INDY-1 Fitness App

Software Requirements Specification

CS 4850 Section 2

Spring Semester 2024

Professor Perry

02/13/2024

Luke Zeches

Adam Borowski

Michael Bolnik

Table of Contents

1. Introduction

1.1 Overview

With our project, we plan to make a fitness app that makes exercising fun and engaging. While users are on the app, it will use a GPS system to track their location. With this information, our app will count the distance traveled and save it into a database. To gamify this system, there will be multiple destinations that one can virtually travel to. For example, if a user wished to simulate a trip across the United States, our app would track the distance traveled and show how far along they were. It would also give milestones, such as reaching the Grand Canyon. While on a trip, the app will show your current location on the path using Google Street View. Every user will also be able to see their total distance traveled and the number of destinations reached. For security reasons, every user will have to create an account with a username and password and must login when opening the app. Our app will be built for Android devices using React Native, to allow for future cross-platform development. It will start with a simple UI and a few separate locations to visit but can be updated over time to add more destinations to track.

1.2 Project Goals

Many people do not get as much exercise as they should, due to time constraints or lack of motivation. We hope to solve these issues with our app. By giving users goals to work toward while exercising, people will feel more incentivized to walk or run in their daily lives. There are smaller goals that will contribute to our main goal as well. These include the aims of interface design and systems. The systems we plan to build will make it easy for users to begin a trip and track their progress which will push them to use our app more often. Our user interface will play into this too. The design will make our app easy to use at a glance with few learning curves. More details on the user interface will be discussed in a section below.

2. Design Constraints

The design of our project is formulated under a few set constraints that we will work around and use as a baseline. We will need to implement a few tools in order for our functionality to work properly and give us the full systems scope that we are looking to put together. For our database, we will need to use SQL as our programming language and SQL Server in order to implement our database functionalities and store the user data that is needed. The option to add additional security to our database through the form of encryption is also possible in order to safehouse and data that might need to be kept confidential through any agreements that are made with customers. In order to sync the database to the app, we would most likely have to use a REST API to connect to the database and allow for the proper secure connection to our database.

The second option for our database constraints would be to use the built-in React Native functionality for creating a local database which would allow for an alternative form of data storage built into the framework. In terms of language constraints, we will mainly be using SQL and Javascript, with our runtime environment being Node.js.

We will also be using a Google Street View API in order to develop our mapped representation of the source and destination locations to show the user-chosen path that they would like to explore during their fitness journey. We will also be constrained to the chosen app or tool that will be used in order to deliver our active fitness data, which will be transformed into a usable form of media on our user interface.

In terms of any corporate or regulatory policy restraints, we would need to work around any issues regarding the usage of data in terms of both fitness and personal data and make sure to keep that data secure and inform the user what data is necessary for our app to provide the full functionality that is applicable.

2.1 Environment

Our app will be developed for Android version 13.0. The platform we will use to code our app is React Native v. 0.73.4, the latest long-term support version. Front-end and back-end development will be done in Android Studio using an emulator to perform testing. Version control will be done using Git, with the remote repository hosted on GitHub.

2.2 Project Scope

Our project will integrate multiple systems in order to accomplish our goals. For our first release (the prototype) we plan to have the basic requirements such as a login screen, account settings, and the ability to start a trip. This version would track the user's distance traveled and use this information to determine how far they have traveled along the planned trip's route. A progress bar would be implemented that shows the percentage of the trip completed.

Our final release will include a database that stores user information about their profile. This includes total distance traveled and the number of trips completed. Instead of a progress bar, we will update the trip screen to show a map of the entire trip along with the user's location on it. We also plan to integrate Google Street View which will show the user their surroundings on the trip.

3. Functional Requirements

3.1 Account creation screen

        3.1.1 Description: allows the user to create a personal account to store their data. The user can choose to log in to an existing account if they already have one. The provided data is used to create an account in the database.

        3.1.2 The following information will be provided by the user:
                3.1.2.1 First and last name
                3.1.2.2 Email address
                3.1.2.3 Username
                3.1.2.4 Password

3.2 Login screen

        3.2.1 Description: allows the user to enter a username and password to verify their identity and gain access to their account. The user can choose to create an account if they don't already have one.

3.3 Allow user to create new trip/continue trip

        3.3.1 Description: users can start a trip if they aren't on one already. If the user is currently on a trip, they can press a button to continue their progress.

3.4 Track and plot user's movement

        3.4.1 Description: our app will track how far a user traveled during a current time frame. This information will be used to determine their progress on a current trip

3.5 User view of current trip and progress

        3.5.1 Description: users will be able to view information about their current trip. This includes how far they've traveled on it and elapsed time.

3.6 User view of surrounding area at plotted location

        3.6.1 Description: Users can view their surroundings based on where they are on the trip.

3.7 User view of profile information

3.7.1 Description: the user will be able to view their personal information associated with their profile, as well as statistics about their app usage. These statistics include information about previous trips completed and total distance traveled.

3.7.2 The user will be able to view the following information about their previously completed trips:
   3.7.2.1 Start and end location
   3.7.2.2 Start and end date
   3.7.2.3 Total distance traveled
   3.7.2.3 Total time spent exercising

4. Non-Functional Requirements

4.1 Usability

In terms of usability, there will be multiple requirements that are implemented in order for the user to have an easy and efficient experience interacting with our app. We will look to have a very simple log-in screen that looks to verify the username and password information in order to link to the account data. The opening screen will be devoid of any unnecessary information and be free of any distracting graphics with a simple, sleek look. The simplicity of the opening screen will also promote error avoidance as there will not be many ways for the user to cause an error, outside of possible issues in input information that will be dealt with by error handling in our software. Maneuverability will be simple with the only option from the log-in screen being to hit a button that will allow for the movement to the next screen.

There will also be an option for creating an account which will require the user to fill in details that will allow for the successful verification of the account. This screen will be extremely simple to use with bars to fill in for the information that we require to be filled out. This will end in with a simple create account button which will store the information and bring the user back to the log-in screen so they can go ahead and log in. There will be some built in functionality to detect errors in the user inputted data so that all the logged information is valid for the input fields.

For the trip selection screen, there will only be simple options that can be chosen so that the user will not have difficulty in understanding what each option is. The buttons will be displayed in a fashion where they will be very visible and a simple click will take the user to the destination they prefer. This screen will not have to deal with any error detection as the only options are to click one of the options listed and choose what to do. This will also promote efficiency and accessibility with quick options that can be chosen without having to search for them on the screen. In terms of the actual trip screen, it will be a little more complex in design as we will need to have multiple displays of data occurring on the screen at one time in set

locations on the interface. This will be a slight learning curve for a new user but will be relatively simple to where the user will be able to memorize and grow comfortable with the interface design after a few attempts of usage. There will also be a button that will need to be pressed in order to start the trip and end the trip that will allow for the user to simply begin and end their trip. The accessibility and efficiency of the interface for this system will be relatively simple in regards to the functionalities that the user shall be accessing as the interface will provide a rather easy to use screen that will allow for the quick action of what this system is designed to do.

The usability of the statistics page will also be rather simple as it will provide a quick report of certain stats that describe the user's fitness journey recorded in the app. The main practices that we looked to invoke in our project were keeping the interface as simple as possible and placing the users in control of the interface by giving easy to use tools and simple accessibility to the functionalities of the system.

4.2 Security

The main security risks are the user's login credentials, and the user's personal data that is stored as a part of their profile. This information is most at risk when it is being sent between the client and server. Any data passed between the client and server will be encrypted to a reasonable modern standard. In any case where fitness data is being used, the application will require permissions from the user.

5. External Interface Requirements

5.1 User Interface Requirements

Our user interface design will allow users to quickly and easily navigate our app with little confusion. Starting with the login screen, it will prompt users to enter your username and password with two textboxes or create a new account if they don't already have one. The account creation screen would include text boxes for first name, last name, username, password, and email address.

The home screen would show information about the current trip such as progress and total distance traveled so far. If no trip has been started it will prompt you to start one. On the home screen, a side menu will contain buttons to go to various other screens. The menu button will start minimized and can be expanded by tapping on it. Inside this menu will be account settings, a button to see trips in more detail, start a new trip, or view leaderboards. Account settings will have account information and the option to change passwords all laid out in an easy-to-read format.

Current trip details will show the progress on a current trip on a map along with other details. These details will be accessed by scrolling down from the map. To start a trip, a list of

possible trips will be shown as well as a search bar and a filter for popular trips. The leaderboards would show the current most popular trips and the fastest time to complete each one. A button would allow users to see leaderboards for their current trips or ones they've already completed. A search bar would be included here too. Overall, the layout would remain consistent and simple. All UI elements would use neutral colors and similar shapes to not be distracting or confusing. Every screen will be designed to be easy to find. Certain elements such as the trip view screen are accessible via multiple methods.

5.2 Software Interface Requirements

5.2.1 Google Maps SDK for Android: for providing a visual map for the user to view, plotting routes and distances between locations, identifying locations a given distance along a route, and accessing Google Street View to allow the user to look around their mapped location.

5.2.2 Android Health Connect: for retrieving data about the user's distance traveled while exercising. The user will be required to provide the application access to this data.