# Continued Aggregation and Analysis of HealthCare Data Generated by Wearable Devices

DEC1610  
Client: UnityPoint Health  
Advisor: S. Tirthrapura  
Ben Kixmiller - Key Concept Holder  
Haythem Ebrahem - Key Concept Holder  
Brian Nguyen - Team Communications Leader  
George Ndemi - Key Concept Holder  
Johnathan Campbell - Team Webmaster  
Aakash Sheth - Team Leader

[dec1610@iastate.edu](mailto:dec1610@iastate.edu)  
Revised: March 11th, 2016

|  |  |
| --- | --- |
|  |  |
| [1 Introduction](#h.30j0zll) | 2 |
| [1.1 Project statement](#h.1fob9te) | 2 |
| [1.2 Purpose](#h.3znysh7) | 2 |
| [1.3 Goals](#h.2et92p0) | 2 |
| [2 Deliverables](#h.tyjcwt) | 2 |
| [3 Design](#h.3dy6vkm) | 3 |
| [3.1 System specifications](#h.1t3h5sf) | 3 |
| [3.1.1 Non-functional](#h.4d34og8) | 3 |
| [3.1.2 Functional](#h.2s8eyo1) | 3 |
| [3.2 PROPOSED DESIGN/METHOD](#h.17dp8vu) | 3 |
| [3.2.1 Collecting data](#h.8pbi6x206atc) | 3 |
| [3.2.2 Application UI and Data Visualization](#h.scgmsw5qvp8u) | 4 |
| [3.2.3 Server and Storing Data](#h.wqyjwvjnxi4v) | 4 |
| [3.3 DESIGN ANALYSIS](#h.3rdcrjn) | 4 |
| [4 Testing/Development](#h.26in1rg) | 5 |
| [4.1 INTERFACE specifications](#h.lnxbz9) | 5 |
| [4.2 Hardware/software](#h.35nkun2) | 5 |
| [4.2 Process](#h.1ksv4uv) | 5 |
| [5 Results](#h.44sinio) | 7 |
| [6 Conclusions](#h.2jxsxqh) | 7 |
| [7 References](#h.z337ya) | 8 |
| [8 Appendices](#h.3j2qqm3) | 8 |

# 1 Introduction

## 1.1 Project statement

We are working on sending data from wearable devices to the patient's chart where they can be used by medical personnel towards improving patient’s care and wellbeing. The data on the patient's chart should be easy to comprehend to both the patient’s and the health care workers on a user friendly UI

## 1.2 Purpose

Since the passing of the Healthcare bill, hospitals are paid by insurance according to the success of treatment. Any readmissions and patients visiting a provider for the same problem reduces the amount of reimbursement. Health care workers would like to monitor the patients, if they are following Doctor’s advice on the number of steps they take everyday, sleep pattern, glucose levels and blood pressure. This information will enable the healthcare workers to know the best treatment with the best outcome for a specific case, thereby decreasing the readmissions and increasing revenue.

## 1.3 Goals

This project entails several deliverables:

* We would like the data that is collected by the wearable devices be sent to the server then to the patient’s chart where clients can access it.
* The current UI is not user friendly, we would like to upgrade it to a better looking one and easy to scroll through.
* Currently, Facebook is the only authentication used, we would like to to enable clients to log in using Yahoo and Google accounts as most clients do not want to link their Facebook accounts with their Medical records.

# 2 Deliverables

For this project Unity Health Point required us to build on the existing apps to :

* Acquire the data from different wearables such as fitbit, garmin, Apple watch by the IOS apps.
* Setup a server, domain and database to save the data.
* push the data to the server.
* Implement different authentication methods such as facebook, google and thump print lock.
* Retrieve the data from the server and display it in different format such as graphs, comparison charts, goals.
* Implement UI that meet the UPH standards.

# 3 Design

Include any/all possible methods of approach to solving the problem. Discuss what you have done so far. What have you tried/implemented/tested etc. We want to know what you have done.

## 3.1 System specifications

Detail any specifications given and/or assumed about the project.

### 3.1.1 Non-functional

* Security
  + the data is transferred to sever will be encrypted
  + no user can access any data of other user
  + users can erase their own confidential data which stored in the phone
* Reliability
  + the app runs 24/7 without interrupt
  + the app is responsive, quick
* Performance
  + the data is compressed with advanced algorithm to help saving storage
  + the app is energy efficient

### 3.1.2 Functional

* iOS app allows user login with several different authentication option such as google, yahoo, hotmail, facebook, etc.
* iOS app automatically receives data which collected from multiple types of devices such as: fitbit, apple watch, garmin..
* iOS app analyzes the data and show useful report to user.
* iOS app send the data to server in appropriate format that can be compatible with existing data from Epic.

## 3.2 PROPOSED DESIGN/METHOD

### 3.2.1 Collecting data

Data collection part will include collecting data from different wearable such as Fitbit, Garmin, apple watch. each of these wearable has a different way of sending data as an example apple watch we can get the information direct throw the healthkit application. for Fitbit the we can get the information only from a user account with fitbit but not for the devices. Our approach is to divide this part to small section each section deal with one device and collect the data. our second approach is to make this section extendable so the client or other developer can add more devices easily.

The data targeted from the each device is the basic information about user activity like steps, heart beat and sleeping time with a room to incorporate more type of data depend on the devices.

### 3.2.2 Application UI and Data Visualization

Following the the data collection we need to represent this information in an interactive way that encourage the user to use it on his everyday activity. in addition to that maintain the security of his data by implementing other method of authentication like thump print lock and google authentication.

### 3.2.3 Server and Storing Data

Day after day it is going to be a huge amount of data so we decided all data will be push to the Unitypoint Health servers where we can store the data and provide access to medical staffs so they can analyse the user data for better care.

## 3.3 DESIGN ANALYSIS

Thus far, we have strongly designed the implementation of our application. This began with a general consideration of what the app should do based on our team discussions, client meetings, as well as input from the CyBiz team. The initial screen sketches are located in the appendix and show a general outline of the workflow of the application. The app is designed to follow iOS Application Guidelines1 as close as possible in order to remain consistent with user experience issues for iOS users.

The design also includes several key decisions on authentication and security. We have implemented Google and Facebook authentication in order to make the app available to as many people as possible. These services are strictly a means of authentication, and will later be converted to a proprietary authentication system provided by UPH once the electronic data is more viable in terms of accuracy, standardization, and integration with UPH’s electronic medical records system.

In addition to authentication, we also plan to implement a biometric authorization with the aid of iOS thumbprint locks. The intention is to have the user log in once through the authentication portal, Google or Facebook, and then store the auth token in the key store, and from then on, allow the user to unlock the app with a registered thumbprint. In case the user prefers to not use the thumbprint, an alternate pin code option will be provided.

Storage of user data has also been heavily considered though not yet implemented. Wearable data will be collected and stored on the phone, but this process does not work for all wearable, such as Fitbit or Garmin. These devices pose a considerable challenge since their data must be aggregated from an external API that needs to be parsed into a standard format and then queried from the user’s phone, as well a remote server as a final data store. Though we have not yet implemented this feature, the intention is first to start by querying data from the respective API’s and display them on the user device. Once that is accomplished, we can move on to storing such data on a remote server.

Overall, our design aims to be modular and extensible through creating interfaces for new wearables, while creating as much isolation between code as possible.

# 4 Testing/Development

## 4.1 INTERFACE specifications

Out project consists of several pieces of hardware as well as considerable software development. We will be interfacing with the hardware through high level abstractions. For example, we will be interfacing with the AppleWatch thought the WatchOS SDK provided by Apple. Data will be collected from the AppleWatch and interface with an iPhone, which has native support for AppleWatch. We will be avoiding any hardware level manipulations unless it provides a point of leverage to the end goal of collecting data.

## 4.2 Hardware/software

Hardware:

* Fitbit
* Garmin Fit
* Apple Watch
* UP Fit
* Iphone

The Fitbit, Garmin Fit, Apple Watch, and Go Fit we will be using to aggregate health data. These are the biggest competitors in the fit watch market.

Software:

* XCode (Swift Development)
* Cocoapods

## 4.2 Process

We are just getting into the real coding portion of our project; previously we had been acquiring all we needed to get started, such as xcode, access to the git repo, and testing devices, as well as forming a strong understanding of what the client needs and what we currently have. Additionally, we have set up Facebook authentication as well as IOS thumbprint authentication. We have been keeping in touch with CyBiz (and intend to keep doing so) and they will be providing us with what information we need to gather, how we need to present it, and other suggestions from practiced doctors and Unity Point. They will have that to us shortly after Spring Break (March 14-19).

While this information is being gathered, we will be working on screenflow and data-aggregation off of non-Apple devices. Unfortunately, the previous year’s design for the application will be unsalvageable, so we will end up removing all of the screens and starting afresh (a request from UPH). This will be our main focus until we hear from CyBiz what data we need. Last years group did leave us with a very competent back-end, however, and that will be extremely useful for speaking with the different devices.

Along with screen implementation, some of us will be working with getting data off non-Apple watch devices and translating it into something the iPhone can understand and store. This includes the FitBit, the Garmin Fit, and the UP Fit.

Storage is another thing we will need shortly. UPH stated that they would be willing to host server space for the storage of development data since we will not be directly touching their database with client information (for good reason). We will be in touch with the senior engineer at UPH to make sure we get this.

# 5 Results

We have been working on getting Google, Yahoo, and Facebook authentication on to the app. We also will include thumbprint lock/unlock to unlock the app in the background to avoid having to sign into Google every time you get onto the app. We started late on the project due to getting the code only a couple of weeks ago. We are a little behind other groups in development. Everyone in our group worked on learning the Swift language last week. Swift is a tough language to learn but everyone in our group feels they are fluent enough now to start working on specific features of the project.

We feel like we have a solid game plan for this semester and for next semester. We don’t feel like we will have to change our plans much for the project. If one person gets behind relative to the rest of the group, we have enough other people that are good at Swift that it shouldn’t take too much time to fix. We talk frequently with each other to make sure everyone is on the same pace. We haven’t ran into many issues so far.

# 6 Conclusions

Thus far, we have achieved the most knowledge when it comes to developing for iOS in Swift. We have learned much of the development process that delivers an app that conforms to iOS User Guidelines. As of now we have completed the following:

* Develop a partial user interface consistent with that outlined in the design phase screen sketches
* Create a login flow using Google authentication, as well as Facebook authentication from last year’s project
* Design a high level overview of what our application needs to achieve in terms of API calls and server access

As a result of work done so far, we have been able to meet some of the goals we have outlined earlier this semester. Even though many of the goals have been achieved, integrating them is still of concern. For instance, we have a partial interface done for the login screen, but the actual sign in has not been integrated. So as of now, our goals follow:

* Acquire the data from different wearables such as fitbit, garmin, Apple watch by the IOS apps.
* Setup a server, domain and database to save the data.
* push the data to the server.
* Implement different authentication methods such as facebook, google and Unity Health Point authentication (Mychart).
* Retrieve the data from the server and display it in different format such as graphs, comparison charts, goals.
* Implement UI that meet the UPH standards.

We have created a timeline using Trello as well as a weekly Google meeting doc that takes into account what each team member is responsible for. As group, we feel that we have broken down the tasks into small enough pieces that are understandable and reasonable. Through weekly check ins, we are able to maintain our development timeline.

# 7 References

List any references used in the document.

<https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/>

<https://dev.fitbit.com/docs/>

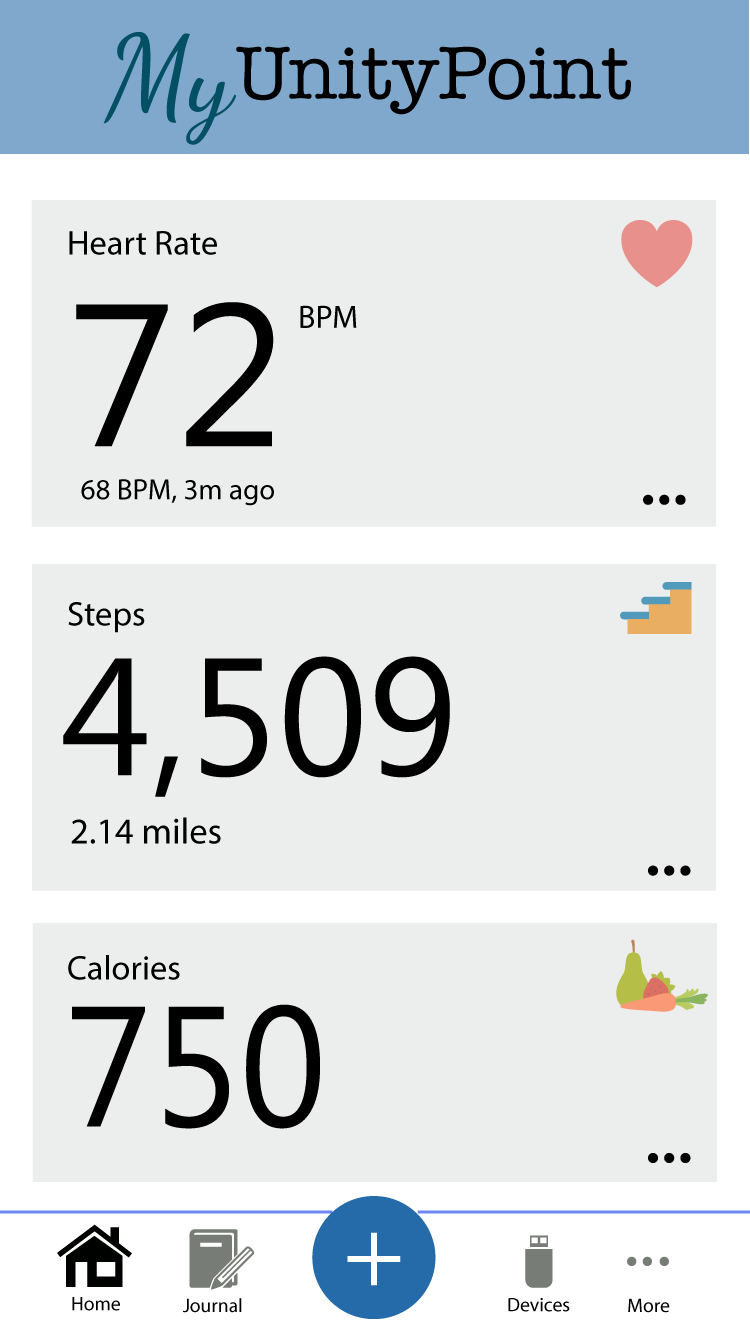
<http://developer.garmin.com/>

# 8 Appendices

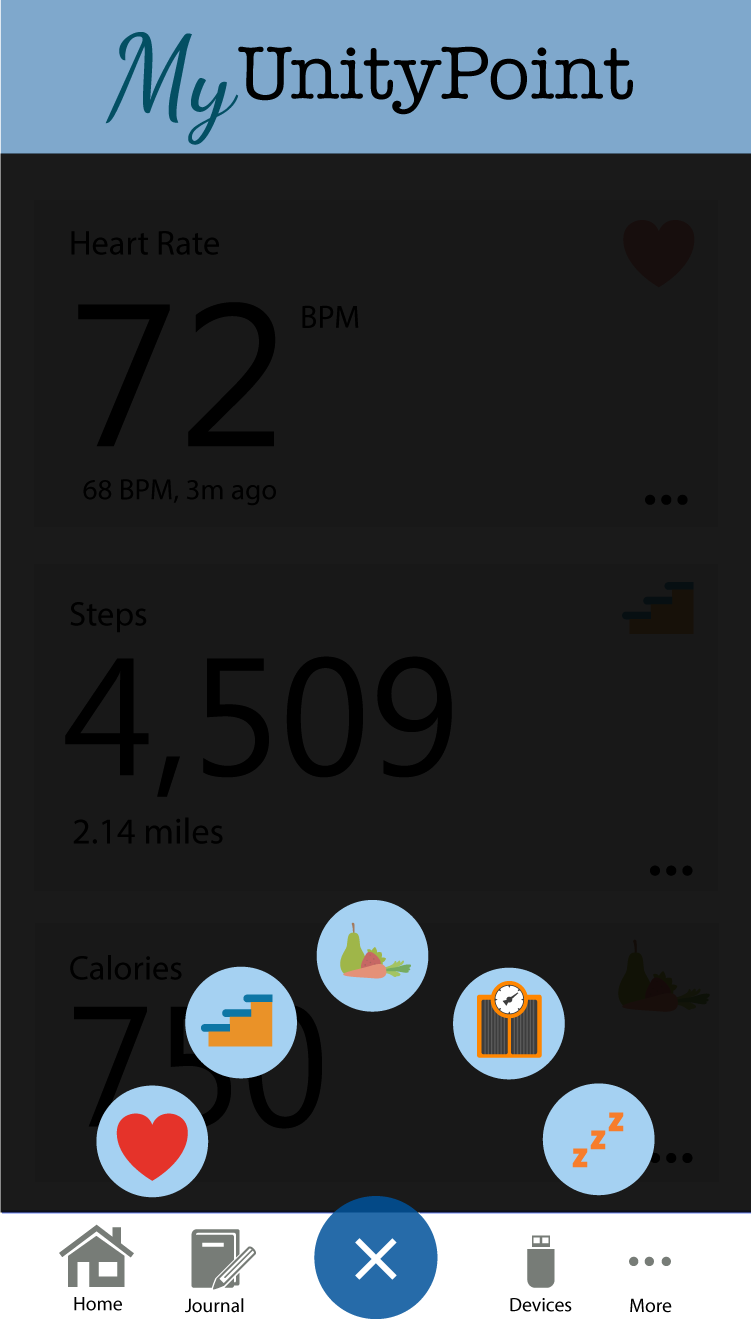
Attached are proposed screen sketches and UI designs for our app. There are several screens not depicted here, but they are still under development.



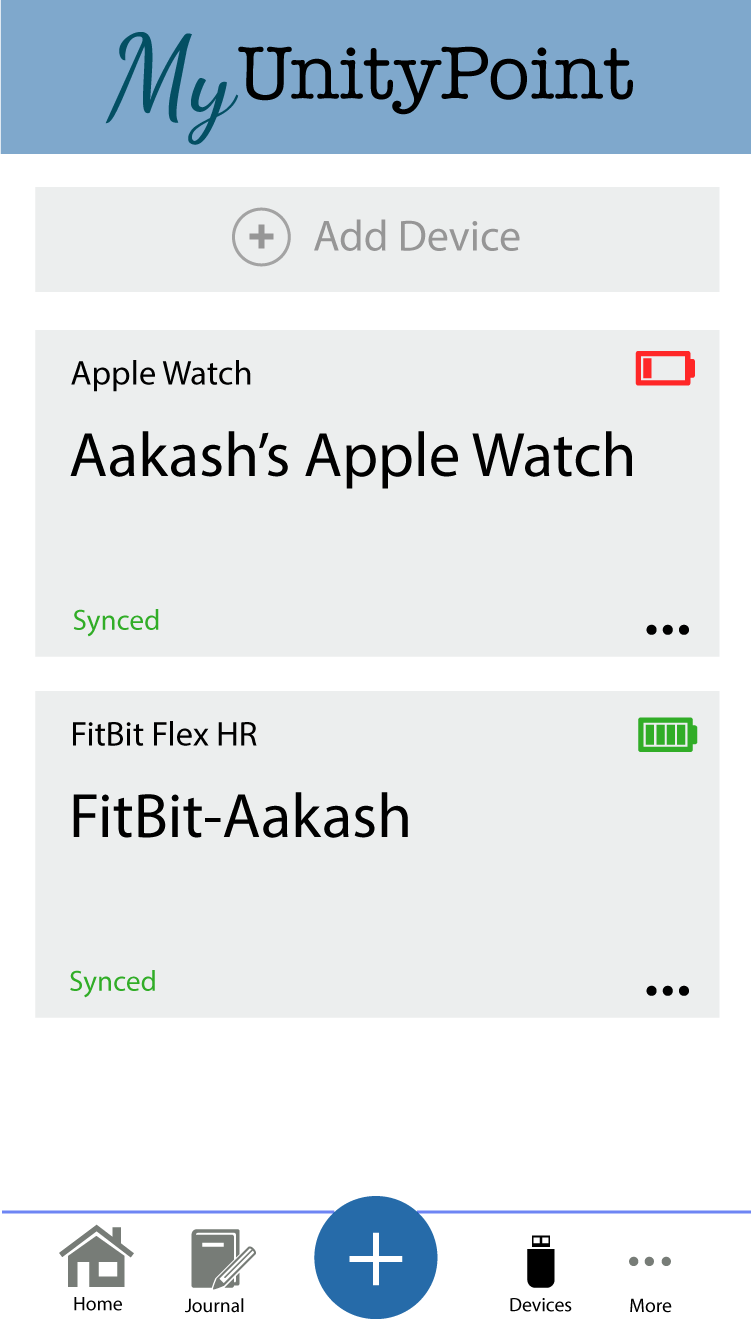
*Figure 8.1 - Login screen authenticating through Google Single Sign On, and Facebook Single Sign On*



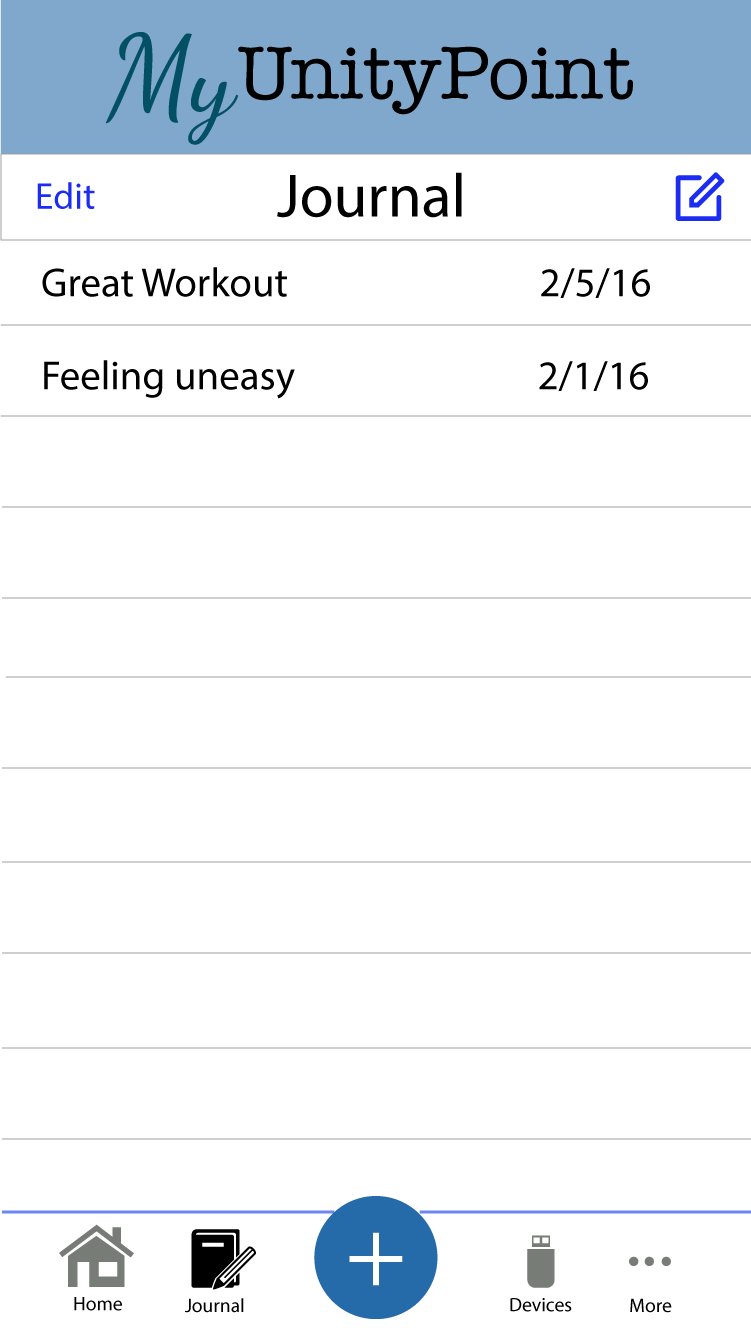
*Figure 8.2 - Dashboard view, representing various metrics*



*Figure 8.3 - Add button view. Basic information addition for various metrics*



*Figure 8.4 - Connected device view*



*Figure 8.5 - Journal view to keep track of user created content*