DiaLog
A Type 2 Diabetes Tracker
Version 2.0

Software Requirement Specification

Submitted to

Department of Computer Science and Engineering

California State University, San Bernardino, California

by

Bilal Muhammad Khan (CEO)

Project manager:

-Eduardo Reyes (007351208)

Assistant project manager:

-Braden Post (006944412) | Back End Engineer

Development team:

- Aaron Massey (007133328) | Front End Engineer

- Andres Ventura (007158353) | Back End Engineer

-Saul Cubillo (007199147) | Back End Engineer | Quality Assurance
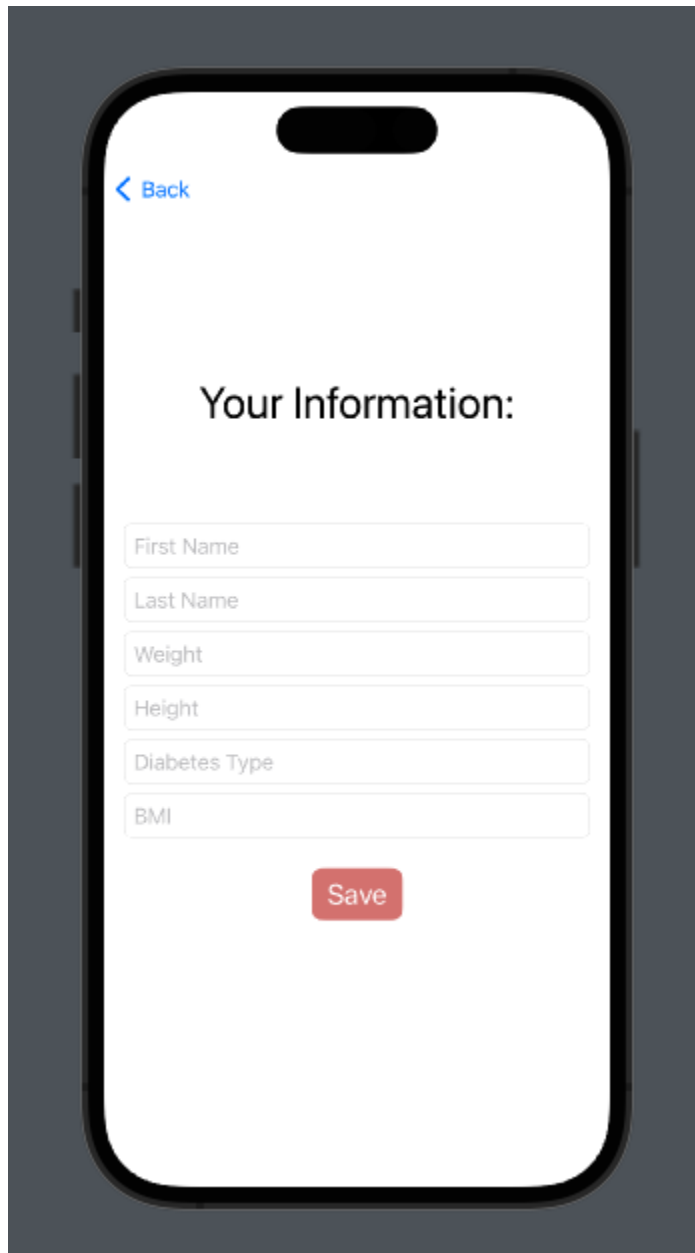
Table of Contents

1. Introduction

1.1 Purpose:

Despite the great and widespread conditions that affect so much of the populace, no mobile apps exist for those suffering with Diabetes. The troubling part of this issue is that many patients either do not have the motivation or resources to seek help unless it's simple and accessible. Such is our motivation, to create an easy to use mobile application targeting those affected by Type 1 and Type 2 Diabetes, but especially Type 2 Diabetes. Currently, there are little to no existing ease-of-use applications for diabetics to regularly and effectively track the metrics related to their health condition, such as glucose levels, and they are left having to estimate themselves how to return to normal levels. With our application, we wish to help these patients calculate their required metrics based on: current blood sugar level, weight, height, BMI, and Diabetes type.

We aim to calculate the needed insulin intake to return to proper levels, track the heart rate of the user to notify potential shocks, and provide suggestions for better health practices. Since diabetes targets many aspects of the body's immune and respiratory systems, we also plan to implement smart watch integration for tracking the user's heart rate during regular activities as another variable for calculation. This tool is

meant to be an all-around use case for its users; a pocketable tracker for the enormous amount of people struggling with Diabetes.

1.2 Intended Audience:

While the application is typically targeted towards those with Type 2 Diabetes, the application can be used by all Diabetes patients. As it is far more severe than its alternative, those with Type 1 Diabetes should be able to find and use the application to its full effect. While the Blood sugar and glucose calculations are primarily for diabetics, the calculations made by our application can be useful for a general user, even if its effects are far more understated for the average person. and Notifications are primarily geared towards providing warnings for preventing Glycemic shocks and Hypoglycemic shocks by measuring the potential of these occurrences through pulse tracking. A port towards the average user experience is possible, but due to the scope of this project, our primary focus is those struggling with Type 1 or 2 Diabetes.

1.3 Intended Use:

The user would use their Blood Glucose Meter (BGM) to take a blood sample, which will return a reading for their current blood glucose level (in mmol/L). They would then input that value into our application. Depending if they were a new user or not, they would also input their biometrics for the first time such as: BMI, blood type, Diabetes type, age, and gender, which would be saved as user data within the app. The app would then calculate how far they were from the normal healthy levels for their body, and would recommend how much intake it would require to get back to proper levels. The app could also connect to a smart watch with a heartbeat sensor for heart rate monitoring, detecting uneven or consistently low BPM in order to alert the user of possible incoming hypoglycemic shocks.

1.4 Definitions, Acronyms, and Abbreviations

BGM: Blood Glucose Meter- The small meter that Diabetics use to measure current blood sugar.

Hypoglycemia: The condition behind an individual's below average blood sugar levels, and a main condition of Diabetics.

Hypoglycemic Shock: A potentially dangerous affliction that occurs when hypoglycemia takes place, indicated by a sugar level of 70mg/bl or lower. This condition is the primary condition this app aims to prevent, as it can cause lightheadedness, immune system dampening, numbness, blurred vision, blindness, heart attacks, seizures, and strokes.

Insulin: a chemical hormone produced in the body that regulates the body's glucose intake and production. The lack of this pivotal hormone within the bloodstream is what causes the condition of Diabetes.

2. Overall Description

2.1 Product Perspectives:

2.1.1 System Interfaces (deployment diagram)

Deployment Diagram Type 2 Diabetes Tracker

<< device >>
Application Server / React Native and Firebase

<< device >>
:DB Server / Firebase

<< iTunes Protocol >>

Mobile Interface

<<component>>
User Database

: client mobile

<<component>>
Application

<<component>>
Database Interface

<<component>>
Diabetes Database

<<component>>
Authentication
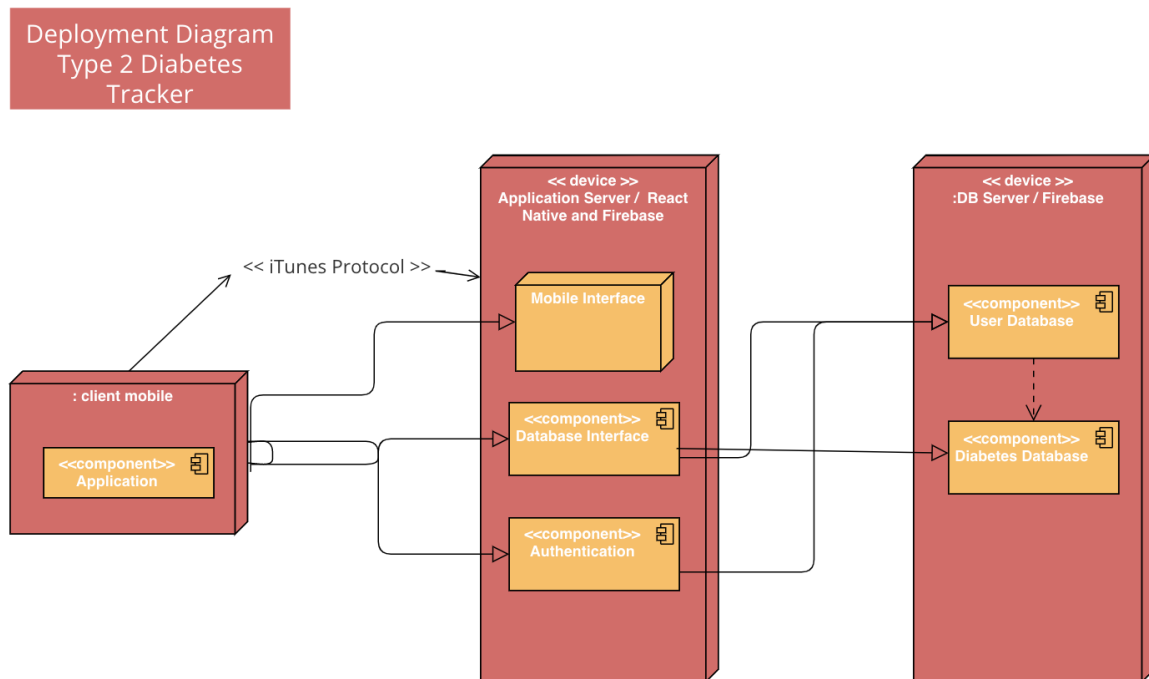
2.1.2 User Interfaces
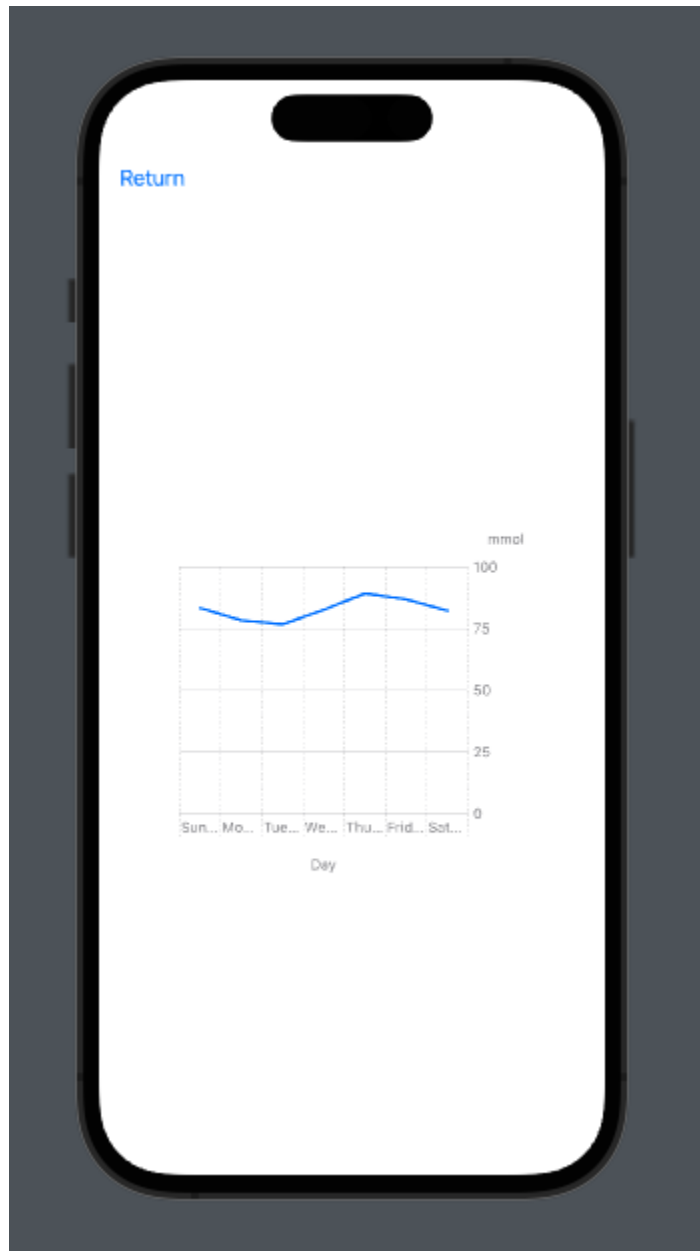
The app will primarily feature a log of a user's blood glucose levels over a selected period of time. This data will be visualized through a time graph and can be adjusted to user preferences. Under the graph will be pre-written notifications,m comments, insights, and alerts that will be shown to the user depending on what their current data or trends are. From within the app, users will be able to enter, edit, and track their biometric information that can be used for more accurate readings and measurements from the app.

2.1.3 Software Interfaces

Each dedicated user of the software is granted a profile and associated database that primarily contains blood glucose levels, but also all other relevant biometric data, as well as with the time in which each measurement was recorded. The data will be sorted in order of time so that it can be sorted graphically by the user interface. The main support will be graphics and charts but users will be able to access a list view of all their past data as well. The time-sorted data will be used for trend analytics and suggestions based on clinical recommendations and averages.

2.1.4 Communication Interfaces

In order for the application to function, we have a login and password management system in order to save users profiles and biometrics for ease of use. Therefore the application requires an internet or data connection in order to communicate with a server for password and user management. This is implemented through the Google Firebase database services.

The application communicates with a server to determine whether or not a user who is logging in exists yet. If yes, the app must void the create new user command. If not, it must redirect the user towards the new user creation window for their information.

Then it must have a redirect loop to send the user back to the login screen. If not, then

logically log them in automatically after account creation.

The program is implemented with some logic for communications, as well as possible implementation with apple watches, it must communicate back and forth for pulse rate data.

2.1.6 Site Adaptation requirements

Our initial implementation of this application is as an iOS application, the majority of which is developed in the Swift programming language. If we wish to port this application to other implementations, such as a web application (likely in HTML and CSS), further optimization may be required. Allowing for PC integration in this way would allow for more accessibility for users.
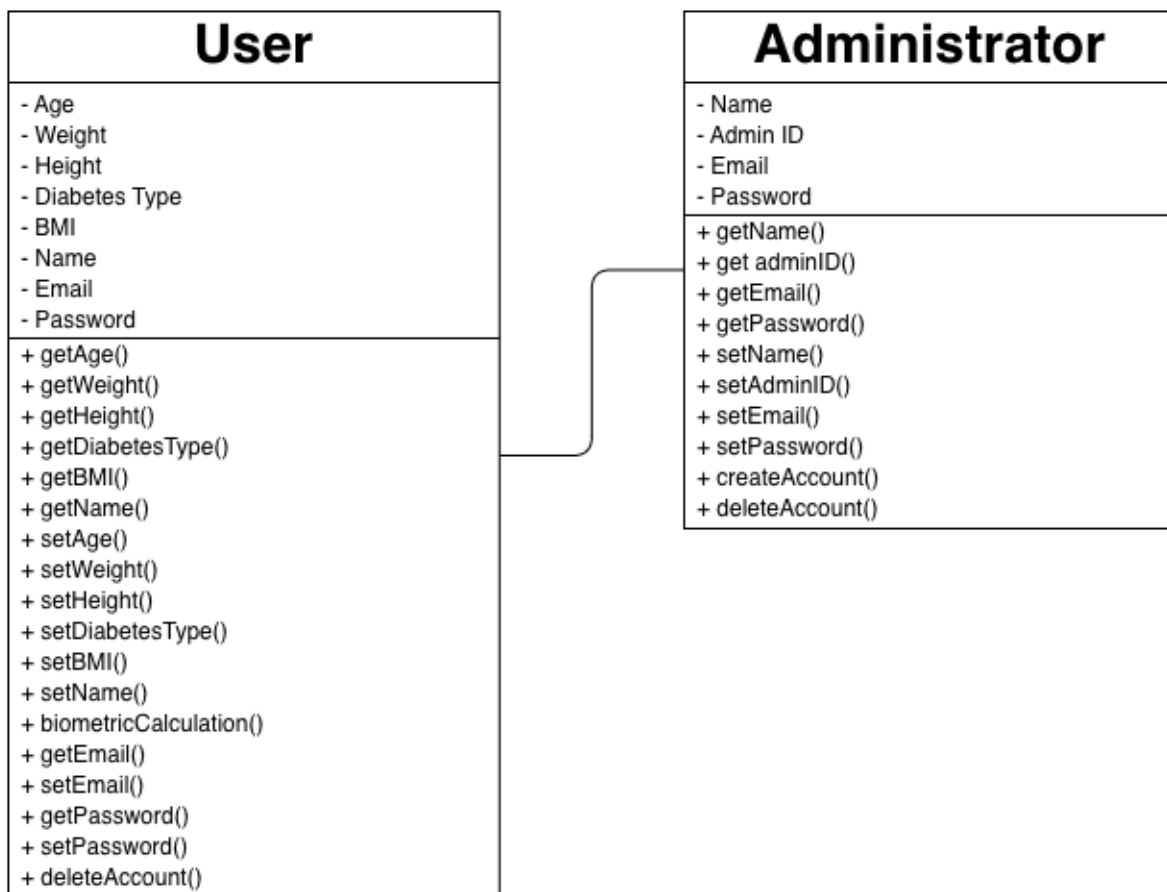
2.2 Use Case Diagram
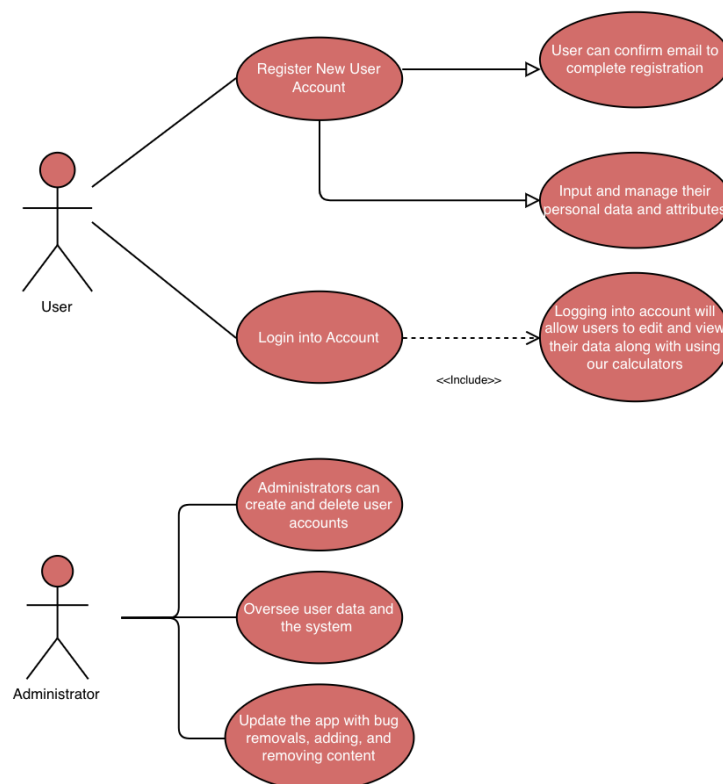
1. **User:** creates and manages their own account. They are also able to login, edit their data, and view their data over a period of time.

    a. **Register New User Account:** Users can register to create an account by signing up with their email and inputting their biometrics.

    b. **Login into Account:** Users can retrieve their information, update their biometric data, view past data, and view graphs of data.

2. **Administrator:** Administrator can create and manage user accounts, manage user data and system, and keep the software updated.

    a. **Create/Delete Accounts:** Admin can create and delete accounts on the request of the users.

    b. **Oversee User Data:** They look over the users data and the system as a whole.

    c. **Update App:** In charge of updating the software to add, remove, and resolve bugs.

2.3 Class Diagram

User
- Age
- Weight
- Height
- Diabetes Type
- BMI
- Name
- Email
- Password
+ getAge()
+ getWeight()
+ getHeight()
+ getDiabetesType()
+ getBMI()
+ getName()
+ setAge()
+ setWeight()
+ setHeight()
+ setDiabetesType()
+ setBMI()
+ setName()
+ biometricCalculation()
+ getEmail()
+ setEmail()
+ getPassword()
+ setPassword()
+ deleteAccount()

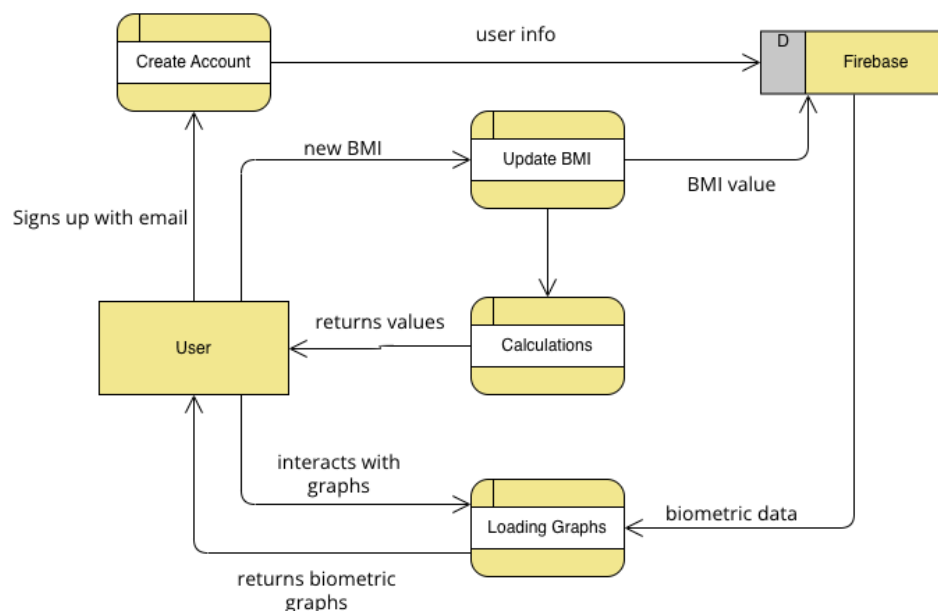| Administrator |
| --- |
| - Name |
| - Admin ID |
| - Email |
| - Password |
| + getName() |
| + get adminID() |
| + getEmail() |
| + getPassword() |
| + setName() |
| + setAdminID() |
| + setEmail() |
| + setPassword() |
| + createAccount() |
| + deleteAccount() |

- This class diagram shows our two objects and how they interact with each other. It also shows the attributes and the functions of each object.

2.4 Sequence Diagram
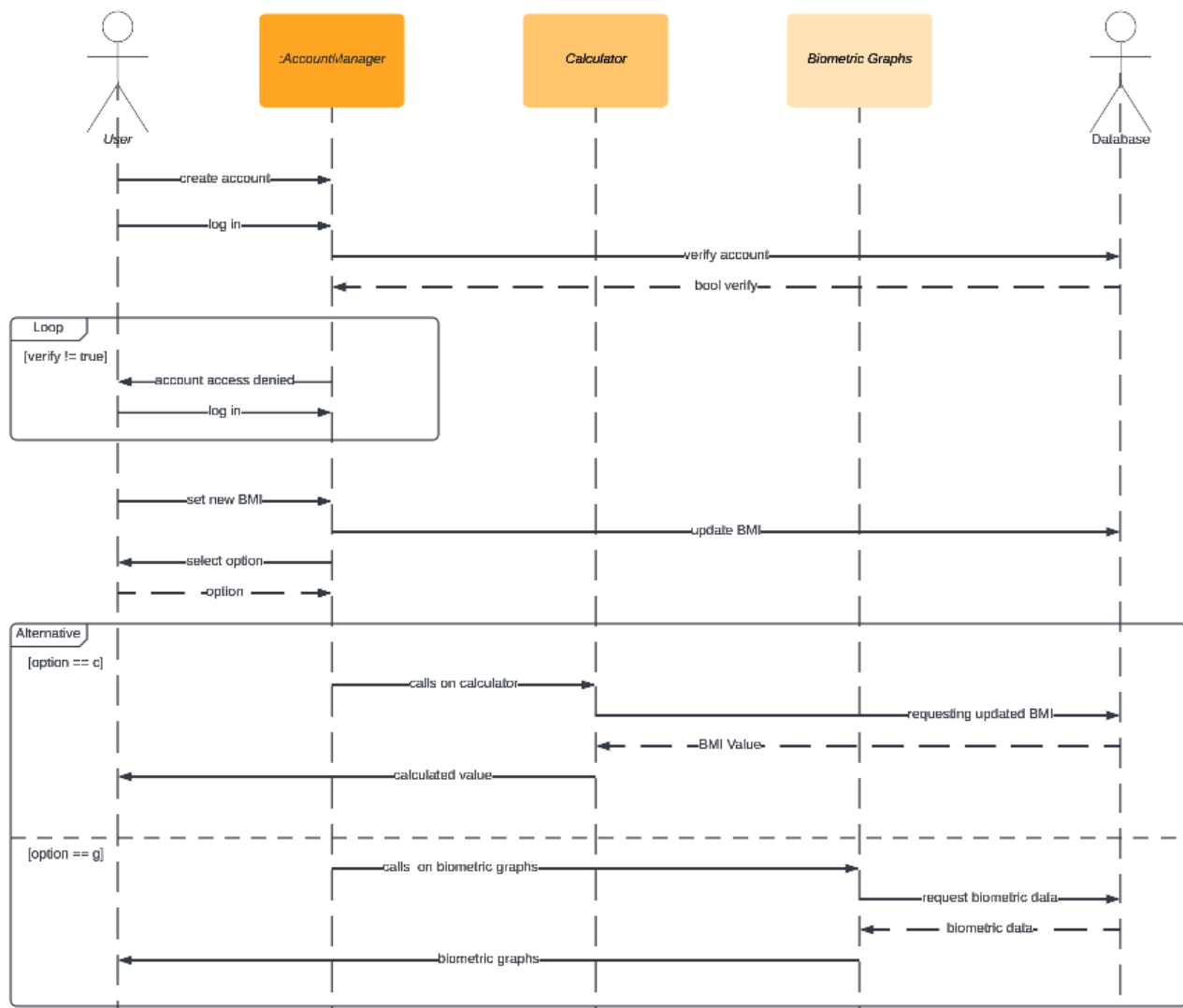
2.5 Data Flow Diagram

2.6 User characteristics
There are a few users in this application such as:
- Mobile user: These users who will access the system and input their information as they should already have an account registered in the application. The language will be set to english initially but should support multiple languages. Users should be able to have minimal knowledge of navigating through the application, because the UI would be simple and have lots of hints and ways to navigate the application.

- Online/Guest user: This user would not be able to access as much as the mobile user with their information already in the system, but they are allowed to use the free calculator for a fast way to check the amount of blood sugar they need.
- App Admin: These users troubleshoot and do maintenance needed on the app incase of hacking, crashes, glitches, bugs etc. Do regular checks and daily testing to see if the app works as intended
- Doctor/ Physician: This user has to put in their certification, to be verified as a doctor/physician, can have one to one messages/appointments with patients(mobile users), and have information of the patient

In the current iteration, the only user implemented is the general mobile user, but extended permissions for future user types are planned.

2.7 User Constraints

With these being still in the alpha stage of product development there will be a lot of constraints we would like to clear in the near future with more iterations of our product. One would be the application's current limitation as only being available as an iOS app on the app store. The app would also be able to only run a limited number of commands, which as of current are to use the diabetes calculator, and to also produce the graph.

2.8 Assumptions and dependencies

- Assumptions included for the application are assuming the user has the application updated, and has an IOS system like an iphone, ipad, Macbook, Apple watch.
- It is assumed that users would update the application to run at maximum efficiency and to be up to date with the features the app provides, and to avoid bugs.
- Information and user data will be protected and confidential, unless asked otherwise to be apart of a survey to either sell information, and to use user data for personal use to expand our product to have more data to work with and to serve our general users, to make it a better experience for our customers

3. Overall Description

3.1 Functional requirements

- Create and store user information based off of a login email, username, and password
- prompt the user to enter any of the supported information of the application, save and organize that information temporally, then immediately access and display that data on a time graph along with previous user inputs
- periodically notify and remind the user to enter their data, the frequency of which can be adjusted to user need and preference
- Compare user data to statistical averages and suggest practices for improvement

3.2 Non-functional requirements

- User login information will be stored and encrypted on firebase cloud services
- Login will be accessible by entering password and email or by entering password and username
- Graph will be displayed on home page of application and be dynamic with inputted data set
- Default notification frequency is daily; can be made to range from hourly to monthly
- Suggestion data will interact with data drawn from recent medical journals and literature on the topic