Alex Boss (ajb112)
Ryan Pond (rp159)
Edward Ruby (ejr25)
Bilva Sanaba (bgs21)
Katherine Van Dyk (kmv10)

Milestone 1: Duke Food Pal (Open Project)

I. Overview

It can be difficult for Duke students to maintain healthy and balanced diets. This is due to several reasons. To start, there is no comprehensive list of all food options at Duke that includes price and nutritional information. Additionally, even if there was such a list, it is incredibly challenging for a busy Duke student to take time out of their day to look through a list of food options to determine what they want. We believe that Duke students get into habits of eating at the same places day to day, and are not aware of other foods which they may enjoy, as well as other healthier options available. Along with this, we also believe that most students are completely unaware of how much they are spending until their food points are uncomfortably low.

In order to face these problems, we plan to create an application that can track students' food point spending habits and generate useful metrics. Tracking food points would allow us to easily display metrics such as: how many food points did you spend today, how many should you have spent, where are you spending most of your money etc. These easily available metrics will enable students to make better decisions on food with money in mind. The application will also give suggestions for similar food that is cheaper if spending is a concern, further reducing the time needed for students to see other worthwhile food options.

Additionally, we plan to compile all information for available Duke food into a database that can be accessed via our application. The application would be able to allow students to easily search for things such as: what restaurants are open right now, where can I get a low-fat food, where can I get a smoothie during the day, etc. These options will enable students to better know about their possible options.

Finally, we plan to create an option for students to keep track of their macronutrient intake. Students will be able to log the food and drinks that they consume and look through an interface which gives information of what nutrients they are not getting enough of or what they are getting too much of. In order to improve this from the many other options available for general food tracking, we plan to use our database of Duke food to make this incredibly easy for a student to log what they eat. Combining this with live-tracking of where students spend their food points, the application will be able to determine where a student bought food and then give suggestions for what item they possibly could have eaten, further improving the process of logging meals. With the food logging, we also plan to include options similar to many other apps, such as suggesting nutrients or meals that could improve a diet.

II. Importance

The project will be useful to Duke students who are concerned about their health or spending habits. Our project is similar to the service of MyFitnessPal which has a huge following. One of the main issues with applications is the time required to use them. The project is important because it will provide a Duke-specific version of MyFitnessPal which is easier to use due to its suggestions being based on a smaller set of Duke specific food options, and due to its being able to track food point spending. Likewise, the project is important on a more general scale because it allows all students a way to quickly notice ways they could improve their health and lives with minimal effort.

III. Approach

A. Framework

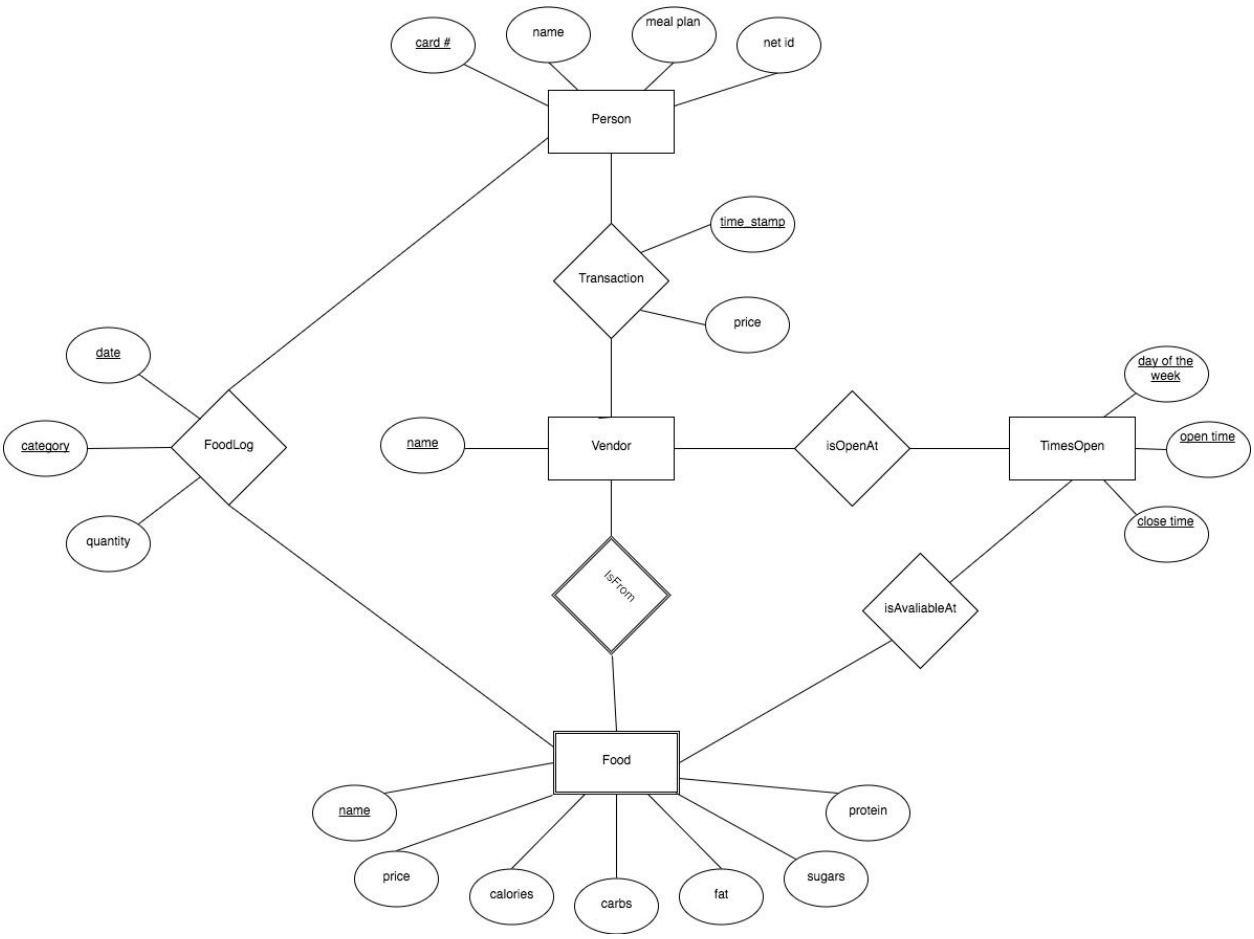
We will be writing an iPhone application which we plan to make available on the app store for all Duke students by the conclusion of the project. We will need a server to maintain our database and a SQL interface to interact with our application. We will be using the iOS development kit, a SQL server to host our database, and some framework to connect the two that will be determined in the future. We are leaning toward using a SQL API integration but we have not looked into the capability of Swift for this.

We will also be interfacing with the Duke Card API to track where students spend food points and how much. Our SQL server will store this information as well as nutritional data like the following: https://www.looppizzagrill.com/media/1062/nutritional-analysis_0614.pdf. We plan to create a database of all the food options at Duke compiled from various online sources. For those vendors who are not chains and therefore not required by law to have published nutritional information, we plan to use data of a similar meal in replacement.

When students log a meal purchased, we will look up that meal, or a similar meal if that meal does not exist, and get its nutritional info (calories, protein, fat, carbs, sugar, etc.). We can interface with the MyFitnessPal API or another nutritional database in order to find meals if certain Duke vendors fail to provide nutrition info. Over time, students will accumulate a macronutrient and calorie profile. We can then help balance the profile by suggesting meals and vendors. We can also help students manage their food points so they don't run out, perhaps by finding cheap meals.

B. Back-end

ER design:



For our database structure, we plan to implement this Entity-Relationship Diagram in SQL. We believe that it will allow minimal repetition of data and tables that are fully decomposed (ie removed of independent variables). The following implementation is present in our source code, attached to this proposal. Not pictured by the diagram, we also implemented a trigger to prevent food from being represented as available when its vendor is closed.

C. Front-end

Three main windows in the Food Pal app will encapsulate data entry, visualization, and analysis. The 'Diary' window, allows students to connect purchases on their Duke Cards to certain meals, or manually enter a meal based on a list of options from a nutritional database. As shown in Figure 1 to the right, students will have the option to add items to three meals: breakfast, lunch and dinner. First they select whether they'd like to 'Connect a Recent Purchase', or whether they'd like to manually enter a food item. The 'Connect a Purchase' tab uses the spending information from the DukeCard API, which details where a student spent their money, but not what food items they purchased. To make the process of entering food as easy as possible on the user end, 'Recent Vendors' will only show options based on where a student recently spent money. A student then selects a certain food item from a list of those sold by that specific vendor.

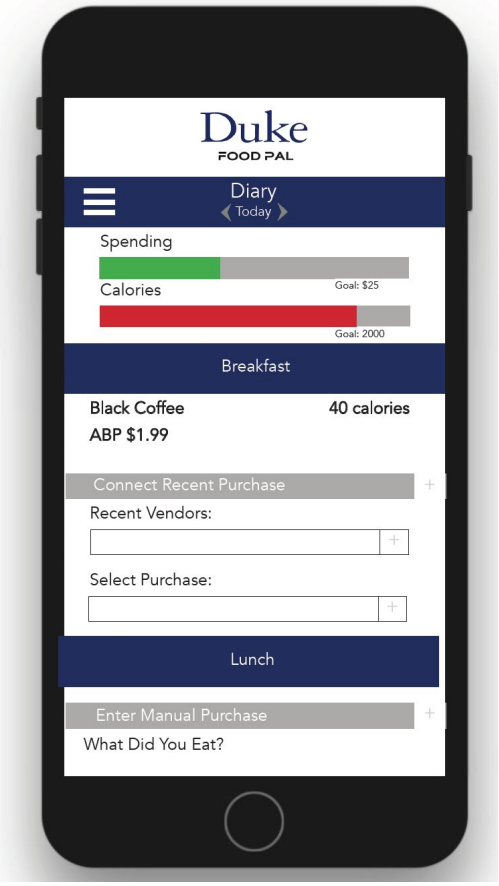


Figure 1. Data Entry Window

If a student consumes food that isn't sold by a Duke Vendor, they can manually enter that meal by choosing a food group and then a specific meal. Based on data from a nutritional database, the calories from the manually entered meal can be estimated and used for nutritional analysis.

For easy access, Spending and Calorie tallies for the day are shown at the top of the 'Diary' window. When a student first downloads the app, they have the option of setting spending and calorie goals to certain thresholds. Using information from the DukeCard API, including how much money a student has spent within a certain time frame, we can easily visualize this data.

The next window is the Nutrition Analysis window (Figure 2), which gives a visual display of where students are eating most, but also where they are getting most of their macronutrients- carbohydrates, proteins, and sugars, as well as general calorie counts. These counts are visualized as a pie chart. When the student first lands on the 'Nutrition' page, the pie chart displays generally where the student has spent the most money. The student can then click on one of the tabs listed below to see in more detail where they have consumed the most

carbs, proteins, etc. Instead of displaying just percentages in these instances, the pie chart will also include quantities of the specific nutrient.

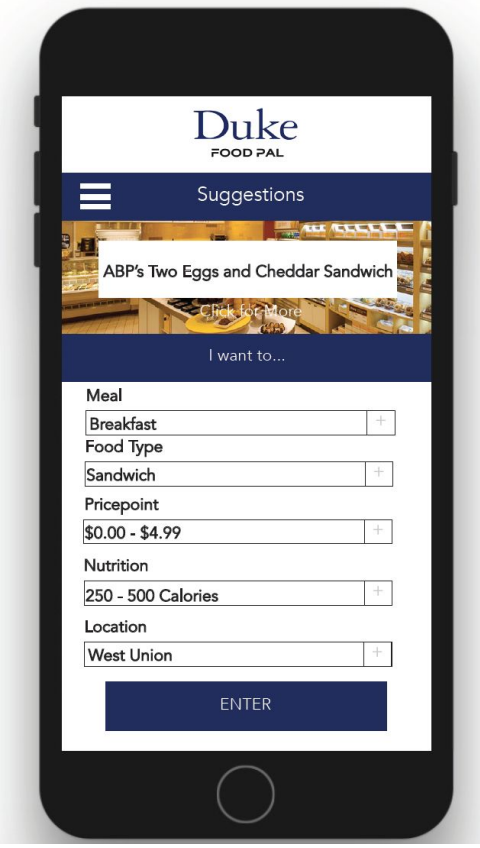


Figure 2. Suggestion Window

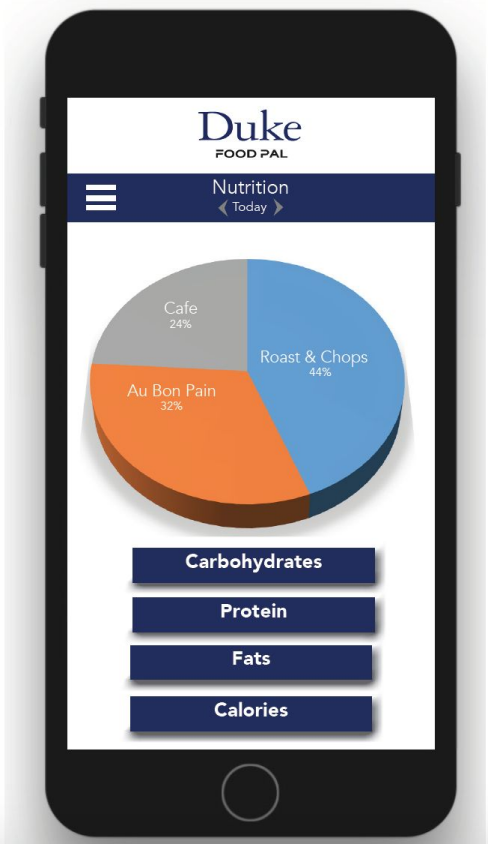


Figure 3. Nutrition Analysis Window

Finally, the Suggestion Window (Figure 3) analyzes student spending, nutrition habits, and previous purchases to come up with a suggestion for a meal. Students can enter information into one or all fields detailing what meal they want a suggestion for, the food type they'd prefer, the preferred location, and specific price and calorie points. Once the student hits the 'Enter' button, a specific order pops up on the screen. If a student is not satisfied with the option they've been given, they can click the 'Click for More' option to scroll through all items. Future optimizations include a general health plan that outputs a meal based on the nutrition profile of a particular student.

IV. Previous Work/Systems

One of the most similar options to the food logger is MyFitnessPal. This is similar to our application's food logging; however, it lacks several advancements that we intend to develop. First, since our application is Duke specific, the food options in our application will be easier to associate with Duke foods, and therefore easier to use for Duke students. Additionally, the process of logging foods is greatly reduced in comparison to MyFitnessPal since our application

will be able to determine where you bought food based on food point spending, and then appropriately give suggestions for what to log. One idea we want to implement from MyFitnessPal would be the suggestions, which analyze your nutrient intake and create suggestions based on your user goals.

As for Duke-specific projects, we were able to find one other project relating to food points. This project appeared to track food point spending and then provide metrics for where you spend money and how much money you have left among other things. This does not touch into the idea of using the data from the Duke API in order to analyze the actual food you are eating. We plan to use ideas from this since these metrics are of a related topic and important for students, but our project is well distinguished due to the different use of the data we can receive from the Duke API.

Another application that is similar to our idea is BillGuard. It keeps track of where you spend your money monthly and will give a summary on this spending. However, what this application lacks is the Duke-specific component that will allow us to tap into food points rather than credit card spending, and allow us to make specific restaurant recommendations and provide a level of interactivity that is unavailable in BillGuard and similar applications.

Overall, there are many independent applications that utilize a few of our ideas, however no application has an inclusive program that merges food point spending, nutritional value, and gives recommendations to optimize both of those areas. There is a web app that shows what restaurants are open, and gives the menus of the restaurants. However, it does not analyze the nutritional value of the menu items, and it is not interactive with what you spend/where you eat, whereas our application does all of these things.

V. TA Discussion

Our group spoke with Seon Kang about Duke Food Pal. After hearing our plan and initial thoughts on our design he provided feedback. Given it was an open project, Seon suggested that we focus on architecting our system such that it allows us to complete the core aspects of our project and then add additional features on top of that. It is important that we design our project in proper way so that even if some features can not be finished, we can still turn in a fully working product. Seon also mentioned that it will become clear what ideas are feasible and what are not as we begin the actual creation of our product. Additionally, Seon challenged us to consider features that use our database in more complex and interesting ways. The database is essential to the core feature of our app, the food tracker, but it is somewhat straightforward in how the database is used. Therefore, we hope to incorporate functions like food recommenders based on nutritional history and price points that require more complex queries and manipulation of our database. Finally, Seon warned us of the possible risk of make an iOS application and that the TAs would likely not be able to assist with any problems we will encounter with swift. Ultimately, Seon seemed positive about our idea and encouraged us to start as early as possible.