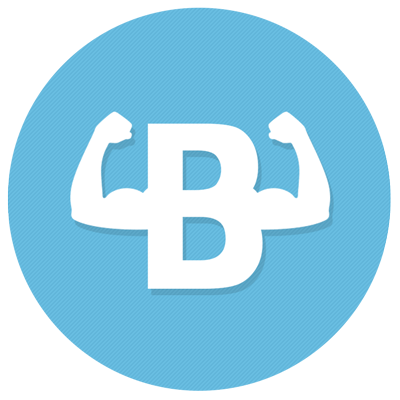
**CS4704 Software Engineering Capstone**

Spring 2016



**BetterU**

“Be you, but better”

|  |  |  |
| --- | --- | --- |
| Julia M. Binger  Mykhaylo D. Bulgakov  Allan F. Chua  Duke J. Forsyth  Filip D. Gouglev | Amanda J. Kahn  Mukund A. Katti  Travis M. Lu  Corey S. McQuay  Ojas D. Mhetar | Ryan T. Munz  Benjamin F. Robohn  Jared J. Schwalbe  Timothy T. Street  Hung T. Vu |

Department of Computer Science

Virginia Tech

Blacksburg, VA 24061

Date: May 1, 2016

Team Number: 1

Submitted to: Prof. Osman Balci

EXECUTIVE SUMMARY

TBD

<< Provide a structured summary of the content of the document, by emphasizing your important contributions to catch the attention of a very busy executive

Provide a numbered list of cloud software features, mobile software features, and other technology features you implemented in your cloud/mobile software-based solution.

>>

**TABLE OF CONTENTS**

EXECUTIVE SUMMARY ii

1. SOFTWARE LIFE CYCLE 1

2. PROBLEM SPECIFICATION 2

3. REQUIREMENTS SPECIFICATION 2

3.1 Enter/Change Goals 2

3.1.1 2

3.1.2 2

3.1.3 2

3.1.4 2

3.2 Set reminders for workouts, food, goals 2

3.2.1 3

3.2.2 3

3.3 Be reminded of goals (via push notifications) 3

3.3.1 3

3.3.2 3

3.4 Get points for meeting goals 3

3.5 View Recommended Foods 3

3.6 Select Recommended Foods 3

3.7 Track calories, grams of fat, sodium etc. daily 3

3.8 Enter a specific dish from a restaurant 4

3.9 Enter Food Consumed To Log Calories 4

3.10 Enter food item and receive nutritional info 4

3.11 User Account Creation 4

3.12 User Account Deactivation 4

3.13 Edit Account Settings and Preferences 4

3.14 Change Password 5

3.15 Storing the User’s progression throughout a period of time 5

3.16 Construct Graphs from User Account Information 5

3.17 Log in/out 5

3.18 Review Graphs 5

3.19 Share accomplishments on social media 5

3.20 Add/Recommend Friends 6

3.21 Remove Friends 6

3.22 Challenge friends across social media 6

3.23 Deleting Social Media 6

3.24 Enter calories burned 6

3.25 Enter workout intensity and duration 6

3.26 View example exercises 6

3.27 Get recommended exercises 7

3.28 Track Running Statistics 7

3.29 Non-Functional Requirements 7

4. ARCHITECTURE SPECIFICATION 7

5. DESIGN SPECIFICATION 7

6. DELIVERED SOFTWARE FUNCTIONALITY 8

7. HOW TO BUILD AND RUN THE DELIVERED SOFTWARE 8

8. CONCLUSIONS 8

REFERENCES 8

TEAM MEMBER CONTRIBUTIONS 10

PROJECT GRADING 11

# SOFTWARE LIFE CYCLE

A **good software engineer** develops software by following the software life cycle shown below.



A **programmer (hacker or ad-hoc developer)** develops software by looking at the problem and directly coding in an IDE. This approach is known as the **Build-and-Fix Approach**, which must never be used!

# PROBLEM SPECIFICATION

In today's society, most people usually target their New Year's Resolution of making themselves a better person in the physical aspect. Meaning that they want to lose unwanted weight and fat. The issue is though is that they are only motivated for so little time, roughly 2 weeks to a month, in the New Year and give up on their resolution because they do not see immediate results. Other people also just use "fad" diets that most are taking a pill or supplement and then claiming results will come immediately. In most cases this is just another failure.

People who truly want to change to a positive healthy lifestyle must understand that it all comes down to the number of calories taken in (through eating) and the number of calories used the physical exercise and activity. So simply having a committed diet and exercise program. The problem with that there is no real self-usable software program that collects the information of calorie intake/outtake and record data from the user of meals consumed as well as exercise performed.

There is also not a true program that can send food recommendations if the software knows what kind of diet the user is on and knows what the user has had for that day (for example if the user already consumed 1500 calories for the day and is on a 1800 calorie diet, the program can take what it knows about the user preferred foods and recommend a 300 calorie meal or two 150 calorie small meals. The program would also let what the user needs to do for both calorie input/output throughout the day to reach the user's goal.

The stakeholders and decision makers who would be interested in the solution of the problem would be people who want to change and better themselves physically through physical exercise and proper dieting. The problem of the person not having full, organized, and easy management of their physical activity, and lack knowledge of being patiently committed to see results can easily be solved through a cloud/mobile application that would store the users results of calorie input and output and information like weight, height, etc. A database of various foods, exercises and the rate of calories burned by the hour would be required to.

# REQUIREMENTS SPECIFICATION

The next step after creating the problem specification is to do the requirements engineering for BetterU. The team completed requirements engineering by examining other nutritional and fitness applications such as My Fitness Pal and asked some other students what they would like to see in a nutritional and fitness application. Once the team gathered all the raw information, the information was then broken down and user stories were created. From the user stories, the team was able to create well-defined requirements that BetterU would have to meet. The user stories and their corresponding requirements are listed below. Some requirements are repeated across different user stories.

## Enter/Change Goals

The user can input fitness goals to personalize their experience within the app. This will allow the user to stay focused during their fitness journey and motivate them to continue working out in order to achieve a better them.

The user will be able to change the goals they set. The reason for doing this could be a user surpassing their original goal, or a user misjudging how much time they will have to work out during the average week. Changing the goals will help users stay on track with realistic improvements.

### 

The app shall allow the user to enter a weight goal of either losing weight, maintaining weight, or gaining weight.

### 

The app shall allow the user to enter distance/weight lifting goals with the options of distances in metric and imperial, as well as amount of weight, in pounds or kg.

### 

The app shall allow the user to enter long term and short term goals with goals per month, year, and end goal.

### 

The app shall allow the user to enter/change goals under “Goals” in the user account information tab/page.

## Set reminders for workouts, food, goals

The user should be able to set personal goals since each person's goals will vary. Some users will be want to lose weight while other will want to gain muscle effectively gaining weight. Later users will be able to get points for each goal that they make that scale in amount based on the type of goal.

### 

This application shall allow users to set reminders for goals previously set.

### 

This application shall allow users to get a set amount of points per goal achieved.

## Be reminded of goals (via push notifications)

Since the user will be able to set the goals for their account they must be able to receive notifications for the goals. They could also set an alert should a long achieving goal hits a certain threshold you would be reminded. An example would be if a user had a achievement of losing 50 pounds and they lost 25 they should get a reminder.

### 

This application shall remind the user of his/her goals at set intervals.

### 

This application shall allow the user to set/modify notification intervals

## Get points for meeting goals

Users will receive points for completing goals. The larger the goal, the more points rewarded. Then they can compete with their friends to see who can score more points each week/month. We’re confident that this gamification will increase motivation for maintaining and completing goals.

This application shall reward varying point values for completing goals.

This application shall reward points for completing daily, weekly, and monthly challenges.

This application shall share scores with friends to promote competition.

## View Recommended Foods

As a user is about to decide what they should eat today, BetterU will be able to recommend a food based on the user’s goals and current nutritional and caloric intake of the day.

Recommended foods will also consider the current meal of the day, and if requested, can offer popular foods for that meal.

The app shall recommend what to eat based on goals and current nutritional/caloric intake.

## Select Recommended Foods

Once the user selects a food recommended to them, the app will take the food entry in our database and retrieve all nutritional information associated with it. The information will then be counted into the user’s daily intake.

The app shall be able to access our food database and add the nutritional data from the database into the user’s daily database.

## Track calories, grams of fat, sodium etc. daily

After the user inputs the data of the food they have eaten, the app will be able to aggregate key metrics such as macronutrients (protein, carbohydrates, and fats), as well as sodium, vitamins, etc. These will be broken down on a daily chart so the user has a summary of the nutrition they eat each day.

The app shall allow users to view charts/graphs regarding their progress

The app shall collect user information regarding their current nutrition and goals

## Enter a specific dish from a restaurant

A user can browse the restaurant’s menu in the app and look at the nutritional information of each dish. This decreases much of the hassle presented by entering nutritional information, and helps the user make healthy choices.

The app shall allow users to log calories by specific food item

The app shall allow users to log menu items from specific restaurants

## Enter Food Consumed To Log Calories

There is a user that wants to enter food items into the application to log calories consumed. The server-side application will search the USDA’s National Nutrient Database based on user’s food search. Once the food item is found and selected, the app will update the number of calories consumed on that day for that user. The new consumption data will also be appended to the Progression Database.

The app shall allow users to log calories consumed or burned

The app shall allow users to search for and select food items from the USDA’s national nutrient database

The app shall provide a tab specifically to enter in food information

## Enter food item and receive nutritional info

The user should be able to search for food items on our application and then receive nutritional facts such calories, ingredients, sugars, proteins, etc. If this information is quick and easy to access then the user will be more inclined to research the facts about what they eat before they put it in their body.

This application shall allow the user to search for nutritional info based on name of food.

This application shall utilize a free API for the nutritional info.

## User Account Creation

There is a Virginia Tech student who wants to lose weight, but he is having a hard time keeping track of his progress. The student then decides to download BetterU from the Apple Store, hoping that he can finally organize his workout routines. He opens the application and decides to create an account. BetterU then prompts the student to enter his username, password, height, weight, age, and weight goals.

The application shall allow the user to create an account with a username, password, height, weight, weight goals, gender, and age.

The application shall allow the user to maintain their account with the option to change their personalized data.

The application shall allow the user to have their account recorded into a cloud database, so that they are able to take their account with them on any mobile devices.

The application shall allow the user to create an account if they have a social media account, such as Facebook.

## User Account Deactivation

There is a Virginia Tech student who has lost 100 pounds and became a better version of himself. He is the embodiment of “Be U, but Better” so he has decided to delete his account. All he has to do is select ‘Delete Account’ in the menu and it will be deactivated.

The app shall change a user’s account to ‘inactive’ in the database

The app shall leave the user information in the database, in case they want to return later

The app shall have a ‘deactivate’ button in the settings menu

The app shall send an email regarding account deactivation

## Edit Account Settings and Preferences

Every user is unique and will want the app the function differently for him/her. So the user will be able to change how frequent the push notifications appear and what he/she is notified about (Updates about friends, Goal Reminders, Enter Information Reminders), what the app emails the user about, reset user data, change profile information (Height, Weight Age, Sex, and Weight Goals), Set regiment type (Lose Weight, Maintain Weight, Build Muscle), and Connect/Disconnect from Facebook.

App shall allow user to change the frequency of push notifications from Monthly, Weekly, Daily, or every 4 hours

App shall allow user to change the frequency of email notifications from Monthly, Weekly, Daily, or every 4 hours

App shall allow user to change whether he/she is notified about goal reminders via email and/or push notifications

App shall allow user to change whether he/she is notified about friends’ achievements via email and/or push notifications

App shall allow user to change whether he/she is notified about enter daily log information via email and/or push notifications

App shall allow user to reset nutrition and workout data

App shall allow user to change their height

App shall allow user to change their weight

App shall allow user to change their age

App shall allow user to change their sex

App shall allow users to create/edit weight goals

App shall allow user to select between regiments of lose weight, maintain weight, build muscle, or improve cardio

## Change Password

There is a user that has decided to change his/her password either because they want a new password or they have forgot their password and need to reset the password. The user will have to enter their current password once and the type in the new password twice to confirm it was typed in correctly. If user forgot their password, then they will hit the reset password button and an email will be sent with a temporary password to the email in the user’s account information. The user will then proceed as before, but use the temporary password instead.

User shall give us a valid email address during account creation

User shall enter account name/email address when requesting temporary password

User shall enter current/temporary password once

User shall enter new password twice

Password shall contain at least 2 numbers, 2 lower case characters, 2 upper case characters

Password shall allow users to use special characters in their password

Password shall be between 8 to 25 characters in length

App shall have a reset password button/link to get temporary password

App shall automatically create a temporary password and send the password to the user via email when requested by user

## Storing the User’s progression throughout a period of time

The server side of the application will have a progression database which will store the User’s progression of calories consumed when logging in. This is needed so that that application can show charts and graphs of the user’s progression by pulling the data off of this database, analyze this data to see if the user is meeting their goals, and so on.

The server side of the application shall store the user’s information to track their caloric intake, weight loss, physical activity, and completed exercises.

The user shall store the information into the application that will be stored into this said database.

## Construct Graphs from User Account Information

The server-side application will retrieve information regarding calories consumed, minutes active, etc. from user records in the Progression Database. From this data, the application will generate graphs giving users a visual representation of their progress. Data will be displayed via line/bar graphs, where previous dates are along the x-axis.

The Progression Database shall provide information about specific users by day to the server-side application

The app shall record information about user activities by recording it in the Progression Database

The app shall provide a separate tab for viewing graphs/user-related data

The app shall provide functionality to allow the user to scale the viewport of the graphs (i.e. view over different periods of time)

The app shall provide graphs of varying types to the user, such as minutes active, caloric intake, etc.

The app shall allow users to view charts/graphs regarding their progress

The app shall construct graphs from user account data showing user trends over time

## Log in/out

The user logs into their account by supplying a username and password that is associated with their account preferences and data stored in our users database. The user will be prompted to log in when they first open the application. They can either log out manually at any time, but will remain logged in if they close the app without logging out. If the user wishes, they may toggle their settings to log out automatically every time the application is fully closed.

Each user shall have a username and password that serves as a key into the user database, which holds all of the data for that particular user.

The user shall be able to enter their username and password that they can enter into the appropriate fields on the login screen to reach the home screen of the application and have access to all of their user data and preferences.

The user shall be able to log out during their session to close all views of the app and disallow access to any screens besides the login screen. After “Log Out” is clicked, no more user information is accessible from the device until a new login is processed.

The user shall be able to access the “Log Out” button within 2 clicks from anywhere in the app.

The user shall be prompted to log in to the app from the login screen if and only if it is their first time opening the app or the last thing the user did was log out.

The user shall be able to go to the settings screen and select “Log Out On Close,” which shall make the app simulate the action of a manual log out whenever the user fully closes the application, in order to preserve their privacy if desired.

## Review Graphs

The user logs their calories in/out and active minutes over time and is able to go to the data section of the app to look at how these have changed over time. The graphs are generated based on the user data stored in their associated account. The user is able to easily interpret and see trends in the data from the home screen of the application. To see how trends in the data, they can adjust the scope.

The app shall allow the user to be able to view how the data they log has changed over time according to net calories and active minutes.

The app shall allow the user to be able to adjust the scope of the graph from day, week, month, to all time to see their progress.

The graphs of the data shall be viewable from the home screen of the app.

The default view of the data on the home screen shall be of the calories in/out and active minutes of the current day as two separate graphs.

## Enter calories burned

Users will be able to keep track of the calories burned during their workouts. This will allow them to chart their progress. The information will be used with the user’s nutritional intake so that they can manage their fitness goals.

The app shall allow users to log calories consumed or burned

The app shall allow users to view charts/graphs regarding their progress

## Enter workout intensity and duration

Users will be able to accurately measure how many calories are burned based on the exercise and the time interval. The more intense the exercise, or the longer you exercise, the more calories are burned.

The app shall allow users to log calories consumed or burned

The app shall allow users to view charts/graphs regarding their progress

## View example exercises

The user will be able to browse through a catalog of exercises to gauge the difficulty and purpose of the exercise. The user will also be able to determine how the movement is performed.

The app shall display a catalog of exercises that shows how movements are performed, difficulty of each movements, and the muscle/muscle groups targeted.

The app shall allow the user to search through a catalog of categories based on difficulty and/or muscle group

## Get recommended exercises

The user will be able to see recommended exercises based on the user’s specific fitness goals. The exercises will vary in difficulty and provide the user with options that they feel most comfortable with.

The app shall recommend workouts based on the user’s goals

The app shall allow users to enter in injuries so that no dangerous workout recommendations are made

The app shall allow users to enroll in workout plans and get reminders about them

## Track Running Statistics

Users who run or jog as an exercise will be able to track the time and distance they spend for each session as well as the route they have taken shown on a map.

The app shall allow users to track their run time and distance.

The app shall be able to plot the route on a map.

## Non-Functional Requirements

The application shall be Royalty-Free and be developed through open-source technology

The application shall remember mobile devices and allow users to automatically login if they have cookies enabled

The application shall make database queries no longer than 2 seconds

The application shall hide user’s personal and account info from developers and other users

The application shall prevent users from accessing other user's profiles

The application shall make recommendations based on user’s preferences and are in accordance with our nutrition database.

The application shall be available on Web and iOS

The application shall not break any privacy laws

The application shall have Terms and Conditions for use of app

The application shall be usable offline and Sync with the cloud when app reconnect

The application shall allow user to enter food/workout and other updates in under 20 seconds

# ARCHITECTURE SPECIFICATION

TBD

Provide a graphical description of the architecture.

Specify the Java EE technologies used under the Client-Server Architecture together with the Service-Oriented Architecture if calling upon web services provided by other cloud software applications and APIs.

# DESIGN SPECIFICATION

TBD

Provide a graphical description (e.g., **storyboard**, **images, diagrams, drawings**, etc.) of the functionality of your cloud software application’s design.

Include UML diagrams, specifically class diagrams.

# DELIVERED SOFTWARE FUNCTIONALITY

TBD

Describe the functionality of your deployed cloud/mobile software application by using screenshots of user interfaces. Start with the URL for accessing your software. Then, screenshot by screenshot describe how your software is used. The entire software functionality must be described in sufficient detail.

# HOW TO BUILD AND RUN THE DELIVERED SOFTWARE

TBD (This section will contain item 1 below)

1. Develop step-by-step instructions on how to build and run your cloud and mobile applications on Dr. Balci’s iMac computer.
2. Place all your well-documented project files and associated resources on a flash drive.
3. On the day of delivery, come to Dr. Balci’s office with the flash drive and follow the written instructions in this section and deploy your cloud and mobile software applications to run on Dr. Balci’s iMac computer.

If your instructions are incomplete and/or inaccurate and I fail to rebuild and run your application, I cannot grade it.

# CONCLUSIONS

TBD

REFERENCES

TBD

Balci, O. (2016), “CS4984 Cloud Software Development Course Website,” <http://manta.cs.vt.edu/cs4984>

Oracle Corporation (2016), “NetBeans,” <https://netbeans.org/>

<< Include other references used >>

TEAM MEMBER CONTRIBUTIONS

Each team member provides a numbered list of his/her accomplishments in a concise and clear manner below.

|  |
| --- |
| Julia M. Binger |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Mykhaylo D. Bulgakov |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Allan F. Chua |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Duke J. Forsyth |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Filip D. Gouglev |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Amanda J. Kahn |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Mukund A. Katti |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Travis M. Lu |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Corey S. McQuay |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Ojas D. Mhetar |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Ryan T. Munz |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Benjamin F. Robohn |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Jared J. Schwalbe |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Timothy T. Street |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |
| Hung T. Vu |
| 1. Contribution 1 2. Contribution 2 3. Contribution 3   << continue >> |

PROJECT GRADING

* Quality of this Report 20%
* Quality of Code Documentation 10%
* Quality of Technical Work 70%