Senior Design Project II
EE 59867

Spring 2017

Rescue Senior

Project Report

(Code References/Instructions Included)

Submitted to:

Professor Jizhong Xiao

Group Members:

Faisal Farooq

Aaron Almonte

Md Rahman

Stephen Oakey

Contents

Background

Abstract

Our project aims to showcase a prototype that utilizes indoor location technology in conjunction with wearable health technology to help senior citizens in need of a health emergency.

What is Rescue Senior?

Rescue Senior is a health emergency solution for senior citizens. It utilizes health data from health tracking devices and to provide prompt emergency response it fetches indoor location data from Bluetooth Low Energy beacons. In case of an emergency, the system kicks in and trigger an alarm that allows rescuers to help patient in need.

Fitbit Health Tracking Device:

For this project prototype we are using Fitbit charge 2 for health tracking. We are focused on tracking continuous heart rate using Fitbit Intra day series data and other vital health metrics such as daily step count in real-time, calories burned and time spent in a certain heart rate zone.

Estimote Bluetooth Low Energy Beacons:

To localize a patient in large indoor environment in case of an emergency, we are using Estimote bluetooth low energy beacons. Beacons communicate via bluetooth protocol, which is very common in any modern technology. All the modern smart phones come equipped with bluetooth. Beacons help with determining precise location of an object or person. Beacons are particularly helpful in mapping indoor spaces. We are utilizing their indoor location capability to track and pinpoint a target in our case a person inside a building with accuracy of meters.

Statement of the problem

Senior citizens want to live independently and not be a burden on anybody. A study by *U.S Census Bureau, National Center for Health Statistics* shows the following:

- In terms of residency, about 29% or 13.3 million of non-institutionalized older persons live alone.
- The same data source reveals that most older persons have at least one chronic condition and many have multiple conditions.
- Another interesting statistics is about the increasing population of people aged 85+ years old is projected to triple from 6.2 million in 2014 to 14.6 million in 2040.

If we give seniors the freedom to live alone or independently, while making sure that if and when they need help in case of a health emergency they get it, we can better their lifestyle and give caretakers and family members the peace of mind.

This problem is also related to the elderly living at home alone. The nurses or caretakers come visit them during their scheduled hours but they don't have context on their health status. Also, if they happen to leave the premises the caretaker or family members are unaware of that, which in case of a health emergency can result in a serious risk.

To hedge the above situation and many other scenarios not listen here, we have designed a prototype that showcases that we can monitor elderly patients health around the clock while utilization of beacons allows us to be aware of their location indoors or warning that they are not indoors or on premises.

Family and doctors or caretakers can request on demand and have access to their location even indoors and health data in real time, which is vital in case of an emergency.

Our solution gives appropriate personnel the ability to track a patient's general health and location and empower the independence of patients and family members.

Rationale

The solution our team has implemented for this prototype is simple and fault tolerant. Using estimote bluetooth low energy beacons along with the most common wearable, Fitbit. It is a system build on the premise of simplicity and automation when and where possible.

The most common constraint is batteries. In our system, the batteries on the beacons last 3 to 5 years. Fitbit needs charging every 4-5 days, depending on the usage and frequency of activities.

Our system needs one time setup of beacons along with Fitbit health tracker. From the wristband we would be able to collect vital health statistics continuously throughout the day such as heart rate, steps taken, location and sleeping patterns. From the beacon we can track the presence or absence of elderly patients and also be able to locate them outdoors through GPS and indoors through beacons.

Design

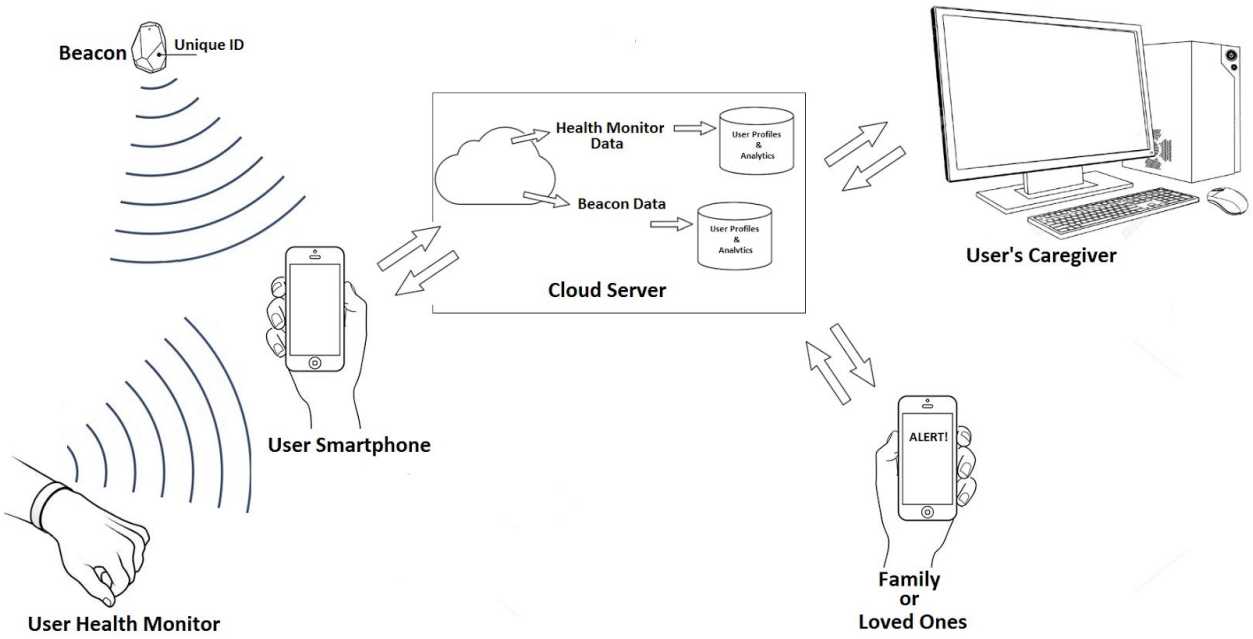


Figure 1: First tier design flow of the system

Our system is tied together using a microservices architecture methodology in software design. The tangible design is as follows:

Estimote beacons or any other type bluetooth low energy beacons can be placed in an indoors of a building at strategic locations to maximize the signal and their overall efficiency in terms of timely localized signal strength. The elderly patients' location will be determined based on the node they are closest to. This is determined using the RSSI (Received Signal Strength Indicator) value. The beacon scanning will be done using the mobile device, where Rescue Senior application is running in the foreground and is scanning for beacon signals at approximately 1 seconds interval. The mobile device will detect the beacon signal and send it to our cloud database, where we are storing and updating last seen values in real time. This centralized cloud real time database allows us to tell the last seen location of the patient or real time location to the rescuer in case of health emergency.

Fitbit health tracker continuously tracks a user's heart rate and other vital health statistics such as steps taken, calories burned and sleep patterns. Rescue Senior app collects this health vitals using Fitbit API in real time and displays in the mobile and saves the last updated values in the cloud database. This information is in return made available to the rescuer in time of an emergency.

Fitbit - Beacons - Cloud Database

Our microservices architecture design then allows us to hook into different APIs, databases and hardware design and collect this data and analyze it in the cloud. This information is made available to the rescuer, who can be a family member or caretaker, on demand or in when in case of a health emergency.

This combination of indoor localization in conjunction with wearable health tracking allows us to provide the core value proposition of this type of product and service, which is to give elderly patients the freedom to be be independent and give family members the peace of mind that if and when their loved ones need help, a timely complementary system is in place.

Technical Architecture

This section of the report will go over the technical architecture for this project. The idea was to used all the contemporary web and mobile technologies so that future development is easy.

Mobile Application

Big portion of development is Rescue Senior mobile application. This mobile application is essentially a front-end to everything the system does. To choose what technology or platform to use for mobile app development was big strategic task.

We chose Ionic, an open source framework for building mobile apps. Ionic is the beautiful, free and open source mobile SDK for developing native and progressive mobile apps. Ionic is fully cross-platform. We built our app using ionic with focus on android and compatibility for iphone and windows platform. Ionic provides premier native plugins to utilize native device features like Bluetooth, HealthKit, Fingerprint Auth, and more with Cordova/PhoneGap plugins and Typescript extensions.

Ionic is built to perform and behave great on the latest mobile devices with the best practices like efficient hardware accelerated transitions, and touch-optimized gestures. Ionic provides UI components for designing compelling user interface. Ionic uses Cordova or PhoneGap to deploy natively, or runs in the browser as a Progressive Web App.

Web Application - Server Side Code

Why do we need a web server or application?

One of the main features our system provides is the ability to trigger emergency alarm and alert rescuers, aka family members or caretakers, to help the elderly patients. In our design, the approach was to not have the rescuers install any component of the system. Give them the freedom to rescue the elderly patients without any technology overhead.

The simplest solution was to send them a text-message/phone call or an email in some on demand cases for communication with them. For this purpose, our web application comes into play. Our web application houses the vital info about patients. It has their profile and a dashboard page which features real time health stats such as heart rate, which is vital to know in case of cardiovascular emergency, and real time indoor location.

This allows us to have rescuers provide simply their phone numbers, so we can alert them with a text and this text contains a link to patient's info. When clicking on the link it opens up to a webpage that has all the requires real time and other patient info a rescuer might need.

Technology Stack for Web Application

We chose Ruby and Ruby on Rails web framework along with HTML/CSS and JavaScript for building web application. Ruby on Rails web framework. Basecamp, Github, Shopify, Airbnb just to name a few. For database we chose PostgreSQL. Our production servers are on Heroku (Free for now).

Our web interface is designed using Twitter Bootstrap 3.0. Bootstrap allowed us to make the website look beautiful and functional on different screen sizes. It also allowed us to include different graphics and user interface components.

Cloud Database

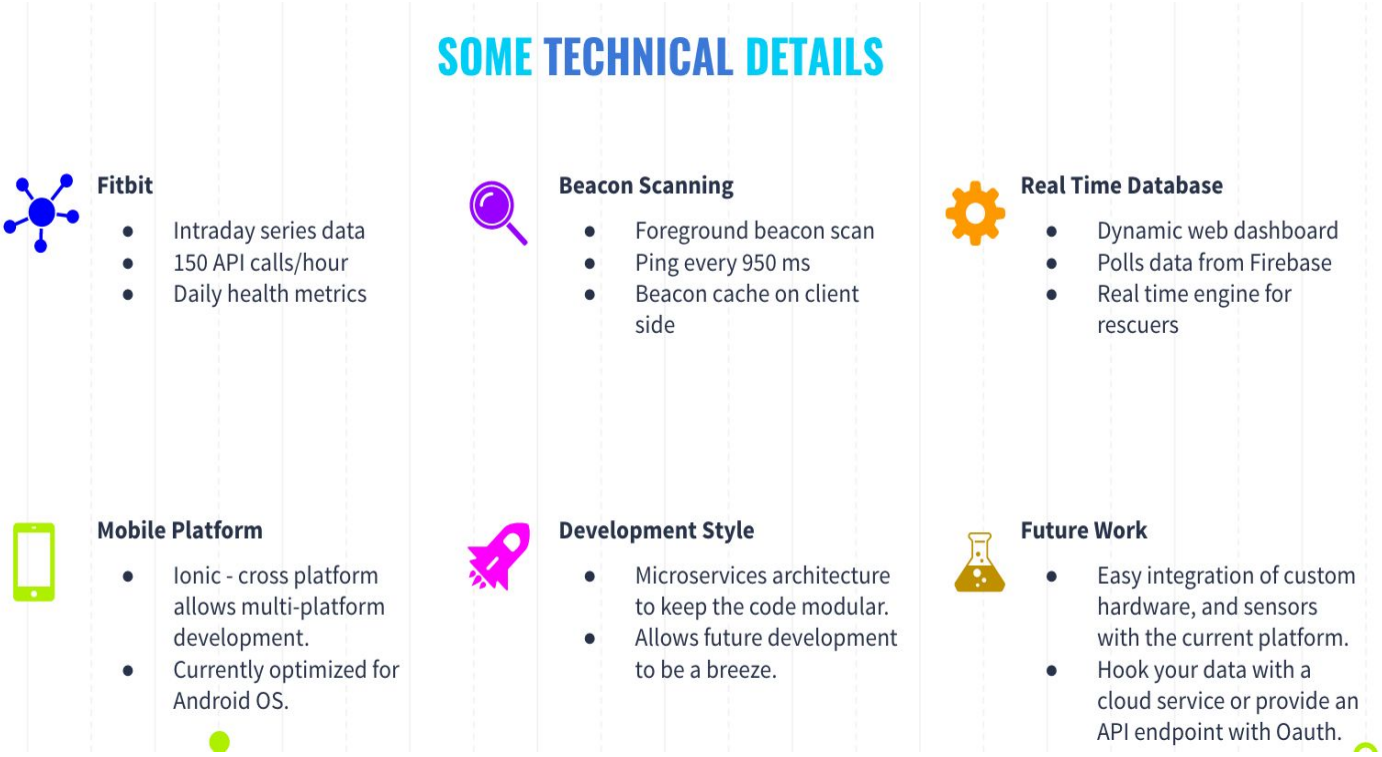
To store real time health data and indoor location data, we are using Firebase Realtime Database (Google company). Firebase allows us to to store and sync data with NoSQL cloud database. Data is synced across all clients, in our case mobile app and web app, in real time and it remains available when the applications go offline.

Figure 2: Firebase Realtime Database

Data is stored as JSON (JavaScript Object Notation) and synchronized in real time to every connected client. It is useful for cross-platform apps with iOS, Android and JavaScript for web to have one *Real Time Database Instance* and automatically updated with the newest data.

Development & Implementation

We have developed a mobile app and a web app along with setup of a real time cloud database to meet all the requirements of our project.



The system workflow is as follows:

- User is logged in the mobile application
- Authenticate fitbit using mobile dashboard or through web app
- Bluetooth and GPS are turned on
- Mobile application starts scanning for the beacons in the foreground
- Fitbit data is synced with mobile dashboard
- System scans for abnormal heart rate continuously
- In case of emergency, trigger an alarm and send request to rescuers
- Rescuers receive a text message with patient health data and indoor location
- Rescuers respond and help the patient

Flow Diagram

The final system flow diagram is shown below to articulate the system design.

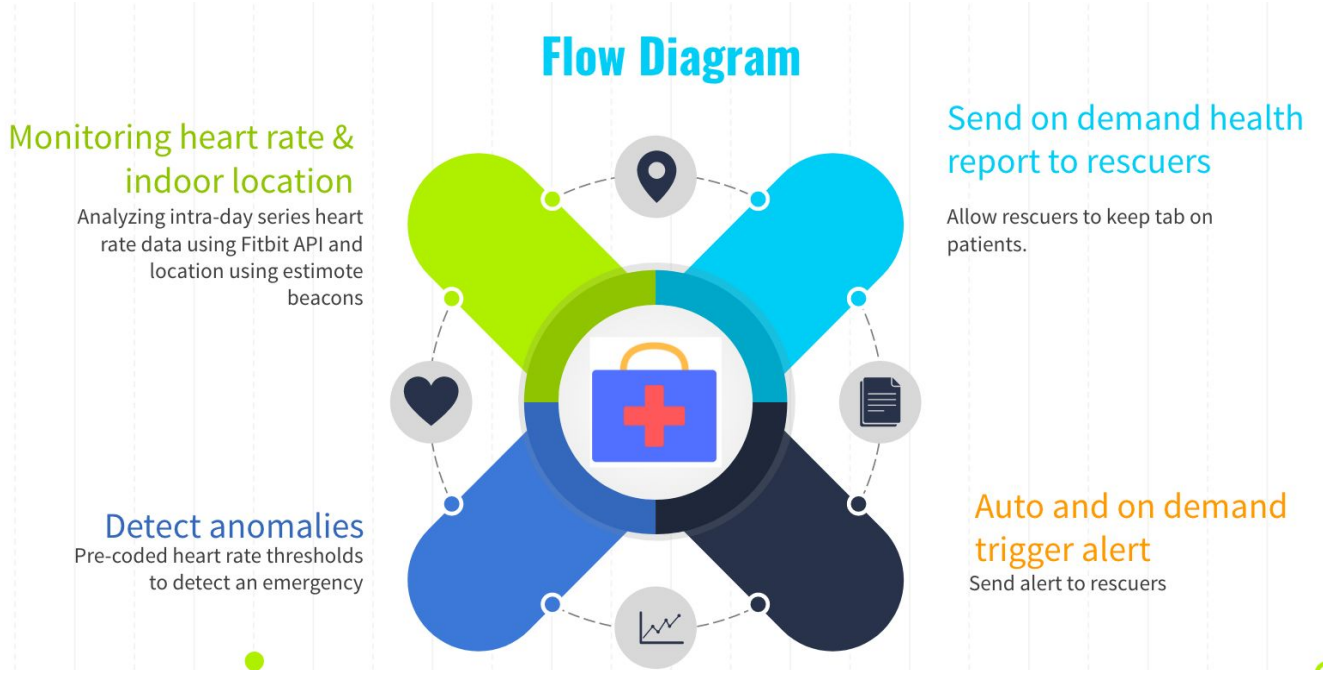


Figure 3: System Flow Diagram

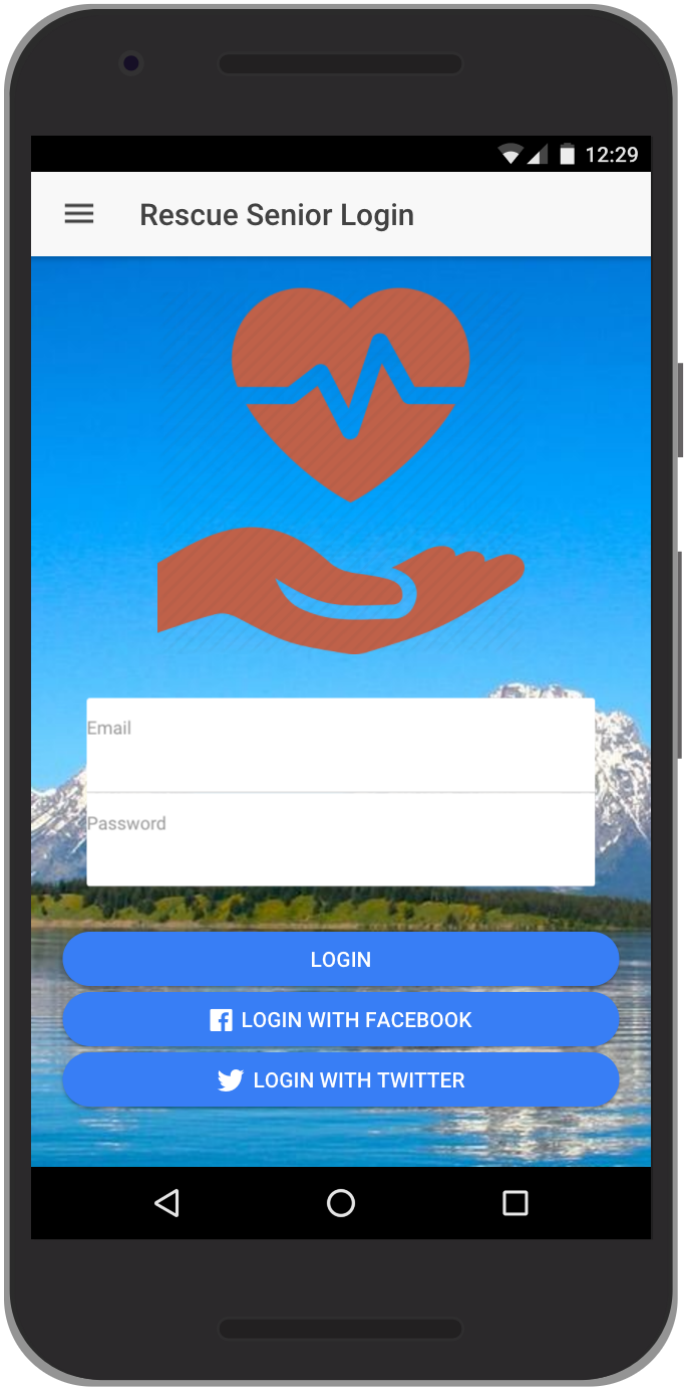
Mobile Application Details - App Screenshots

Below screenshots are from our functional mobile application. Code for this mobile app is live on Github at https://github.com/chaudhary27/Rescue-Senior-Mobile-App

Instruction for cloning the code and running on your computer or mobile are provided at the end of the report.

Screen 1: Login

Login screen features social login such as Twitter or Facebook as well as email/password login.

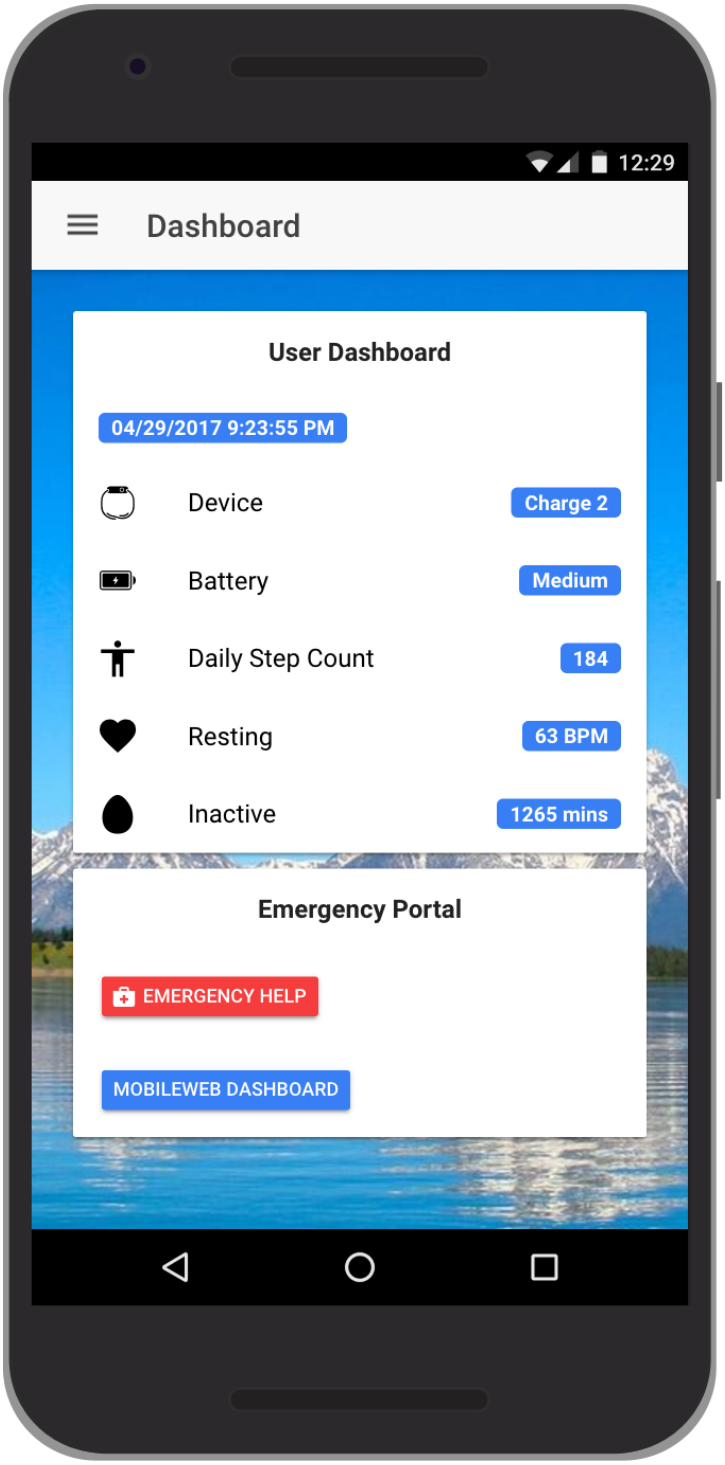


Screen 2: Dashboard

Dashboard Screen showcases:

- User daily metrics
- Wearable device details
- Emergency button
- Mobile dashboard button

On tap of “Emergency Help” button - a text message is sent to a rescuer with link for user health data and indoor location. This data is fetched real time and updates in the link as it becomes available.

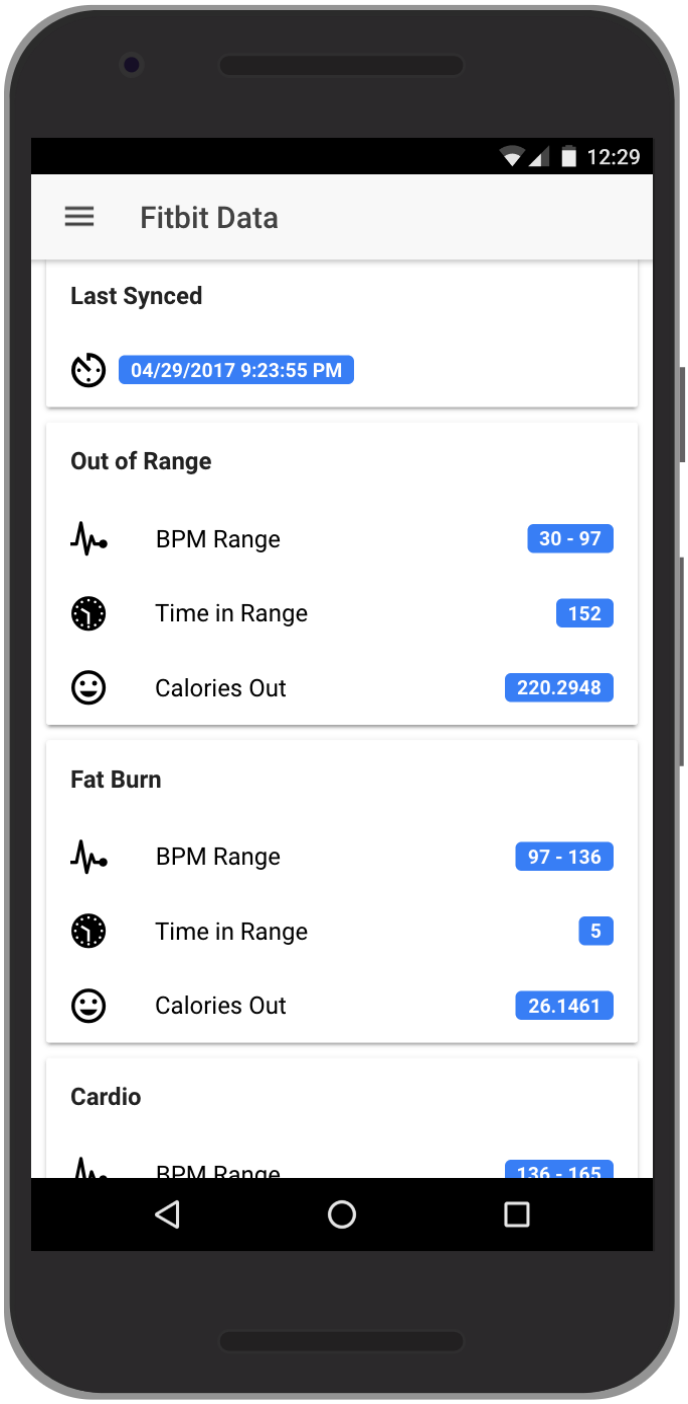


Screen 3: Fitbit Real Time Data

Fitbit Screen showcases:

- Timestamp of last sync
- Heart Rate Zones
- Calories burned in different zones

This data is fetched and updates real time using the Fitbit API.



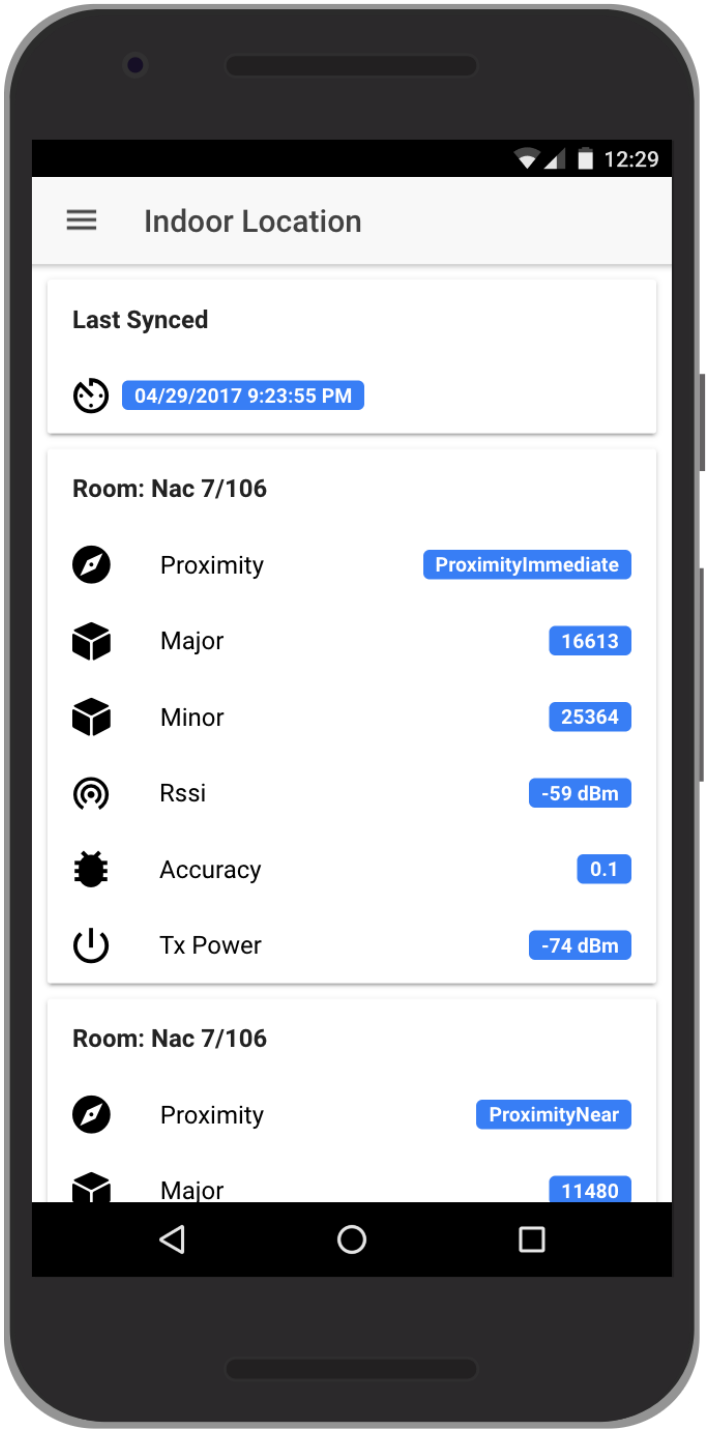
Screen 4: Indoor Location Real Time Data

Indoor Location screen showcases:

- Timestamp of last sync
- Room number where the beacon is placed
- Proximity in qualitative form
- Beacon RSSI
- Accuracy of data
- Tx signal power

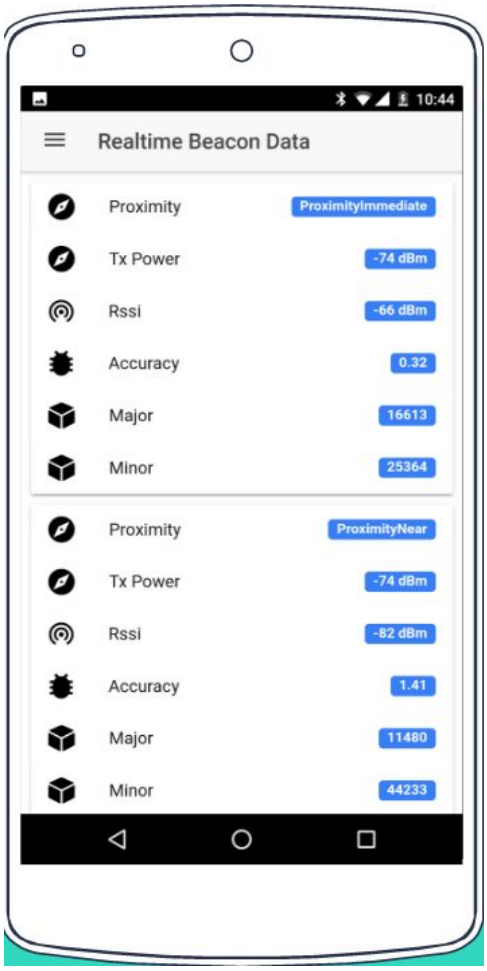
This data is fetched and updates real time using the native Cordova Bluetooth plugin in the mobile application. This screen data will update and in result will update the cloud database.

In case of no beacons in the vicinity. The last seen location will be shown here from the database. In case of emergency the last seen location with timestamp will be sent to the web server to display in the emergency text
link.

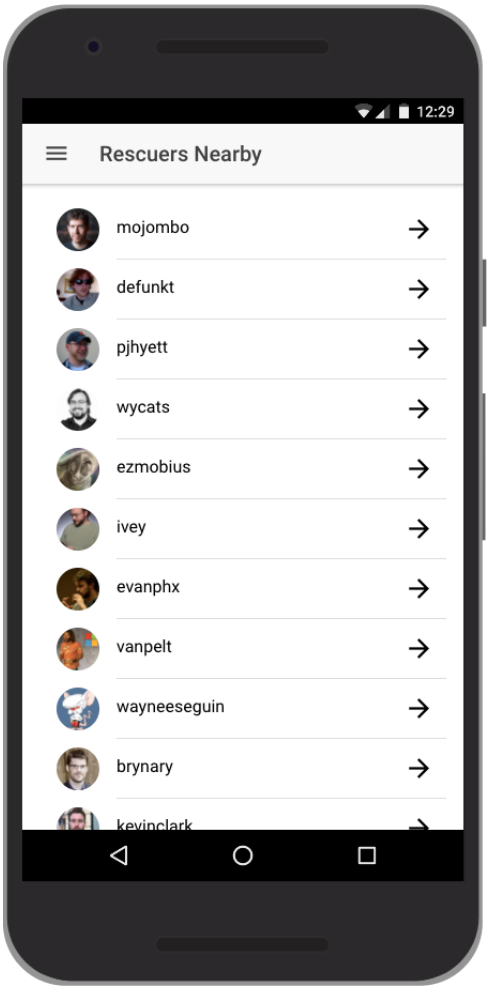


Additional features

- Foreground beacons scanning allows scanning of beacon even when the application is not running in the background
- Mobile Dashboard is mobile view friendly and provides rescuers with google maps navigation
- Cordova beacon plugin is compatible with different kinds of beacon manufacturers
- App is cross platform. It runs on iPhone and Windows phone as well.
- Rescuers nearby screen (not functional because of scarce data)



Foreground Beacon Scanning



Rescuers Nearby Screen

Web Application Details

Below are some of the screenshots from our functional web application, which is live on production at http://www.rescueseniors.herokuapp.com. The code for the application is live at http://www.github.com/chaudhary27/Rescue-Senior-Web-App

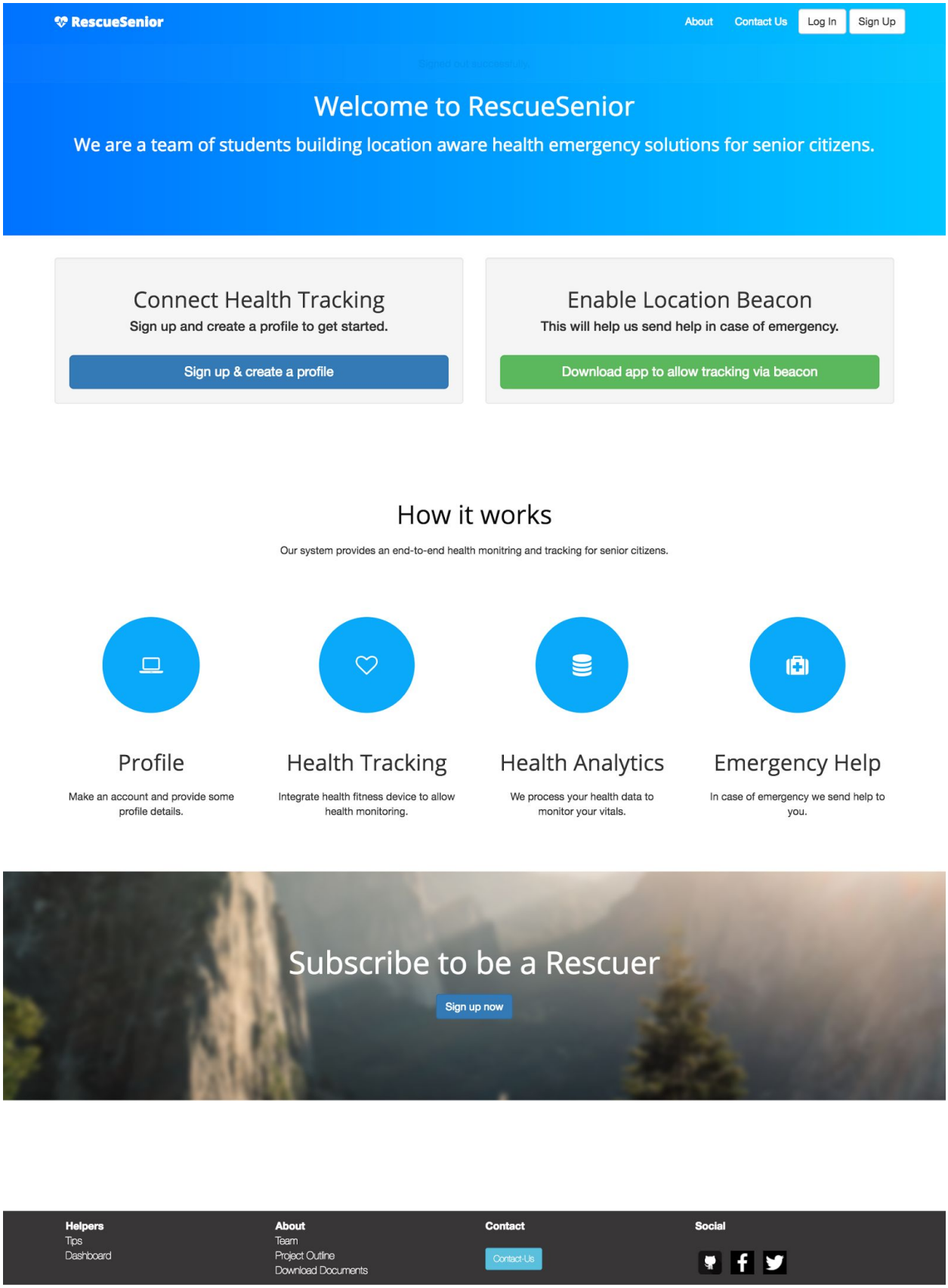
Instruction for cloning the code and running on your computer are provided at the end of the report.

Home Screen

When you head to the web address, a user is welcomed with home screen. On this screen you can sign up, if this is your first time interacting with Rescue Senior, or you can continue to log in and that will lead you to this same home screen with different actions on the screen. You can make a profile and authenticate application to use your fitbit data

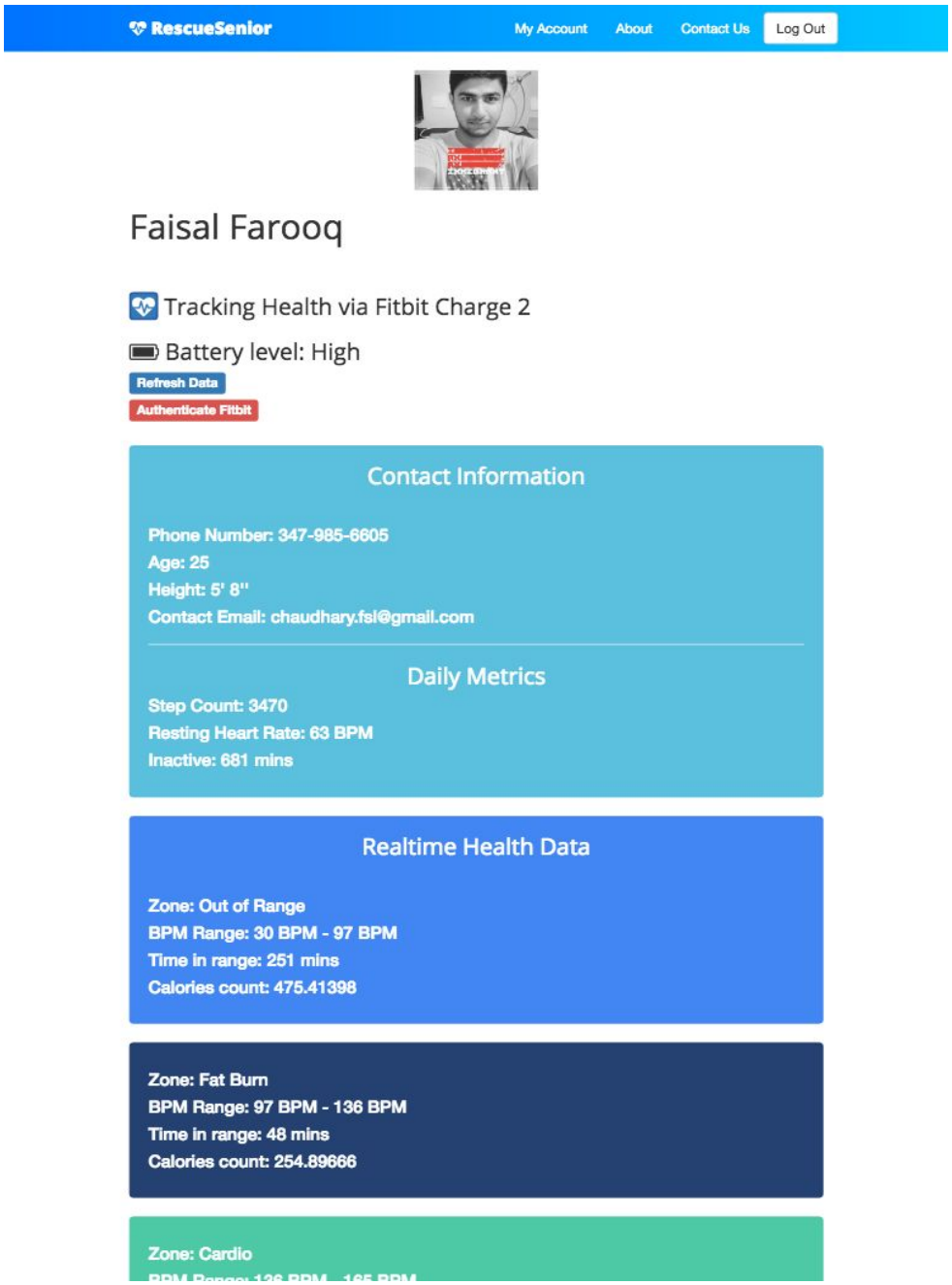
On the next page is a screenshot of our complete web page home screen. It features a How It Works section and different Call To Action buttons to lure the user to sign up and subscribe to the service.

[Continued on next page]


RescueSenior
About
Contact Us
Log In
Sign Up
Welcome to RescueSenior
We are a team of students building location aware health emergency solutions for senior citizens.
Connect Health Tracking
Sign up and create a profile to get started.
Sign up & create a profile
Enable Location Beacon
This will help us send help in case of emergency.
Download app to allow tracking via beacon
How it works
Our system provides an end-to-end health monitring and tracking for senior citizens.
Profile
Make an account and provide some profile details.
Health Tracking
Integrate health fitness device to allow health monitoring.
Health Analytics
We process your health data to monitor your vitals.
Emergency Help
In case of emergency we send help to you.
Subscribe to be a Rescuer
Sign up now
Helpers
Tips
Dashboard
About
Team
Project Outline
Download Documents
Contact
Contact-Us
Social

Mobile Dashboard

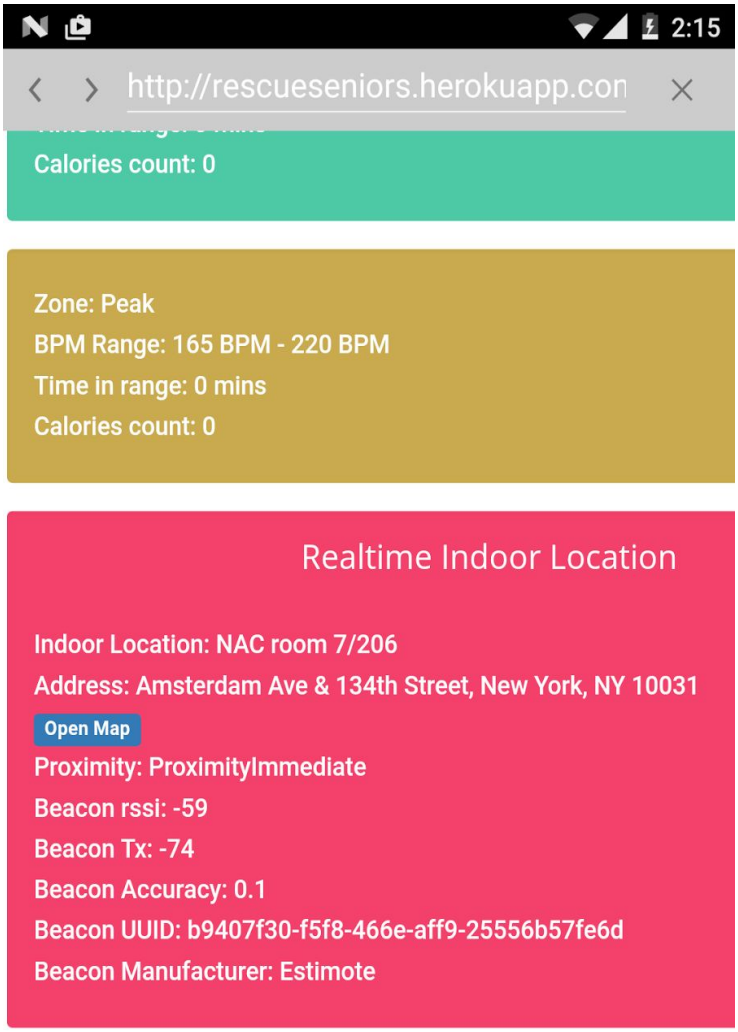
When you log in and provide your details like profile picture, phone number, contact email etc, you are directed to this dashboard, where you can authenticate fitbit using the button on the screen - this will prompt you to enter your fitbit login and password and redirect you back to this dashboard with showcasing your fitbit data and also if you have downloaded our app; your indoor location data is also displayed and updated in realtime.



User Dashboard: Showcasing realtime health data and indoor location

Realtime Indoor Location

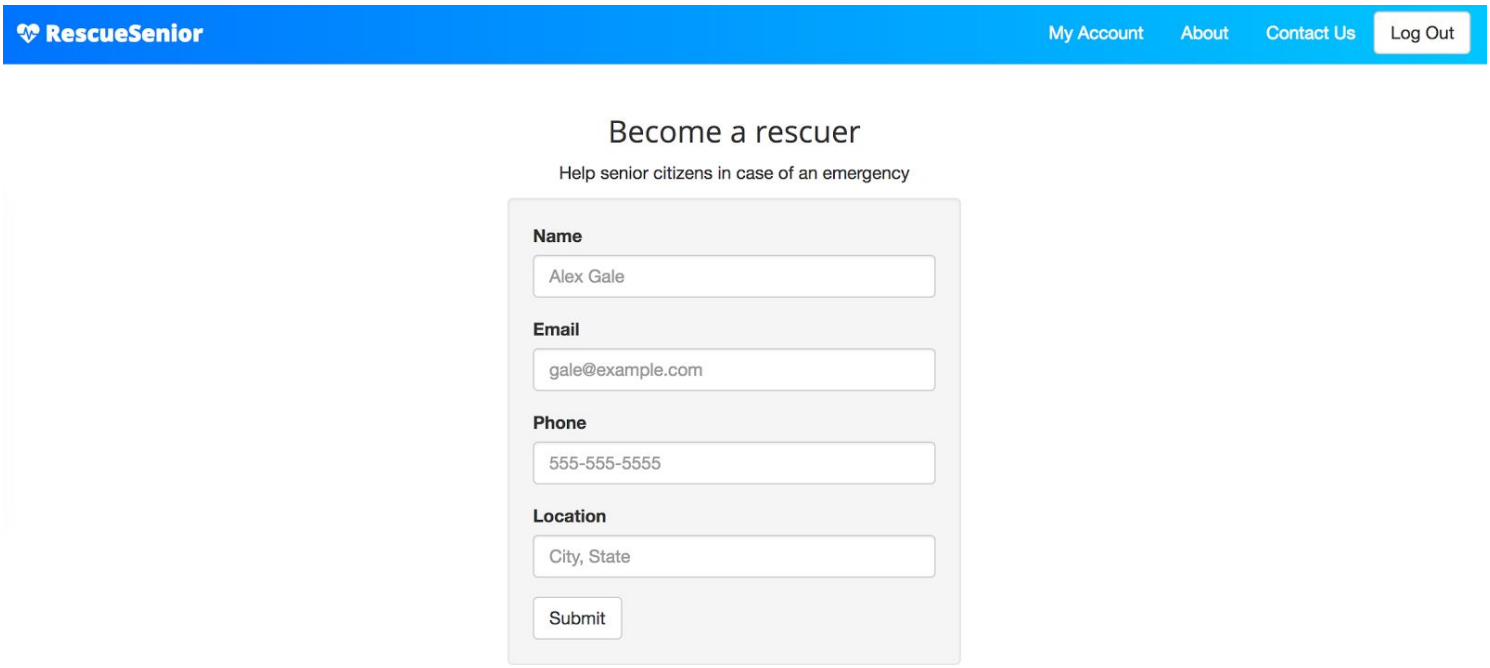
Dashboard also features realtime indoor location of a user. It fetches data from Firebase database and as the user moves around in a building or leave the premises, dashboard updates the data in realtime for that user. It also features GPS address for that user and navigation direction using Google Maps.
One thing to note here is that in case of emergency, a rescuer basically sees this page to determine the patient's location and glance over the health vitals.
This link is made available dynamically in the text, which is triggered on demand or in case of an emergency.



Mobile Dashboard: Realtime Indoor Location with Navigation

Subscribe To Be a Rescuer

Our web application also has this form where you can subscribe to be a rescuer. We built this so that in the future with this project, we can collect enough rescuers in a given geography to dynamically assign rescuer given a patient's location and proximity.



Rescue Senior: Subscribe to be a rescuer form

Web Application Summary

Rescue Senior web application serves as a backend service of our system. It talks to our Realtime Firebase Database and fetches indoor location data and health data in realtime. It preserves the last seen data on the screen. It caches this data on the client side. It allows authenticate using Fitbit and the whole Auth process is encapsulated in a different module, which is independent of Rescue Senior web application. The reason for this encapsulation is that if in the future, there is need to add more than on wearable or a different kind of wearable or quite possibly a custom made health wearable. All of that can be done quite easily without breaking the main application or refactoring code. Simply authenticate using a different wearable on the dashboard. Setup its data service in the cloud and fetch data and serve in the application.

Evaluation of Project Deliverables

Evaluation of project deliverables is as follows:

→ **Mobile Application**
Rescue Senior mobile application is fully functional. It is cross platform to maximize usage. It can be installed on an Android, iOS or Windows phone. Users can make an account and authenticate Fitbit wearable to fetch health data. Application scans for Estimote beacons in the background.

→ **Web Application**
Rescue Senior web application is fully functional. It is mobile optimized, which means it is friendly to look at on mobile screen sizes or other screen sizes such as tablets. User can make an account, create a profile and authenticate Fitbit.

→ **Cloud Database**
Rescue Senior is using Firebase Realtime database for mobile application and PostgreSQL database for web application. Most of the heavy load realtime data is routing through the Firebase database. This data is then served back in the mobile application and web application where needed.

→ **Production Environment**
Our web application is live on the internet, which means anyone with an internet connection can use our service. Our mobile app is not live in the app store yet. It requires a lot of configuration for production environment to be deployed publicly. We are in the process of receiving a permission from Fitbit to acquire users intraday series continuous data for realtime processing. Our team has already reached out to Fitbit development team and requested this permission. Fitbit is currently evaluating our application use case and architecture and then they will make a decision to allow usage of real time continuous health data such as heart rate or step count etc.

For future development, once the permission is granted, the mobile apps can be configured and deployed to the Google Play store and Apple App store.

References

[1] http://docs.ionic.io

[2] https://creator.ionic.io

[3] https://www.fitbit.com/charge2

[4] https://github.com/zokioki/fitbyte-rails

[5] http://developer.estimote.com

[6] https://www.heroku.com

[7] http://www.phonearena.com/phones/Google-Nexus-5_id8148

[8] http://www.heart.org/HEARTORG/Conditions/More/MyHeartandStrokeNews/Heart-Attack-or-Sudden-Cardiac-Arrest-How-Are-They-Different_UCM_440804_Article.jsp#.WFtpaLYrKHp

Acknowledgements

Our team would like to thank the following people and institutions for guiding us through the senior design project. Without these people, and institutions or companies we would not be able to complete our project in time. They all were instrumental in every step we took.

People

- **Professor Jizhong Xiao:** Professor Xiao worked with us patiently and helped us stay on track. He listened to our weekly updates and gave valuable feedback in terms of project implementation and timely completion.
- **Professor Zhu:** Professor Zhu gave helpful lessons last semester and gave valuable feedback during demo days. He organized an expo where we were able to showcase our project to target audience and gain pragmatic feedback from them.
- **Bing Li:** Mr. Li was helpful at the beginning of the project to help us steer in the right direction in terms of technical architecture.
- **Professor Edward Baurin:** For project management classes and guidance. We are confident we will be using the knowledge you imparted for years to come after graduation.

Institutions

- The City College of New York - Grove School of Engineering for providing us the academic and research facilities and environment where we were able to conduct our research and find mentors.

Open Source Software

- Ionic Cross-Platform Mobile Development SDK
- Github Student Pack
- Heroku Free Version for Beginners
- Fitbyte Gem by Zokioki on Github
- Cordova Plugins for Native mobile features

Code

Our code for the whole project is on Github and it can be downloaded to use and replicate our project.

Github Repositories:

- **Mobile App:** https://github.com/chaudhary27/Rescue-Senior-Mobile-App
- **Web App:** https://github.com/chaudhary27/Rescue-Senior-Web-App
- **Fitbit Auth:** https://github.com/chaudhary27/Rescue-Senior-FitbitAuth
- **Cloud Database:** https://rescuesenior-d31bf.firebaseio.com

The instructions for cloning the code and running on your system are as follows:

Note: These instructions are also available on corresponding github code repositories.

→ **Web App**

Run the application locally

1. Clone the repo
2. cd to the app dir
3. Run **bundle install** to download all the gems
4. Run **rails s** and navigate to localhost in the browser

→ **Mobile App**

Run the application locally

1. Clone the repo
2. cd to the app dir
3. Run **npm install** to download all the dependencies
4. Run **ionic state restore**
5. Run **ionic build android**
6. Run **ionic run android**

To run the app in the browser on your computer do the following

1. Run **ionic serve** and navigate to localhost in your browser
2. Run **ionic serve -l** to see the app in mobile platform

End of Report.