Project Requirements Document v1

Team #stub In Collaboration With Teladoc Health

Benjamin Lee
Dylan Kupsh
Paul Kuang
Jonathan Xu
Rohan Aren

Introduction

Background

Teladoc Health is a company that focuses on enabling virtual healthcare. OVer the course of a year, Teladoc conducts over 10M telehealth consultations for general med, behavioral, dermatology and acute care. In recent times, reliable, quality telehealth services are needed more than ever. To accommodate this growing demand, Teladoc Health has proposed the development of software that interfaces with consumer health peripherals and records the user's status so doctors can have readily available data rather than requiring patients to come in for a physical checkup.

Telehealth consults offer several logistical advantages over in-person consultations. Due to neither the physician nor patient need travel, virtual meetings provide easy and convenient scheduling. Additionally, the Medical equipment and hospital resources used in regular doctors appointments are able to be reallocated for other tasks. Limiting the number of in-person hospital visits also prevents the spread of disease; a beneficial side effect for patients who would normally be too afraid to come into the hospital. During the Covid-19 Pandemic, these advantages have created conditions for the usage of telehealth consultations to skyrocket.

However, telehealth consultations have one main drawback: currently, remote consultations are limited by the lack of medical instruments that would normally be accessible in a doctor's office or other physical location. A key aspect of a normal doctor's visit is a general wellness checkup, which can involve measuring heart-rates, respiratory rates, or any other physical assessments. Without precise medical instruments available during a telehealth call, doctors have a limited amount of objective information they have to make a preliminary diagnosis, offer treatment suggestions, or provide treatment plans. Physicians often have to rely on patient testimony alone to understand a patient's situation, which can be unreliable. Consumer health peripherals, including Apple Watches, FitBits, and mobile phones, provide physicians with an alternative method of gathering objective health data. These devices hold a lot of latent, useful medical information that can be leveraged to offer physicians some tools to gain better insight into a patient's situation even over a telehealth call. By taking data from these peripherals and relaying it to Taladoc's patient-physician

communication software, we hope to create a convenient solution that brings virtual telehealth consultations closer to the experience of a physical, in-person variant.

What is the problem?

In the virtual environment, there is no way for physicians to directly take a vital reading or other physical assessment of a patient's condition. The only metric by which a physician can make an initial diagnosis or suggest a treatment plan is patient testimony, which can raise questions of reliability and accuracy. Patients may not be aware of terminology to describe a certain symptom, or might miss subtler indicators that would normally be picked up by hospital equipment. Even if a patient has medical equipment at home, there is always a chance of misinterpreted or misspoken information that can jeopardize the accuracy of the device's measurements. A direct interface between telehealth software and at-home peripherals would alleviate these concerns and provide physicians with some degree of certainty that the information they are receiving is objective and direct from the source.

Connecting directly with consumer health peripherals also offers some information that cannot be measured in the hospital. For example, a physical exam can only record the patient's status at that moment, and there is no way to know the patient's condition over a period of time. Consumer health peripherals like the Apple Watch and Fitbit provide this service, but information stored by these devices is not readily available for doctors to access. Ideally, our project would provide an easily accessible interface for physicians to view data not just at the time of a telehealth consult, but leading up to and possibly even following a meeting, such as to track the effectiveness of a treatment plan in the days or weeks after a consultation. There is also potential for further back-end analysis and processing of peripheral data after collection, making many useful applications of this information possible.

How is the problem addressed today?

During virtual consultations, if physicians want to collect patient vitals, a patient must either manually relay this information or rely on third-party software. Some forms of data, such as an EKG, cannot be manually described. Additionally, Human error in measuring or relaying any information can cast doubt on its reliability. Moreso, while

third-party services with the purpose of recording and transmitting this health data exist, they are often costly and not directly integrated with the communication software, making them clumsy and inconvenient for physicians to use. There are also no in-depth means to analyze this data besides a manual inspection.

Goals

The project's main goal is to bridge the gap between physical patient-physician interactions and telehealth consults provided by services such as Teladoc's Solo application. A major step in bringing virtual consultations closer to the physical experience is allowing doctors to take real-time measurements of a patient's condition, collecting information such as heart rate, respiratory rate, and temperature. Additionally, due to their continuous collection of data, interfacing with health peripherals has the potential to offer information not normally available during a doctor's visit, such as health trends over time both leading up to an interaction and following a prescription or treatment plan. Finally, we hope to provide meaningful analysis of this data both during an appointment and over its collection period.

How will it be done?

The project consists of two main components: a mobile application implemented with React Native to receive data from consumer health peripherals, and an interface between this mobile application to Teladoc's communication software in order to make this information available to healthcare providers. Because team members have access to different mobile device OSs, we have decided upon React Native as a convenient middle ground that can be easily deployed to both iOS and Android devices. As patients go about their daily lives, their peripherals continuously collect information regarding their activity level and vitals. Using existing peripheral APIs and Teladoc Cloud, our application will allow users to consolidate their vitals data in a single location, from which physicians will be able to directly access objectively collected health information over a virtual format.

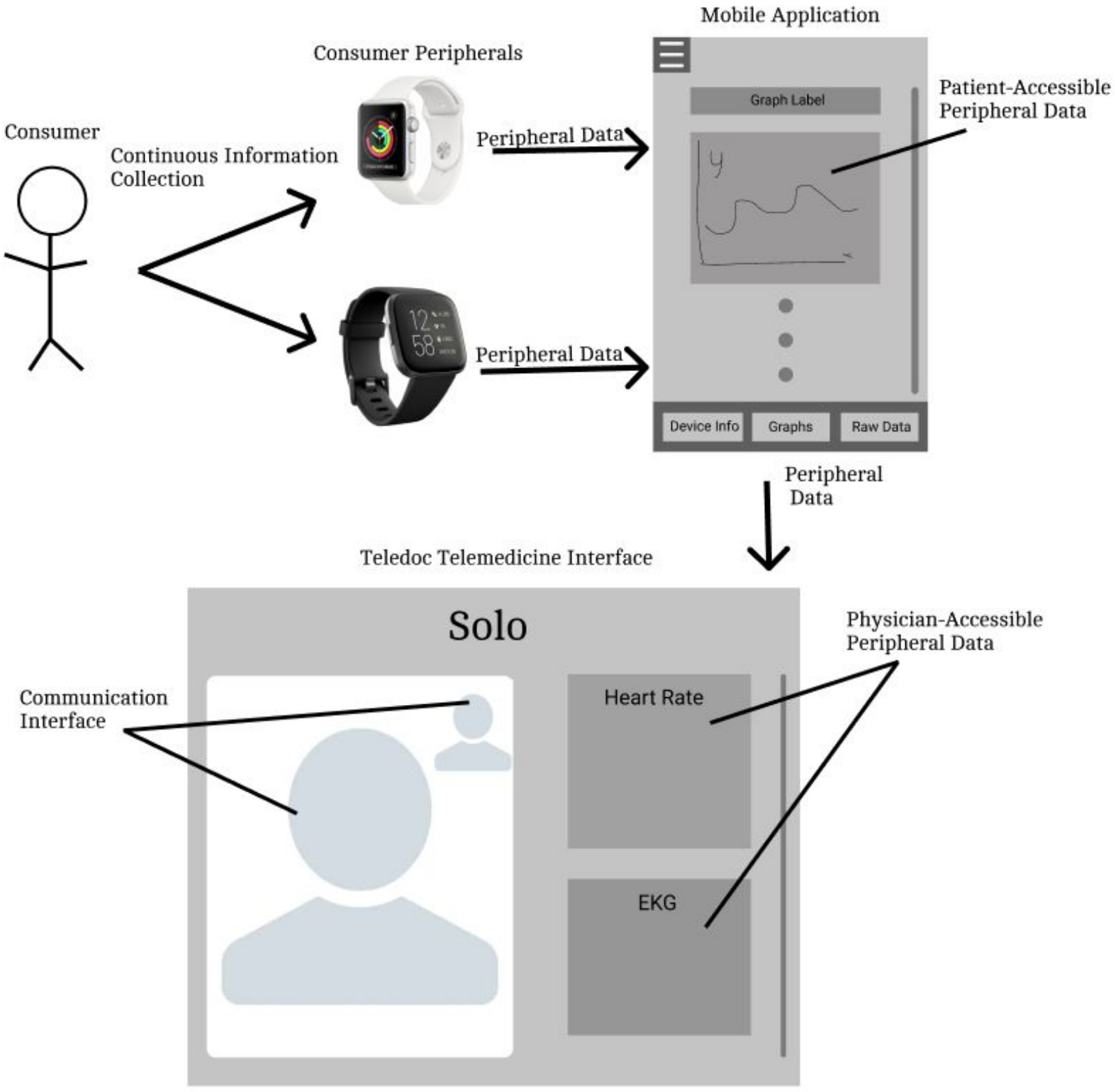
Assumptions

- Peripheral data is reliable and accurate

- Patients utilize their peripherals enough to provide a minimum amount of information
- Patient data will always be accessible through peripheral APIs

System Architecture

Our mobile application will collect data from consumer peripherals by utilizing their respective APIs. From this application, we will interface with Teladoc's communication software to display this data during patient-physician interactions.


Mobile Application
Consumer Peripherals
Consumer
Continuous Information Collection
Peripheral Data
Graph Label
Patient-Accessible Peripheral Data
Peripheral Data
Device Info
Graphs
Raw Data
Peripheral Data
Teledoc Telemedicine Interface
Solo
Physician-Accessible Peripheral Data
Communication Interface
Heart Rate
EKG

User Stories

As a **Patient**:

Story/Github Issue	Testing Criteria
I want to have control over what data to send so that I can feel like my data is private. Issue Link: https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/1	Scenario: User has synced vitals data. ● Users will select what vitals data to send over to Solo. ● In the Solo session, ensure that only the approved data is sent and displayed on both sides. Scenario: User has not synced vitals. ● The user will be prompted to sync data before attempting to send vitals to Solo. Scenario: Data fails to send/fetch. ● Check network connectivity and prompt the user to check connection if offline. ● Otherwise, retry the sending/fetching operation once. ● If the send fails again, inform the user of the error with “Failed to send/fetch data.”
I can use our app to find my health peripheral information such that I can get a clear understanding of recent vital trends. Issue Link: https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/2 https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/3	Scenario: User has synced vitals data. ● Navigating to an information page shows a history of vitals information in an appropriate format, such as a graph. Scenario: User has no synced data. ● The activity should show text that guides the user into syncing their vitals data.

I want to record my blood pressure changes during the day, so that my doctor can view my health status and have a more accurate diagnosis. Issue Link: https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/4	Scenario: User has network connectivity, has blood pressure measurements from peripherals. ● App should retrieve blood pressure history when attempting to sync. ● This information should be viewable to the user and over Solo if permission was given. Scenario: No blood pressure data found. ● The information page should display "No blood pressure measurements found."
I want to be able to send types of data that concerns me and double check if my information is correct. Issue Link: https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/5 https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/6	Scenario: User has outlier data, which is incorrect. ● Users will be notified of outlier data in the information page, under a Concerns section. ● Users can add a correction to the data, which will be marked as a correction in Solo. ● Alternatively, users can note down the concerning data within the app as a reminder in their next appointment. Scenario: User has no outlier data. ● Ensure that the Concerns section is hidden in the information page.
I can track their symptoms as monitored by health peripherals, and be informed by the app if they are displaying signs that they should seek medical attention thus reducing the load on healthcare professionals by preventing false alarms, and hastening patient-physician interactions by making both parties aware of symptoms before a consultation.	Scenario: Data shows concerning symptoms. ● Display a notification from the app: "Concerning vitals data found." ● Allow users to note down the symptoms for their next appointment.

Issue Link: https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/7	Scenario: Concerning data is clearly impossible. ● Ex: If the user suddenly records a weight of several thousand pounds. ● Filter out the data when it is synced. ● If newer, more reliable data is not present, prompt the user to record it again.

As a **Doctor**:

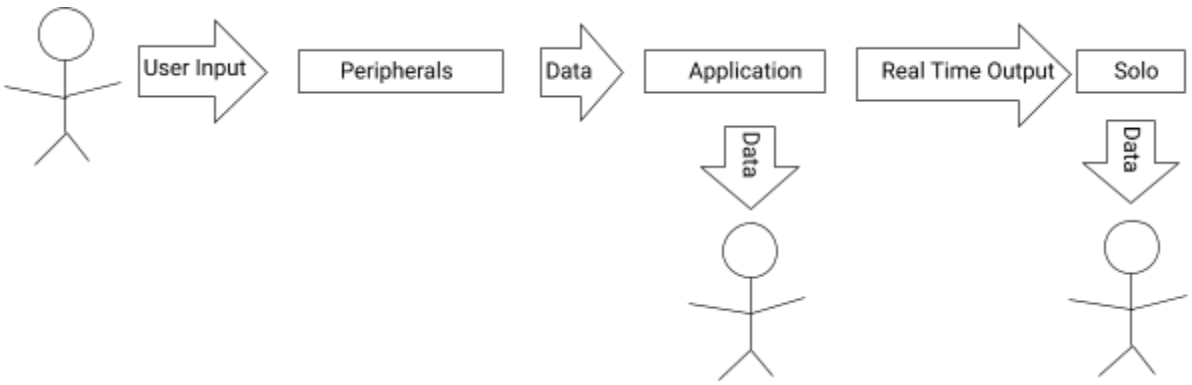
Story	**Testing Criteria**
I want to check if my patient is doing the daily exercise correctly, so that I know what exercises patients are doing the best. (Chiropractor)	Scenario: User is prescribed an exercise routine. ● The user can record their exercise in a log and send it over Solo. ● The doctor should be able to adjust the routine and send it back to the user.
I can use our app or interface to gather information on a patient's vitals via a peripheral over a remote call, allowing medical professionals to obtain objective data without needing an in-person consultation. (Physician)	Scenario: Doctor requests information from patient. ● Doctor can prompt the patient to send data through the app from Solo. ● The patient can choose what data to send to Solo. ● Doctor's view of Solo should listen for new data and update when the upload is successful. Scenario: Data send fails. ● If the data does not send within 1 minute, update the doctor's view of Solo to display "Data failed to send."

I can use our app to gather basic information on the patient such that I can skip performing basic tests and avoid biased answers by the patient. (Physician)	See above.
I want to see concerning vitals information in a prioritized window so that I can prioritize discussion of important details with my patient. Issue Link: https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/8	Scenario: The app detects concerning vitals data prior to sending. ● The app should note concerning items as a higher priority. ● The doctor's view of Solo should highlight the prioritized information with a [High Priority] tag.
I want to check patient metrics with baselines and averages to better understand the severity/level of their health. Issue Link: https://github.com/benjaminlee0/InterfacingConsumerHealthPeripherals/issues/9	Scenario: Metrics within healthy ranges. ● Display patient vitals info, with a message that states "Within healthy ranges." Scenario: Patient has metrics outside healthy ranges. ● Display patient vitals info, with a message that states "Data outside healthy range of X to Y" where X and Y are the limits of the range. ● Change formatting of unhealthy data to stand out.

System Models

System Requirements

- The mobile application shall continuously collect certain data from consumer peripherals without any action from the patient.
- Data from all interfaced peripherals shall be displayed to users in the application.
- During a telehealth consult, data from the application shall be continuously accessible to physicians.



Appendices

Technologies employed:

- React Native
- Apple HealthKit
- FitBit Web API
- OAuth 2
- Solo