<NUTRiAPP & RELEASE 1>

Design Documentation  
Prepared by NUTRiAPP 1>:

* Aiden Green <ajg7654@rit.edu>
* Chinedu Tabansi <cdt2390@rit.edu>
* Daniel Chung <dec8768@rit.edu>
* Elizabeth Ostrowski <ego4995@rit.edu>

Summary 2

Domain Model 3

System Architecture 4

Subsystems 5

FoodDatabase (Composite) 5

Actions (Command) 8

UserGoals (Strategy) 11

Status of the Implementation 13

Appendix 14

# Summary

NUTRiAPP is a command line application that helps you track and achieve your weight and exercise goals. The app can help you gain, lose, or maintain your weight depending on what you select and will keep track of your calorie intake depending on what foods you eat and provide you recipes based on what food you have. If you go over or under your weight goal the application will automatically switch to keep you on track. For example, if you want to maintain your current weight but you start gaining weight, the app will switch to helping you lose weight to get back on track.

NUTRiAPP also keeps track of different foods and their nutrition information to make sure you’re getting a balanced diet as well as enough calories to meet your goals. It keeps track of all the food in your pantry and alerts you when you need to buy more ingredients when they get low. It uses those ingredients to help you create recipes and meals, finding the best options to help you reach your weight goals.

There are also options for working out. You can select from low, medium, and high workouts. If you eat over the recommended calories, the app will suggest a workout for you to do so you can lose weight and keep in track with your weight goals. All of the workouts you did, along with the weight at the end of the day, are recorded in your history so you can go back and see how your weight has been over time.

NUTRiAPP will also keep users to keep a history of their daily calories consumption, workouts performed and the user’s personal statistics and details. Because of lack of time, no database implementation was done so most records will be stored in csv files. The project has a flexible design, keeping in mind that there could be future changes can be added without having to rewrite the entire application.

# Domain Model

This section provides a domain model for the project.

Chart, diagram

Description automatically generated

# System Architecture

This section provides a model of the subsystem components that make up the overall software architecture for the project. Draw the subsystems as simple boxes with relationships between them. Provide a narrative that describes the responsibilities of each component and the interfaces that are provided between subsystems.

# Diagram Description automatically generated

# Subsystems

This section provides detailed design for specific subsystems described in the system architecture.

## FoodDatabase (Composite)Text, timeline Description automatically generated

Diagram

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Name:** FoodDatabase | | | **GoF pattern:** Composite |
| **Participants:** Daniel Chung, Eliza Ostrowski | | | |
| **Class** | **Role in GoF pattern** | **Participant's contribution in the context of the application** | |
| FoodDatabase | Client | Works with all elements through the component interface. The FoodDatabase can store Leaf as well as composites. | |
| Food | Component | Defines interface declares the common operations for both leaf and components. | |
| Ingredient | Leaf | Is the most basic element of the tree, doesn’t have sub-elements. Everything is a either an Ingredient or a collection of ingredients. | |
| Recipe | Composite | Represents a collection of ingredients. It adds up all the individual ingredient’s nutrition and displays  instructions for the user. | |
| Meal | Composite | Represents a meal. A meal is a collection of recipes and combines all the nutrition of all recipes to get a total count. | |
| **Deviations from the standard pattern:** None | | | |
| **Requirements being covered:** 4) The app will keep track of various forms of food. 5) Ingredients are the most basic food type. In addition to the basic properties above, ingredients also have an amount in stock. 6) Recipes are also a type of food. 7) Meals are also a type of food. 8) The app helps users to prepare meals. | | | |

## 

## Actions (Command)

# Diagram Description automatically generated with medium confidence

Diagram

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Name:** Action | | | **GoF pattern:** Command |
| **Participants:** Daniel Chung | | | |
| **Class** | **Role in GoF pattern** | **Participant's contribution in the context of the application** | |
| ParseAction | Invoker | The ParseAction class main responsibility is invoking the commands to send to the receiver. | |
| Action | Command | The Action interface is responsible for laying the basic structure of a class that implements Action (Command). | |
| SeeShoppingList | Concrete  Command | A concrete command that calls to see the  FoodDatabase shopping list. | |
| Eat | Concrete  Command | A concrete command that removes a food from the database and then registers the consumed food into Goal. | |
| Workout | Concrete  Command | Represents a meal. A meal is a collection of recipes and combines all the nutrition of all recipes to get a total count. | |
| CheckHistory | Concrete  Command | A concrete command that allows the user to navigate through the his history. | |
| FoodDatabase | Receiver | The Database class is responsible for receiving the commands from the invokers and performs the requested command. | |
| Goal | Receiver | The Goal class is responsible for receiving the commands from the invokers and performs the requested command. | |
| Profile | Receiver | The Profile class is responsible for receiving the commands from the invokers and performs the requested command. | |
| **Deviations from the standard pattern:** There are several receiver classes in the implementation, those being FoodDatabase, Goal and Profile. | | | |
| **Requirements being covered:** 4) The app will keep track of various forms of food 9) The app helps the users to create shopping list based on specific criteria. 10) App allow sto track workouts. 12) The user may browse their personal history. | | | |

## User Goals Subsystem (Strategy)

Diagram

Description automatically generated

A picture containing diagram

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Name:** User Goals Subsystem | | | **GoF pattern:** Strategy |
| **Participants:** | | | |
| **Class** | **Role in GoF pattern** | **Participant's contribution in the context of the application** | |
| Goal | Context | Represents the user in the system. Keeps track of their personal information, weight, and the goal they are striving towards. | |
| WeightGoal | Strategy | Defines the calorie loss or intake that the user will be striving to get weekly. | |
| MaintainWeight | Concrete  Strategy | A goal that a user chooses that changes the calorie intake goal percentage per week such that it should help keep the user’s current weight the same. | |
| LoseWeight | Concrete  Strategy | A goal that a user chooses that changes the calorie intake goal percentage per week such that it should help lower the user’s weight on average by 1-2 lbs. | |
| GainWeight | Concrete  Strategy | A goal that a user chooses that changes the calorie intake goal percentage per week such that it should help rise the user’s weight on average by 1-2lbs. | |
| **Deviations from the standard pattern:** None | | | |
| **Requirements being covered:** 2) The user may choose from one of several different goals. 3) In some circumstances, the app will automatically transition from one goal to another. | | | |

# Status of the Implementation

As of now, the UserGoals and FoodDatabase subsystems are successfully implemented, we are able to read/write to the ingredients and keep track of the user’s current progress saving to profile, add and remove from the FoodDatabase. We are also able to change the Goal of the user depending on if he’s over or under his desired weight. Currently missing is the main loop for the program, Command subsystem, creating custom recipes/meals and creating Threads to keep track of the day.

# Appendix

This section provides fine-grained design details for all of the classes in your design. You will capture this information using the CRC (Class-Responsibilities-Collaborators) card format below.

|  |  |
| --- | --- |
| **Class:** App |  |
| **Responsibilities:** Main class for the program. |  |
| **Collaborators:** |  |
| **Uses:** Profile, FoodDatabase, Goal | **Used by:** ... |
| **Author:** Daniel Chung |  |

|  |  |
| --- | --- |
| **Class:** Profile |  |
| **Responsibilities:** Keep the user’s profile information, save and load profiles. |  |
| **Collaborators:** |  |
| **Uses:** App | **Used by:** ... |
| **Author:** Daniel Chung |  |

|  |  |
| --- | --- |
| **Class:** Recipe |  |
| **Responsibilities:** Keep a collection of ingredients that composes a recipe, and calculates the total calories. |  |
| **Collaborators:** |  |
| **Uses:** Food | **Used by:** FoodDatabase, App, Goal |
| **Author:** Daniel Chung |  |

|  |  |
| --- | --- |
| **Class:** Ingredients |  |
| **Responsibilities:** Keep information of single ingredients. |  |
| **Collaborators:** |  |
| **Uses:** Food | **Used by:** FoodDatabase, App, Goal |
| **Author:** Daniel Chung |  |

|  |  |
| --- | --- |
| **Class:** FoodDatabase |  |
| **Responsibilities:** Keep information of the ingredients.csv file. Read and write to the ingredients.csv and return foods that are requested |  |
| **Collaborators:** |  |
| **Uses:** Food | **Used by:** App, Goal |
| **Author:** Daniel Chung |  |

|  |  |
| --- | --- |
| **Class:** Meal |  |
| **Responsibilities:** Keep a collection of recipes that composes a meal, and calculates the total calories. |  |
| **Collaborators:** |  |
| **Uses:** Food | **Used by:** FoodDatabase, App, Goal |
| **Author:** Daniel Chung |  |

|  |  |
| --- | --- |
| **Class:** WeightGoal |  |
| **Responsibilities:** Contains the user's weight goal (lose, maintain, gain). Suggests workouts, foods, and a calorie intake. |  |
| **Collaborators:** |  |
| **Uses:** Workout, Profile, FoodDatabase | **Used by:** Workout |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** Workout |  |
| **Responsibilities:** A suggested workout for a user. It helps a user lose weight/subtract calories |  |
| **Collaborators:** |  |
| **Uses:** WeightGoal | **Used by:** WeightGoal |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** History |  |
| **Responsibilities:** A history of a user. Keeps track of a user’s weight and workouts so they can look back on it. |  |
| **Collaborators:** |  |
| **Uses:** Profile | **Used by:** Profile |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** MediumWorkout |  |
| **Responsibilities:** Defines what a medium work out is and how many calories it burns |  |
| **Collaborators:** |  |
| **Uses:** Workout | **Used by:** Goal |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** LowWorkout |  |
| **Responsibilities:** Defines what a low work out is and how many calories it burns |  |
| **Collaborators:** |  |
| **Uses:** Workout | **Used by:** Goal |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** HighWorkout |  |
| **Responsibilities:** Defines what a high work out is and how many calories it burns |  |
| **Collaborators:** |  |
| **Uses:** Workout | **Used by:** Goal |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** LoseWeight |  |
| **Responsibilities:** calculates the target calories that need to be consumed, based on the current weight and the target weight. |  |
| **Collaborators:** |  |
| **Uses:** WeightGoal | **Used by:** Goal, App |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** GainWeight |  |
| **Responsibilities:** calculates the target calories that need to be consumed, based on the current weight and the target weight. |  |
| **Collaborators:** |  |
| **Uses:** WeightGoal | **Used by:** Goal, App |
| **Author:** Eliza Ostrowski |  |

|  |  |
| --- | --- |
| **Class:** MaintainWeight |  |
| **Responsibilities:** calculates the target calories that need to be consumed, based on the current weight and the target weight. |  |
| **Collaborators:** |  |
| **Uses:** WeightGoal | **Used by:** Goal, App |
| **Author:** Eliza Ostrowski |  |