CMPT 475: Requirements Engineering, Fall 2022

Project: Personal Food Log App

The goal of this project is to gain real-life practice of the process involved in the requirement engineering of a large software project. In the real world, risks are rampant, project definition boundaries fluid, requirements at the beginning less known, and hard work is required to get on top of things. The same holds true for this project. The project is done in groups of 5.

The Case

We are in an age where Fitbit and similar personal health devices and apps have become prevalent. Digital Health Inc. is a leading company providing such personal health monitoring devices and services. While they excel in calculating daily energy expenditure and have a significant share of this market, they see a big gap in a different but related potentially lucrative market: energy intake. Digital Health Inc. would like to be able to log what food a person is eating on a daily basis. This, together with their existing energy expenditure solutions, would provide a complete energy intake/expenditure system that the company can then sell to healthcare providers, diet and health clinics, athletes, and individuals who either suffer from overweightness or simply want to stay fit.

Their initial research has revealed different methods for food logging, summarized nicely in this article (freely downloadable from SFU campus/VPN). Among those different methods, Digital Health Inc. strongly believes in the potential of *Smartphone Based VBM Systems*, where the user simply takes a picture of the food, and the app calculates the amount of calories and nutrition in that food. They believe this to be the most convenient, practical, and affordable method for logging daily food intake.

Their preliminary scan of existing VBM apps, such as MealSnap or Carbs&Cals, were disappointing. Those apps either were not automatic (someone at the other end actually has to look at the picture and give the estimates), or did not have reasonable accuracy, because even if they identify the food correctly, they do not measure the actual weight of the food. If the company wants to convince clinics and professional athletic businesses to buy the product, its accuracy needs to be a lot better. The company has recently come across a very promising system with reasonable accuracy (94.11%), but it seems that it's only a research proof-of-concept, and nowhere near a proof-of-concept, let alone a product.

As such, Digital Health Inc. has decided to create its own app. They tried to build the app in-house, but they failed! Hence, they are putting the project up for bids.

The Request for Bid (RFB)

Digital Health Inc. is about to post an official RFB for the said project. Meanwhile, they are already sending a draft to their existing suppliers. Your company has a lead who has been able to receive this RFB draft, which your marketing person summarizes as follows:

1. <u>Accuracy, in both food identification and calorie calculation, is the main selling point</u>! The company wants a system which is at or near the top of the list of existing VBM systems in terms of accuracy.

2. Like most other personal apps, ease of use and pleasantness is highly desired.

3. The system must detect each food ingredient that is <u>visually</u> detectable. For example, determining that a food is *salad* is not accurate enough. It should instead report that the food consists of *carrots*, *lettuce*, *tomato*, etc.

4. The system should assume that what exists on the surface of the food, continues down to the bottom of it more or less uniformly. For example, if pieces of beans are visible in a soup, the system should assume the same density of beans throughout the whole soup.

5. The weight of each ingredient should be calculated. This can be done from calculating the visible surface of the ingredient, then calculating the volume of the ingredient, and then using food density tables to convert volume to weight. Once the weight is known, existing nutrition tables can be used to convert from weight to calories, for each ingredient. We can then calculate the total amount of calories as the sum of the calories of all ingredients. A more detailed description is given in this research paper, which again is just a proof-of-concept.

6. To measure the actual dimensions of the image (hence measure the size of food ingredients), the system should use auto-calibration techniques, such as this method. It should not use external objects in the image, such as a coin, the user's thumb, etc.

7. Some ingredients cannot be visually detected, such as the amount of salt, pepper, oil, seasoning, etc. Since no other practical existing method (including clinical methods widely in use at this time) can detect those things either, this limitation is acceptable.

8. The system must be highly scalable. The app is going to upload millions of food images every day. So storage, processing, and communication need to be highly scalable. Therefore, Big Data and cloud computing principles should be considered (such as this work and this work), as should the fact that the company's existing infrastructure uses the Amazon Cloud. This is a firm requirement; no changes should be made to the existing infrastructure.

9. The system should use Artificial Intelligence (AI) for food detection. It can use many existing and publically available food datasets for training. The AI training should be incremental. In other words, as new images are uploaded, the AI module should improve its model with those new images quickly, as opposed to re-training itself completely from scratch with the entire dataset, which is neither practical nor cheap.

10. The app itself should run on the most common mobile platforms: Android and iOS.

Your company

Your company has decided to work on the above and come up with a competitive bid. Your team has been asked to come up with the following documents to be included in the bidding package:

1. Requirements Specifications document, consisting of:
   - Business Requirements Specification (BRS)
   - Stakeholder Requirements Specification (StRS)
   - System Requirements Specification (SyRS)
   - Software Requirements Specification (SRS)
2. Prototype
3. Preliminary Design, showing the main classes, their associations, methods, and attributes

The contents of the Requirements Specifications must comply with IEEE standard 29148:2018 Section 9. When applicable, UML diagrams must be used (Use Case Diagrams and Use Case Scenarios, Use Case Realization Diagrams, Design Diagrams, Sequence Diagrams, etc.)

Project Details

Please carefully note the deadlines given in the table below. Plan and spread the work so that you have sufficient time to finish the deliverables on time. There is considerable amount of work for each deliverable, so start early and **do not wait until the last few days!**

Deliverable	Due
Project Assigned	Effective Immediately
Team formations	March 4, Sign up on Canvas under "Project groups"
Mockup Prototype presentation and feedback	• week of March 14 • will be scheduled by TA • each team will e-meet with the TA for 10 minutes
BRS, StRS	March 28, through Canvas
• BRS (updated), StRS (updated), SyRS, SRS, Preliminary Design, Prototype (essentially updated mockup) • Post Performance Analysis (PPA) • "contributions list", indicating the % of contribution from each team member	April 8, through Canvas

Evaluation:

Mockup Prototype	10%
BRS, StRS (first round)	20%
BRS, StRS, SyRS, SRS, Preliminary Design, Prototype	65%
PPA	5%

Keep in Mind

The first and most important aspect of this project is to acknowledge that requirements engineering is a sizable project in and of itself – without proper attention to it, this project cannot be completed with an acceptable level of quality in the time period specified. Hence, you will develop an appreciation for challenges in requirements engineering. Many details are missing or vague, while existing information is abstract and open to interpretations. Use your creativity, plus good requirement engineering practices, to successfully complete the project.