University of British Columbia, Vancouver

Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: ____1____

Date: __2024-02-09_____

Group Number: ____49_____

Name	Student Number	CS Alias (Userid)	Preferred Email Address
Ian Qin	10687325	p6e5b	qinyi2333@outlook.com
Marvin Wu	76668078	r7d3b	mwu584@student.ubc.ca
Winston Shin	35537142	y1c2b	winstonshin@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

What is the domain of the application?

Our prototype would fall under the self-care/planning/logistics domains. We are creating an app which allows its users to plan their meals in much detail over a week (this duration is subject to change). The users would initially input their budget, location, nutrition/food preferences, and even preferences to eat in or out as a query, and the app would build a set meal schedule best accommodating their preferences. This meal schedule would be a deep dive in planning their meals such as suggesting which grocery stores to visit, which restaurants to eat at, and finally reporting the user's hypothetical nutritional intake/expenses should they follow this meal plan. After this initial schedule, users could choose to save this schedule or further modify to best fit their needs.

What aspects of the domain are modeled by the database?

Dishes would be one major aspect to be modeled by the database. Whether it be homemade or from restaurants, dishes would have an associated cost and their respective nutritional facts. Should the dish be home made, it would include an ingredient list where each item is associated with a corresponding grocery store which carries that item. Another major aspect to be modeled by the database would be the user's personal information and preferences. Their nutritional needs could be broken down into macronutrients/micronutrients. Their budget could be divided into how much is allocated to eating in versus eating out. And the last major aspect to be modeled by the database would be the grocery stores and restaurants. Grocery stores would list the items they sell, and restaurants the dishes they sell.

What functionality will the database provide?

The database will be able to let users retrieve data for themselves or for informational purposes. That could be getting saved meal plans for themselves, seeing nutritional information for certain foods, seeing ingredients for recipes, or determining dishes at certain restaurants. Users can also save or update their data by inserting it into or updating the database, such as for their personal info and current meal schedules with the associated costs, locations, recipes, and dishes. Lastly, users can delete their certain meal schedules from the database if necessary.

Description of the application platform:

The application platform will be through a Node.js environment to create a React app. Therefore, the technology stack that we will use involves Oracle, Express, React, and Node.js, and the project will be written all in Javascript. The frontend will involve using the Material UI library to develop React components, while the backend will involve using Node.js, Express framework, and Oracle module. The libraries that we will use may change as we progress in the project.

Entities, attributes, and relationships in the ER diagram:

User (entity)
- Budget
- Phone number
- Address
- Name
- Nutrition Requirements
- ID

Has (relationship, connecting User and Meal plan)
- Date

Meal plan (entity)
- Name
- Duration

Contains (relationship, connecting Meal plan and Meal)

Meal (entity)
- Name
- Dining Time
- Cuisine

Composed Of (relationship, connecting Meal and Food)

Food (entity)
- Name
- Cost

Has (relationship, connecting Food and Nutritional Facts)

Nutritional Facts (entity)
- Calories
- Total Protein
- Total Sugars
- Total Fats
- Name

IsA (relationship, connecting Food to Take Out and Home Made)

Restaurant (entity)
- Rating
- Address
- Name
- Cuisine

Serves (relationship, connecting Restaurant and Take Out)

Take Out (entity)
- Discount

Home Made (entity)
- Time Required

Made with (relationship, connecting Home Made and Recipe)

Recipe (entity)
- Name
- Instructions

Contains (relationship, connecting Recipe and Ingredients/items)
- Quantity

Ingredients/items (entity)
- Name
- Category

Sells (relationship, connecting Ingredients/items and Grocery Stores)
- Cost

Grocery Stores (entity)
- Name
- Address
- Open hours
- Rating