## Project A

# Fitness App

**Summer Term - July 12, 2015**


| Group Members |
| --- |
| Solomon Huynh |
| Jim Linnenburger |
| Sean Harrington |
| Larissa Hahn |
| Anthony Wilcox |
| Stephen Heng |

## Table of Contents

## Functional Requirements

The fitness app is a mobile application that promotes fitness, health, and nutrition to people globally. After installation of the app, users will be guided through the first time user process that involves entering their personal information such as their location, age, weight, and height. After that, they will enter the option to either lose or maintain weight. If weight loss is chosen, they will be required to enter their desired weight how how many points to lose weekly. The program will then calculate and recommend the number of calories to consume.

Users who finished the first time user process will be defaulted to the home page of the app when opening the program. The home page contains the daily calories left to consume for the day as well as navigations to the major sections of the app as listed below:

Major sections of the app:

1. Food and Fitness Log
    a. Any food and fitness logged will update the number of calories left to consume for the day.
    b. The user can log their meals in the food log. It will be separated by breakfast, lunch, dinner, and snacks in-between them.
        i. When searching the food, it will contain the calories and size of the meal by different measurements of popular restaurant foods, meals like chili with turkey meat, or individual ingredients such as tomatoes, three slices of turkey from ABC Brand, etc.
        ii. When the food is logged in, the calories will be calculated and it will update the calories consumed.
        iii. There will be a memorized system where it will keep a separate log of most commonly consumed meals and meals that are put on the user's favorites to quickly log the meals.
    c. Users can log their workout for the day. The user will enter what kind of workout and the number of calories they burned. This will update the daily calories to be burned.
        i. There will be a memorized system where it will keep a separate log of common (or user favorites) workouts done.
2. Nutrition Information of Foods and Supplements
    a. A database that contains user and professional reviews on the food/supplement.
3. Seek Professionals (Personal Trainer, Dietitian, etc)
    a. Professionals has the option to advertise their services in this section.
    b. Services are broken down by nearby professionals or online.
4. Connect with Others
    a. Forum-like section where there will be major categories such as workout recommendations, sharing recipes, finding workout partners nearby, and other workout/health related discussions.

5. Reports
    a. Contains graphs and information on the weight lost over a period of time (months, years, date range, etc). Can also contain ratios of carbs/protein and other interesting health facts about the user.

## 3 Most Important Use Cases

1. Use case name: **Logging Calories**
   Actor: Losing Weight User
   Preconditions: User has eaten lunch and a snack and would like to log both of those meals at once.
   Postconditions: User has successfully logged his lunch and snack intake calories.
   Flow of Events:
      1. User taps on the Fitness App icon and the home page loads promptly.
      2. He makes the selection from the menu by tapping “Food Log”
      3. Once the screen loads, he selects the “+” icon to log calories.
      4. He selects the food type.
      5. After typing only a few characters, the search suggests the items he ate as well as lists the calories. He is able to tap on the items to add them to his calorie log effortlessly as long as he remembers to enter the correct quantity.
      6. Immediately after hitting the add button, he sees his current calorie intake as well as calories burned and the statistics showing his calorie progress for the day.

2. Use case name: **Logging Exercise**
   Actor: Gaining Weight User
   Preconditions: User performed a strength training workout to build muscle and enhance overall tone.
   Postconditions: User successfully logged his workout.
   Flow of Events:
      1. User taps on the Fitness App icon and the home screen loads promptly.
      2. He makes the selection from the menu by tapping “Workout Log”
      3. Once the screen loads, he selects the “+” icon to add exercise to his daily progress.
      4. After typing only a few characters into the search box, the search suggests the specific exercises he did (ie. planks) and then allows him to select how many reps for each exercise.
      5. Immediately after hitting the add button, he sees his current calorie intake as well as calories burned and the statistics showing his calorie progress for the day.

3. Use case name: **Checking Stats**
   Actor: Losing Weight User
   Preconditions: User has successfully logged both exercise and calorie intake throughout the day and would now like to view his progress at the end of the day.
   Postconditions: User successfully monitors progress using the Fitness App.

Flow of Events:

1. User taps on the Fitness App icon and the home screen loads promptly.
2. He makes the selection from the menu by tapping “Statistics”
3. Screen loads giving the user option to select daily report, weekly report, monthly report, or yearly report
3. Within 5 seconds screen will load showing calorie intake for the day for each meal as well as the day as a whole along with how far above or below calorie target user was. It will also show a breakdown of macronutrients (protein, carbs, and fats) consumed during the day and their relation to their target. If a user missed either their calorie or macronutrient goal for the day by a certain margin (around 10%) then the app will suggest what could be changed. (If you had a bagel for breakfast and came up short on protein the app will suggest to maybe eat a breakfast with more protein.)

## Non-functional requirements with Fit Criteria

For the first time users:

- A new user will be able to set up their personal information, measurements, notification preferences, and goal (lose or maintain weight) in less than 5 minutes.
- The setup page will all be on one screen so the user will not have to navigate between pages.
- The application will be giving the first time user feedback and suggestions within the 5 minutes of first entering in data.

For the Home page:

- Users will be able to see daily calories left to consume immediately upon opening up the application.
- User will be able to navigate to and be able to enter in daily data with 2 or less button presses from the home page.

Food Log:

- Calories left to consume for the day will update within 5 seconds of user submitting a food log into the app.
- Food database will be able to find the food the user wants to enter a minimum of 95% of the time.
- Manually adding food to the database will be able to be done in less than 1 minute.
- For food items in the database, the user will be able to log an entire meal in less than 1 minute.
- Submitting a food log will all be done on a single page of the app.

Fitness Log:

- Calories left to consume for the day will update within 5 seconds of the user submitting a fitness log into the app.
- Users will be able to submit a fitness activity log in under 1 minute.
- The calculated calories burned for a fitness activity should be reported accurate to within 5% of actual.
- Submitting a fitness activity log will all be done on a single page of the app.

Seek Professionals:

- Upon navigating to page to find a professional (personal trainer, dietitian, etc) a user will be able to find a list of providers in their immediate area or online within 30 seconds of beginning their search.
- Professionals who advertise will have their services show up at least 95% of the time a user searches for that type of service within 25 miles.

Reports:

- Reports will be generated within 5 seconds of a user selecting a report to show.
- 80% of reports will be able to fit onto a single screen.

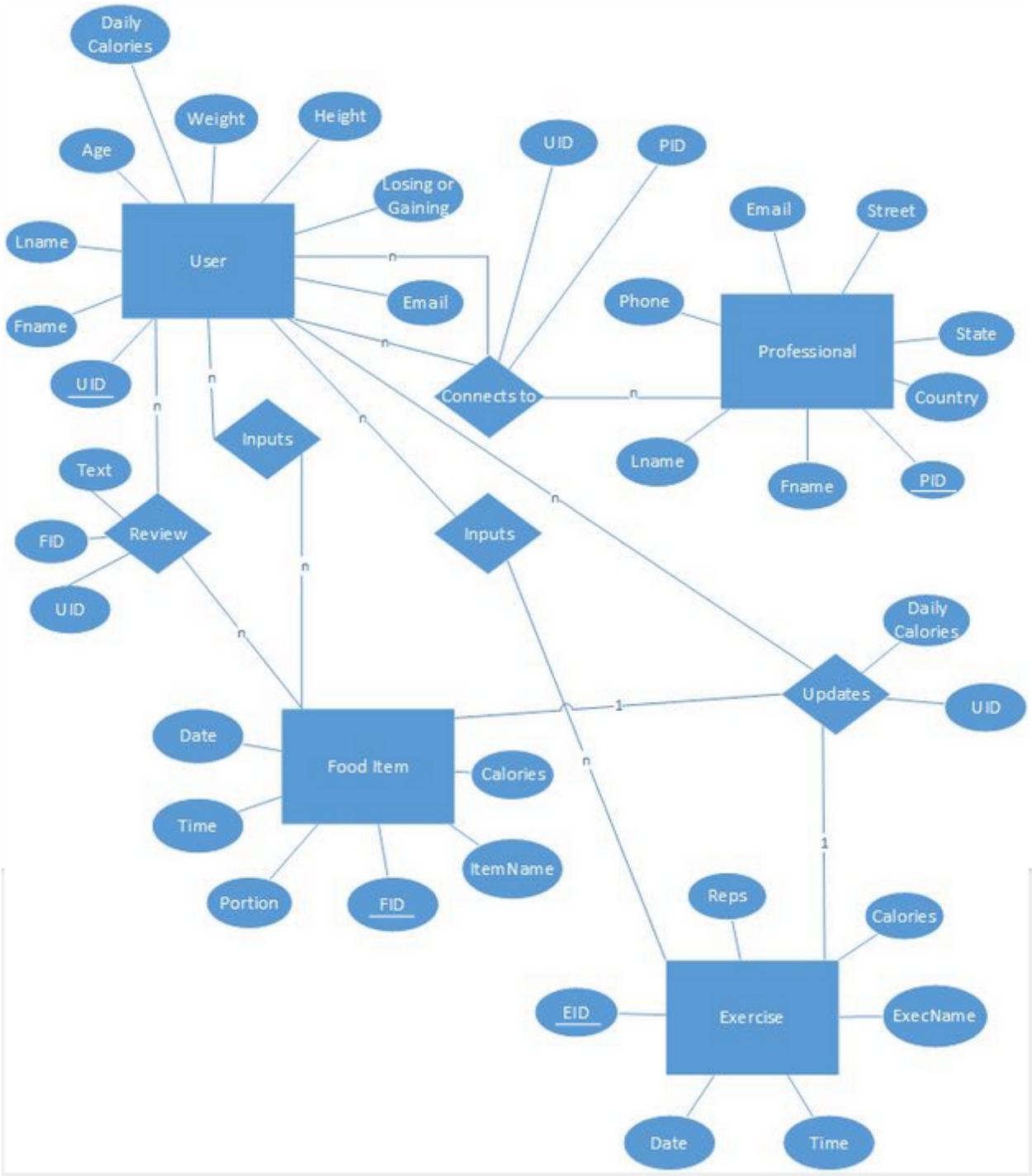
Advertisements:

- Users will be exposed to paid advertisements every time they open the app.
- Advertisements will take up a maximum of 20% of the viewable window or will be visible for less than 10 seconds.

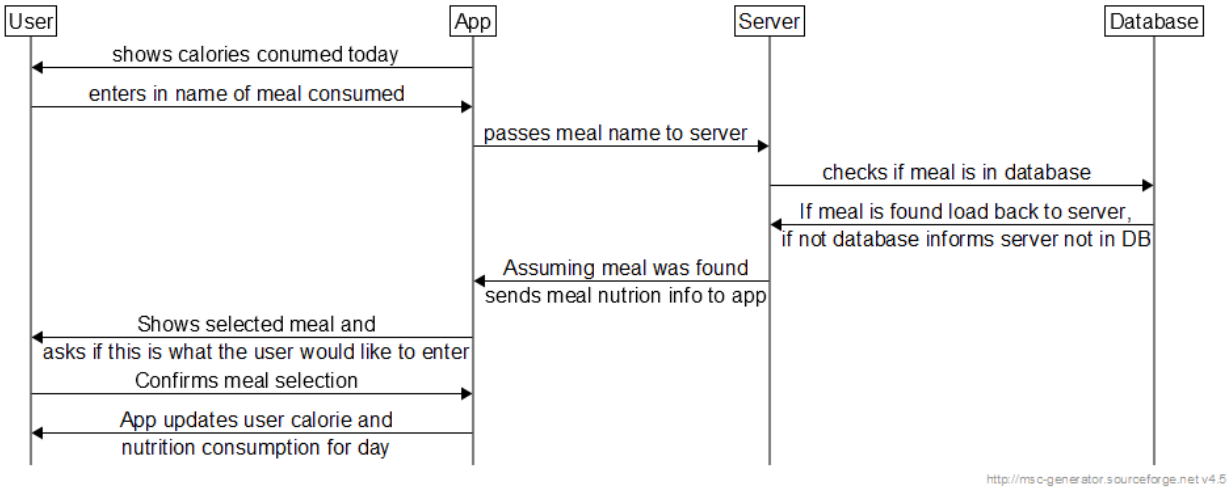
Other requirements:

- This app should be usable on 95% of all ‘smart’ mobile devices.
- Food and professional service provider databases will be updated weekly.

## ER Diagram

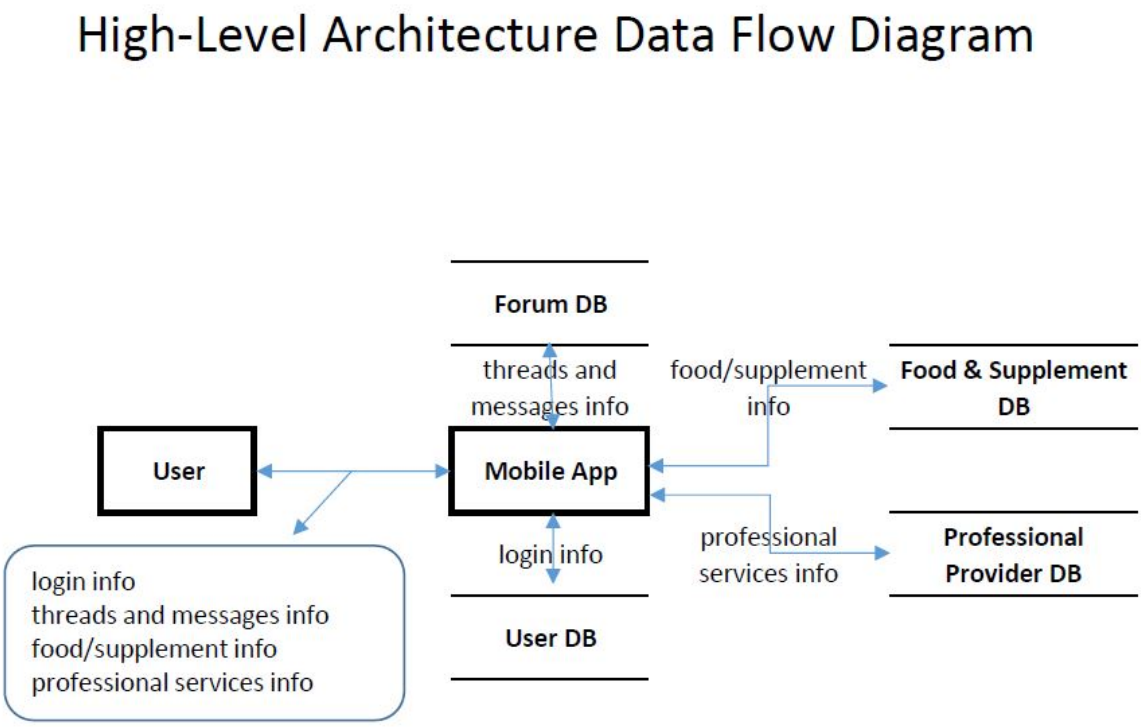

Daily Calories
Weight
Height
Age
Losing or Gaining
UID
PID
Email
Street
Lname
User
Email
Phone
Professional
State
Fname
UID
Connects to
Country
Inputs
Lname
Fname
PID
Text
Review
FID
UID
Inputs
Daily Calories
Updates
UID
Date
Food Item
Calories
Time
ItemName
Portion
FID
Reps
Calories
EID
Exercise
ExecName
Date
Time

## MSC Chart

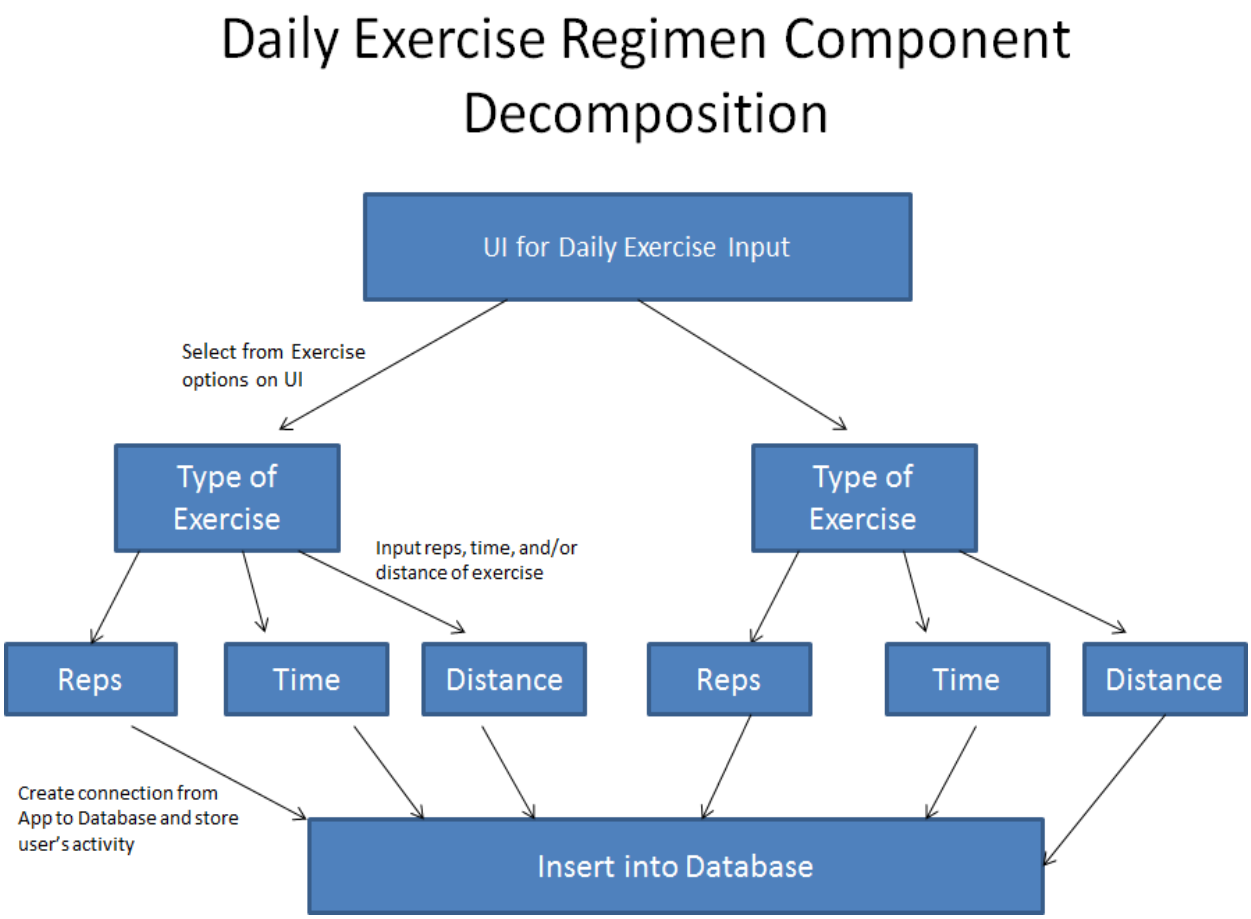

User
App
Server
Database
shows calories conumed today
enters in name of meal consumed
passes meal name to server
checks if meal is in database
If meal is found load back to server,
if not database informs server not in DB
Assuming meal was found
sends meal nutrion info to app
Shows selected meal and
asks if this is what the user would like to enter
Confirms meal selection
App updates user calorie and
nutrition consumption for day
http://msc-generator.sourceforge.net v4.5

## Paper Prototype

## High-Level Architecture

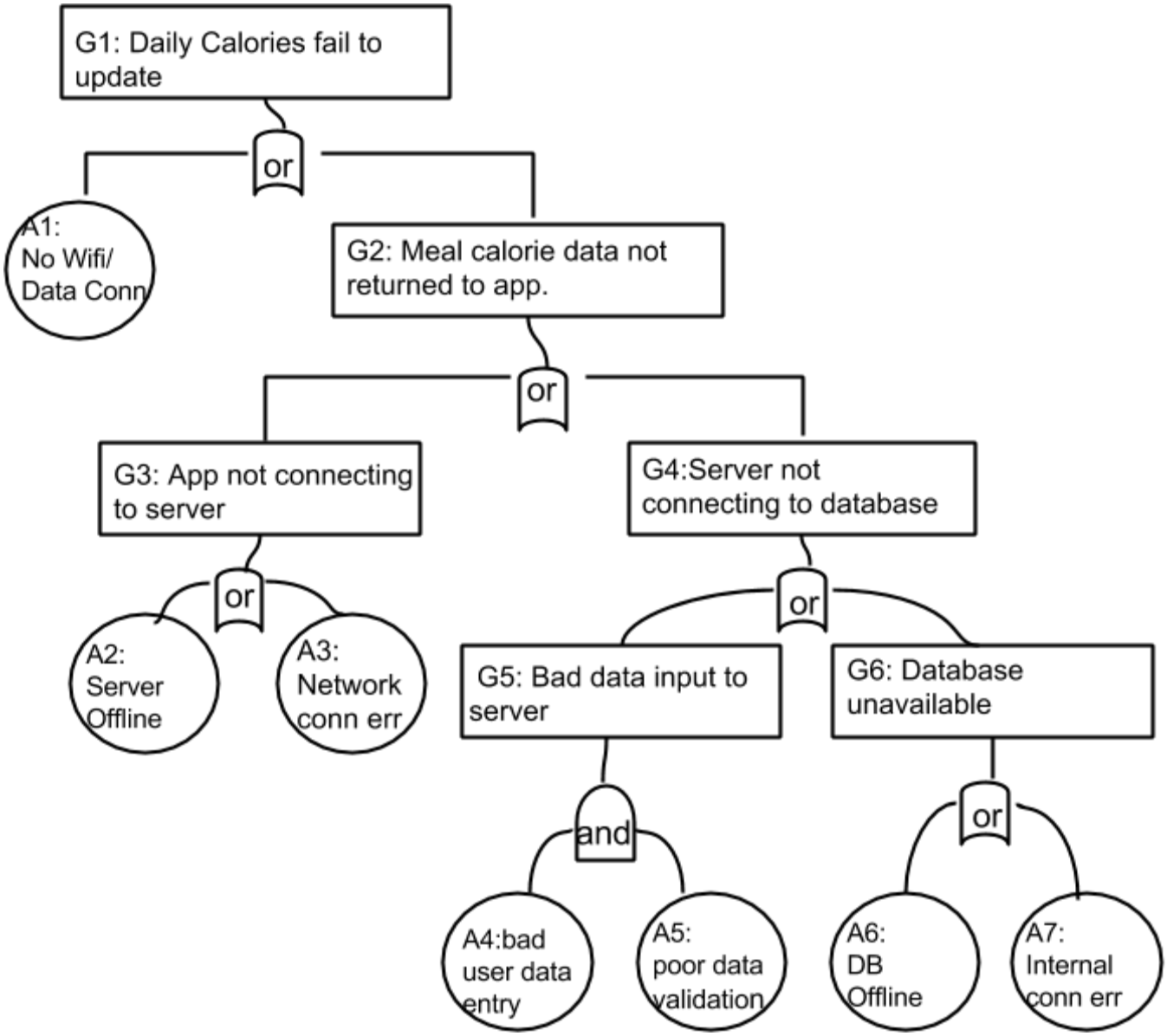


## Architecture Decomposition



## Failure Mode: Daily calories fail to update upon meal entry

Fault Tree



## Requirements Discussion

The architecture chosen for the Fitness App design meets the functional and nonfunctional requirements the design team chose after conducting the requirements evaluation process. The primary stakeholder was presented with the design team's understanding of the requirements. Clarification was given, and the design team further collaborated verifying understanding of the initial vision for the app. The primary stakeholder's initial vision contained multiple use cases making design team discussions of how a person would use the app in a concrete scenario efficient. The team was able to move forward with unambiguous and testable requirements. Clarification from the primary stakeholder and internal discussion indicate that the requirements chosen are also correct.

The overall architecture of the Fitness App is primarily a repository where the app serves as the client and interacts with multiple data storage services and devices dependent upon the user's interaction with the app. Multiple data storage repositories exist in the higher level architecture to facilitate user to user interaction through a database rather than a peer to peer architecture. This holds for user connection with professionals as well.

In comparing the prototype to quality attributes, it's apparent that all of the quality attributes desired by the stakeholder and the team are present. Controls are intuitive without ambiguity or ability to lose the functional flow of the main requirements. With the press of one button, users may log meals, exercise, generate reports, or connect to another user/professional. Users may view their daily progress without pushing any buttons.

Analysis of the failure mode of what we determined to be the most important use case, that is a user logging meals and NOT getting an update to their daily calories, shows an overall positive reflection of the architecture. The team did, however, identify some possible design alternatives and or programming design improvements moving forward. This analysis served the team well as logging exercise and meals have the same failure modes so we were able to view two modes with one analysis. In short, inability to connect to the respective database for the queried data will create a failure. There are multiple ways these can occur, but most of them are outside of the design team's control. Once possible design alternative would be to auto redirect users to manual food or exercise entry with a message telling the user about the database connection failure should one occur. Similarly, the design team may look at system (that is user device system) cost to maintain locally a list of foods and exercises they enter often, even if they have not added them to a favorite list. This would allow the user to update their workout and exercises offline and allow the database queries and updates to sync when whichever component failed comes back online. It would further decrease queries to the database if the app looks locally for the food and exercise information first, decreasing opportunity for failure between the app and the database overall. Finally, use case walk through indicates the same results as other forms of analysis. The architecture itself meets the requirements. The use case walk through reveals the same opportunities for improvement that failure mode did. The design team recognizes these

opportunities as back end coding decisions and feels that they can be achieved within the existing architecture.