must be able to run on Android and iOS devices that still recieve updates from their providers. The app must not have harsh colouring or flashing lights in order to prevent harm to the paramedics.

3 Hardware Design Component

To adequately track the vitals of a patient, there are 3 main hardware components required; the receiver, the transmitter and the sensor.

3.1 Receiver

In modern society all mobile devices such as tablets and mobile phones, can connect to auxiliary devices via Bluetooth and Wi-Fi. Bluetooth communication has limitations on how many auxiliary devices can be connected to a parent device at any given time. The ideal number is 4 with the absolute maximum being 6 or 7. [1] As mass car accident can have anywhere from 3 to 15 people involved, having any paramedic limited to 4 patients per device seems impractical. Conversely, using a device as a wireless hotspot or having all devices on the same Wi-Fi network allows for 10 or up to 250 devices to be connected, respectively. [2] Because of this we have chosen to connect to our sensors through Wi-Fi with the receiver being the devices own internal Wifi receiver.

3.2 Transmitter

To transmit the data from the sensor to the mobile device, a serial WiFi transceiver module. Once the drivers are installed and the device is connected to the wireless network, the sensor will be able to connect with any device on the same network.

3.3 Sensor

Lastly, and most importantly, is the construction of the sensor itself. For the sake of this design, the sensor will only monitor the patients heart rate. A simple heart rate sensor consisting of an infrared emitter and receiver as well as an amplification circuit will be attached to the patients skin to generate a signal. The IR light is shone into the skin and depending on the amount of blood in the skin at that time, the light will scatter accordingly. If enough IR light is reflected, the receiver picks this up and output the corresponding current and voltage. This is then amplified and is sent to a microcontroller. To decrease the size of the overall sensor, an ATmega microcontroller can be used with the appropriate capacitors, oscillators, and power source to avoid the use of a full Arduino board. This chip will receive signals in from the pulse sensing circuit and will then output the signals through the Wi-Fi Transceiver module to the users mobile device.

4 Conclusion

Throughout the development process, both the software and hardware will be tested to ensure it meets design requirements. In terms of software, each module will be verified using test cases that have known outcomes to ensure functionality. The hardware will be tested in a similar way, using test cases to verify the hardware meets specification.

An example of real-world testing that can be used is in verifying the pulse-sensing capabilites. The hardware can be tested on a group member with a known pulse (through traditional human pulse reading), to ensure that the device senses pulses correctly.