

# FoodPants

*Iteration II*



“

Food pants is the  
greatest new app  
coming to the  
market. I wish I  
found it sooner.”

- Abe Lincolns



# Demo User Interface

The screenshot displays the FoodPants application interface. On the left is a vertical sidebar with four red buttons: 'Pantry', 'Recipes', 'Nutrition', and 'Shopping List'. The main area shows a table of ingredients with their quantities. A 'Create Recipe' modal is open in the foreground, featuring input fields for 'Name', 'Servings', and 'Ingredients', along with a 'Submit' button.

Ingredient	Quantity
Banana	3
Apple	2
Oreos	5
Vanilla Extract	2
Sugar	10
Melon	2

Ingredient	Quantity
Muffin	1
Ribeye	4
Bread Flour	1

Search

**Create Recipe**

Name

Servings

Ingredients

Submit



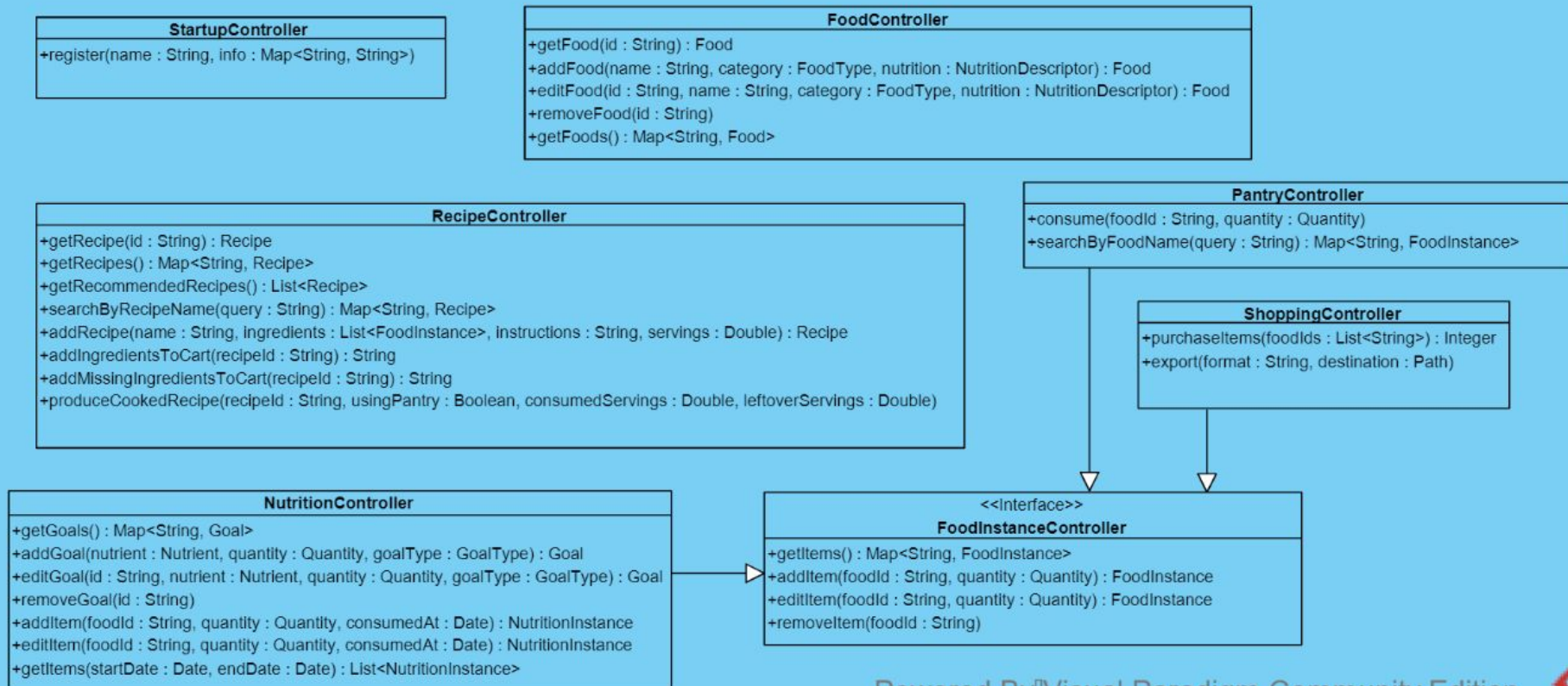




# Class Diagrams



UI



<<enumeration>> Nutrient
TotalFat
SaturatedFat
TransFat
Cholesterol
Sodium
TotalCarbohydrate
DietaryFiber
TotalSugars
Protein

<<enumeration>> FoodType
Fruit
Vegetable
Protein
Dairy
Grain
Other

Services
+RECIPE_SERVICE : RecipeService
+USER_SERVICE : UserService
+FOOD_SERVICE : FoodService
+ID_SERVICE : IdService
+PANTRY_SERVICE : PantryService
+NUTRITION_SERVICE : NutritionService
+SHOPPING_SERVICE : ShoppingService

IdService
#ids : Set<String>
+getId() : String

UserService
+register(name : String, info : Map<String, String>)
+validateUserInfo(info : Map<String, String>)

User
#name
#weight
#gender
+getName() : String
+setName(name : String)
+getWeight() : Double
+setWeight(weight : Double)
+getGender() : String
+setGender(gender : String)

ShoppingService
+removeItems(foodids : List<String>)
+addItems(foodInstances : List<FoodInstance>)
+export(format : String, destination : Path)

FoodInstanceService
#items : Map<String, FoodInstance>
+getItems() : Map<String, FoodInstance>
+setName(name : String, foodInstance : FoodInstance)
+addItem(foodInstance : FoodInstance)
+editItem(foodInstance : FoodInstance)
+removeItem(foodId : String)
+createInstance(foodId : String, quantity : Quantity) : FoodInstance

PantryService
+removeItem(foodId : String, quantity : Quantity)
+getMissing(foodInstances : Map<String, FoodInstance>) : Map<String, FoodInstance>
+addItems(foodInstances : List<FoodInstance>)

NutritionDescriptor
-calories : Integer
-servingSize : Quantity
-nutrientGrams : Map<Nutrient, double>
+getCalories() : Integer
+setCalories(calories : Integer) : void
+getServingSize() : Quantity
+setServingSize(servingSize : Quantity) : void
+getNutrientGrams() : Map<Nutrient, double>
+setNutrientGrams(nutrientGrams : Map<Nutrient, double>)

NutritionService
#goal : Goal
+getGoal() : Goal
+setGoal(goal : Goal)
+createInstance(id : String, foodId : String, quantity : Quantity, consumedAt : Date) : NutritionInstance
+addItem(foodInstance : FoodInstance, numServings : Double) : FoodInstance
+addItem(nutrition : NutritionInstance)

RecipeService
#recipes : Map<String, Recipe>
+getRecipe(id : String) : Recipe
+getRecipes() : Map<String, Recipe>
+getRecipesByIngredients(ingredients : List<FoodInstance>) : List<Recipe>
+getRecipesNameStartsWith(query : String) : Map<String, Recipe>
+addRecipe(Recipe)
+getIngredients(recipeId : String) : List<FoodInstance>
+produceCookedRecipe(recipeId : String, usingPantry : Boolean, consumedServings : Double, leftoverServings : Double)

Recipe
#instructions : String
#ingredients : List<FoodInstance>
#servings : Double
+getIngredients() : List<Food>
+setIngredients(ingredients : List<Food>)

FoodService
#foods : Map<String, Food>
+getFoods() : Map<String, Food>
+getFood(id : String) : Food
+addFood(food : Food)
+editFood(food : Food)
+removeFood(id : String)

Food
#name : String
#category : FoodType
#nutrition : NutritionDescriptor
+createInstance(quantity : Quantity) : FoodInstance
+getName() : String
+getCategory() : FoodType
+getNutrition() : NutritionDescriptor
+setNutrition(nutrition : NutritionDescriptor)

FoodInstance
#quantity : Quantity
+setQuantity(quantity : Quantity)
+getQuantity() : Quantity

NutritionInstance
-consumedAt : Date
+getConsumedAt() : Date
+setConsumedAt(consumedAt : Date)

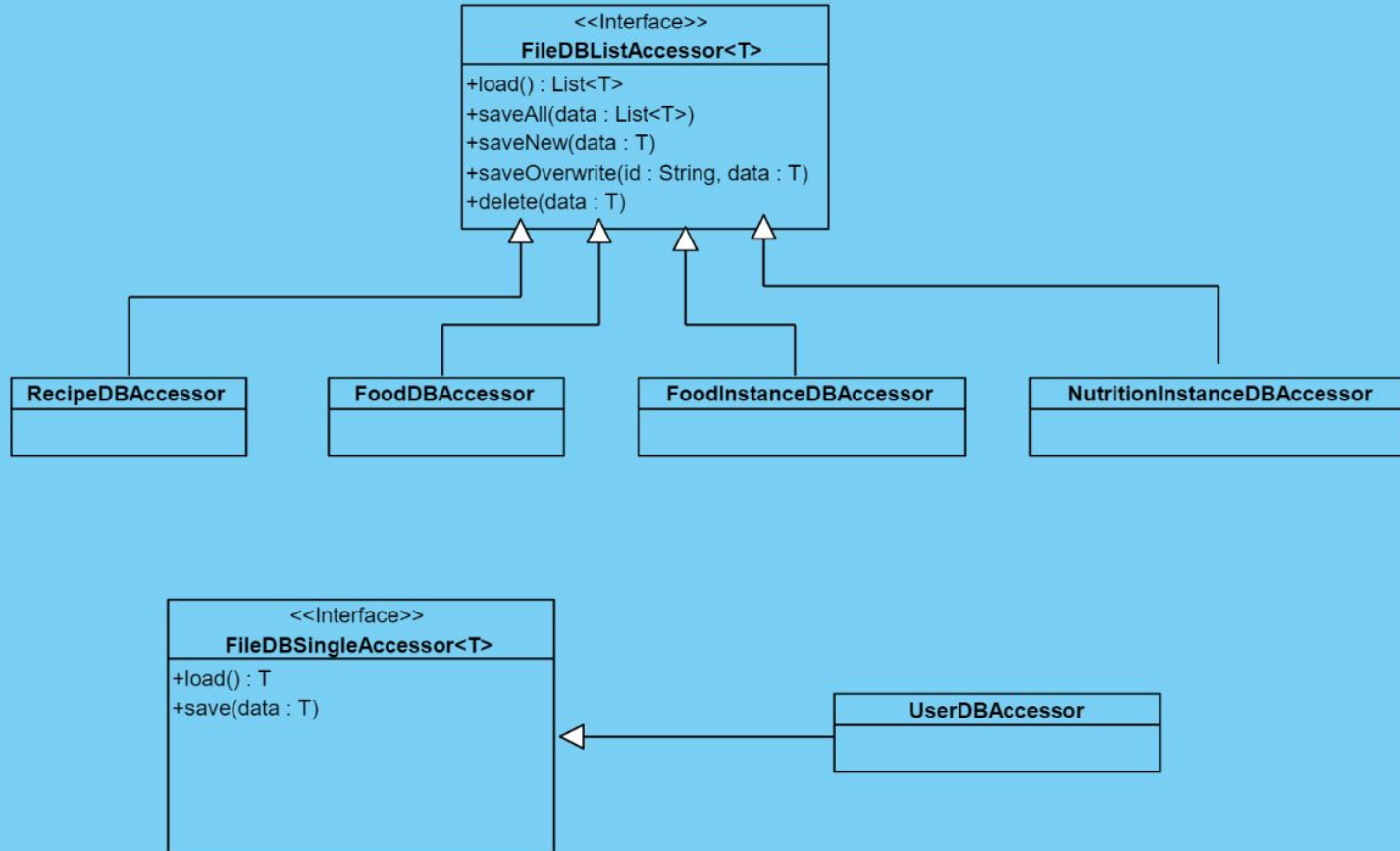
Goal
-goalType : GoalType
-dailyQuantity : Quantity
+isAttained(quantity : Quantity)
+getQuantity() : Quantity
+setQuantity(quantity : Quantity)
+getGoalType() : GoalType
+setGoalType(goalType : GoalType)

IdObject
-id : String
+IdObject(id : String)
+getId() : String

<<enumeration>> GoalType
GoalType



## Database



Class Name	PantryController
GRASP Pattern	Controller
Description	A controller for the pantry. This class handles input events directed at the Pantry portion of the application. It also serves to separate the UI from business logic to maximize reusability.

Class Name	PantryService
GRASP Pattern	Pure Fabrication
Description	This class handles functionality relating to the pantry within the application. It is dedicated to handling business logic.

Class Name	Food
GRASP Pattern	Information Expert
Description	An information expert for food types. This class contains all necessary information to store a food type object.

Class Name	RecipeDBAccessor
GRASP Pattern	Low Coupling
Description	This database accessor helps separate the service from the database and keeps the services separated from each other.

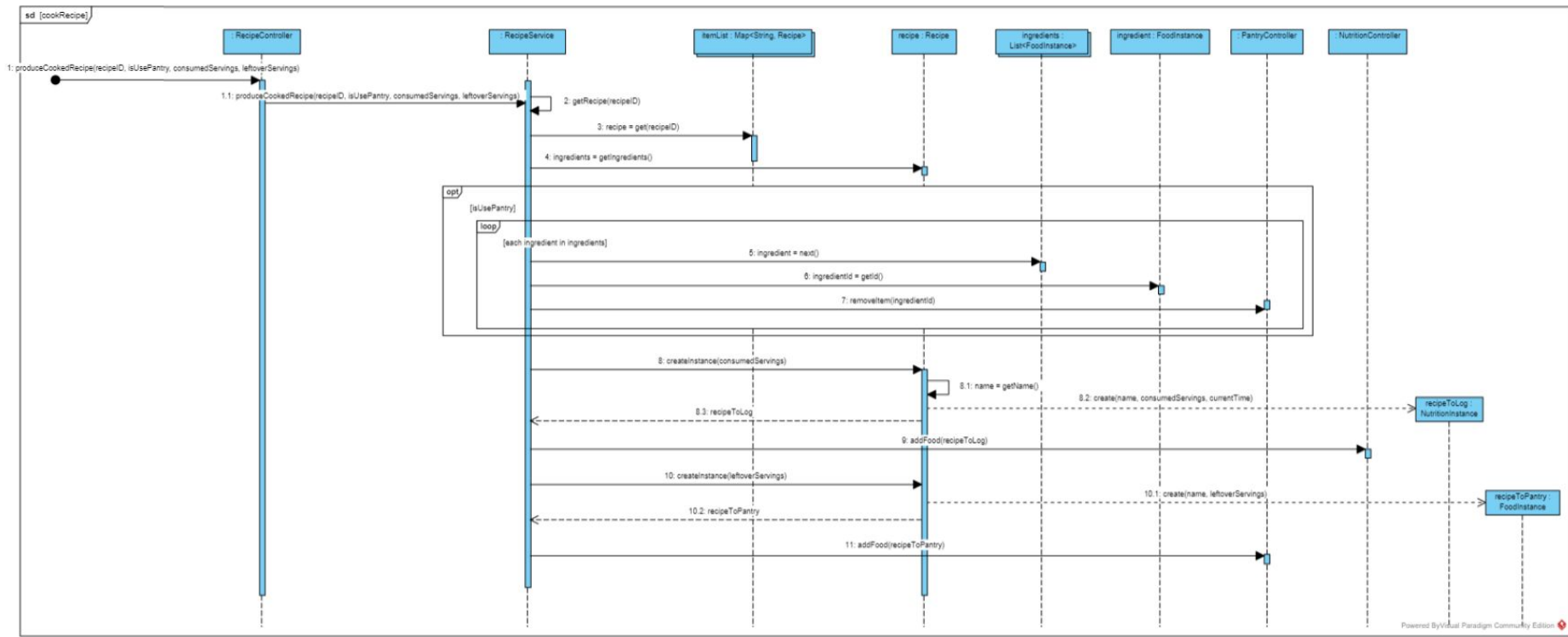
Example GRASP





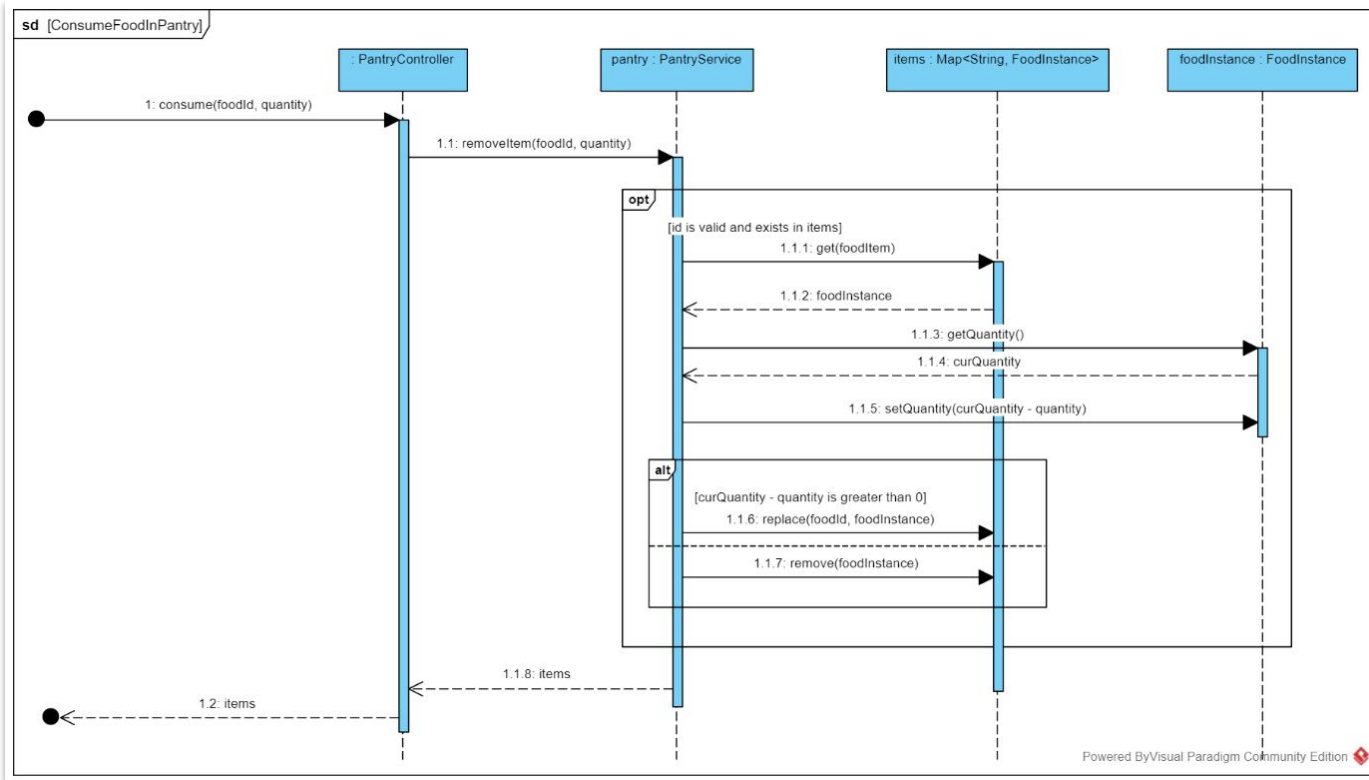
# Sequence Diagrams





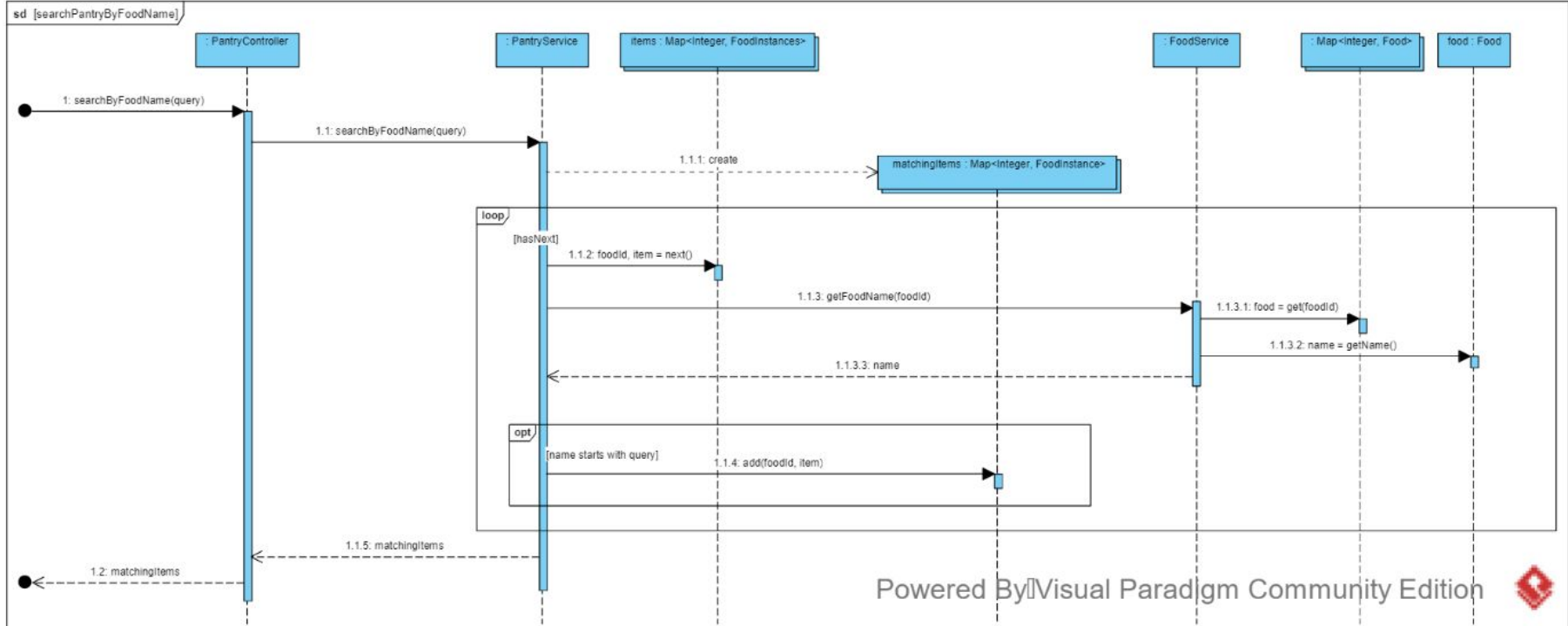
Powered By Visual Paradigm Community Edition

Example Sequence Diagram: Produce Cooked Recipe



Example Sequence Diagram: Consume Food In Pantry





Example Sequence Diagram: Search Pantry

# Test Coverage Plan

Our Test Coverage Plan is designed to prescribe the scope, approach, and high-level overview of all testing activities of the FoodPants project. The plan identifies the items to be tested, the features to be tested, and the types of testing to be performed. Unit testing, integration testing, and system testing will be performed using JUnit to ensure code is implemented correctly and robustly and analyzing the UI to ensure that the interfaces among the subsystems operate correctly and all requirements are met for the user.



# Test Coverage Plan (Cont.)

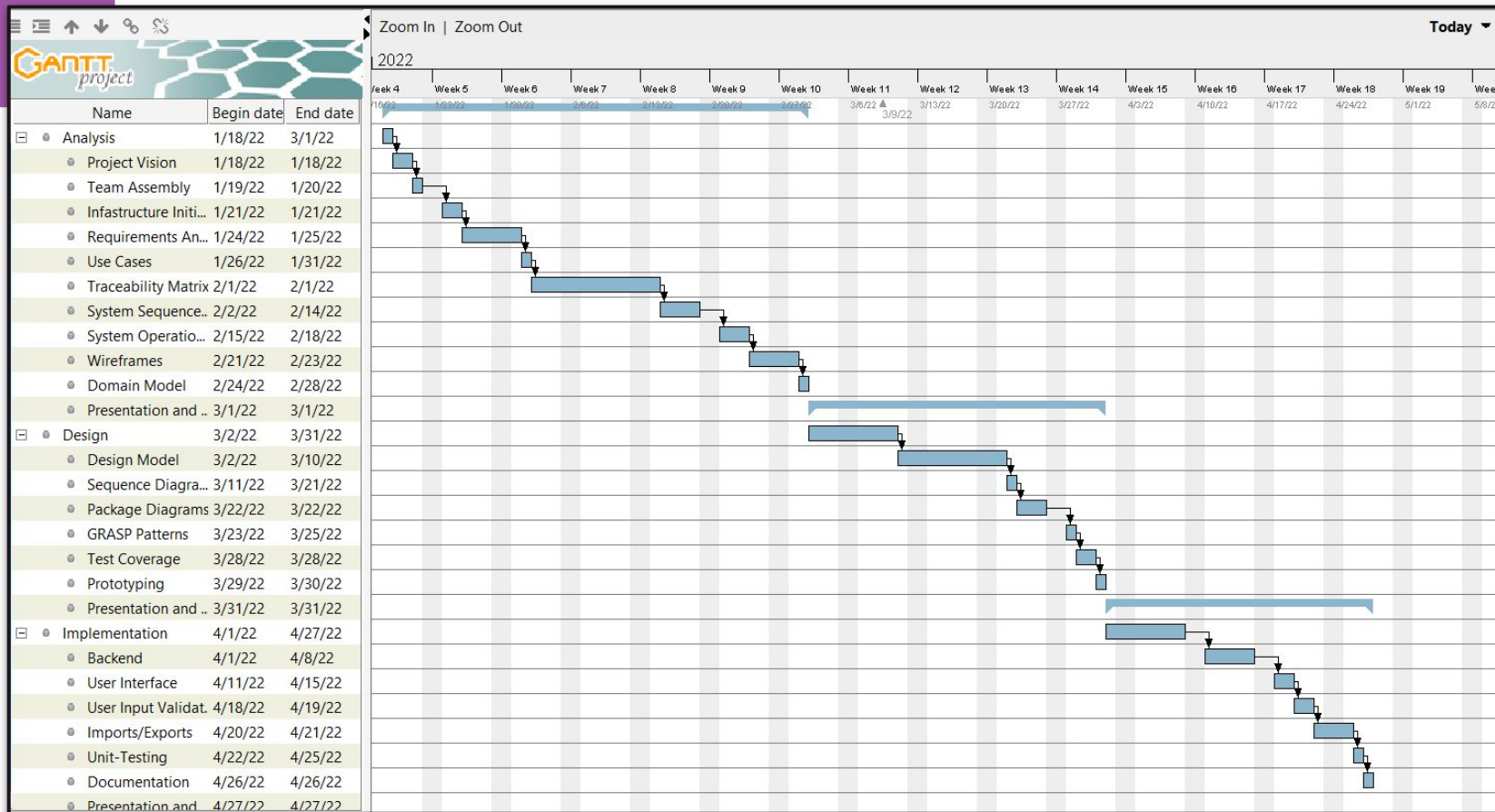
## (1) PANTRY USE CASES (Scope: Pantry system):

- 1.1 Manage pantry (add/edit/remove items manually)
  - Test successful adding of food item (Expected Result: food item should appear in pantry with correct quantity and name)
  - Test successful editing of food item details (Expected Result: food item name or quantity should appear updated to match new entered details)
  - Test successful removal of food items (Expected Result: food item should no longer appear in pantry)
  - Test adding of food item with missing details (Expected Result: program should wait until user enters all missing details or exits without adding)
  - Test adding of food item that already exists (Expected Result: program should add new quantity to already existing quantity of food item that is already in the pantry)
  - Test adding of food item that has new food type (Expected Result: program should prompt use to add a new food type)
- 1.2 Search pantry (by name of food type)
  - Test searching pantry for specific food item (Expected Result: food item with name containing user's query should appear in results)
  - Test searching pantry for query that does not match any food item (Expected Result: program should tell user no match exists)
  - Test searching pantry with query that matches multiple food items (Expected Result: all food items with name containing user's query should appear in results)
- 1.3 Consume specific item (add to nutrition log)
  - Test successful consumption of 1 of food item in pantry (Expected Result: food item should be updated with decremented quantity)
  - Test successful consumption of more than 1 of a food item in pantry (Expected Result: food item should be updated with previous quantity – amount consumed)
  - Test consumption of quantity amount of food item in pantry (Expected Result: quantity left is 0 meaning food item should be removed from pantry)
  - Test consumption of more than quantity amount of food item in pantry (Expected Result: negative quantity left meaning food item should be removed from pantry)





# Updated Gantt Project Plan



# 160+ hours

Tracked via Chronos in Trello



Users ▾		Σ	09 Wed	10 Thu	11 Fri	12 Sat	13 Sun	14 Mon	15 Tue	16 Wed	17 Thu	18 Fri	19 Sat	20 Sun	21 Mon	22 Tue	23 Wed	24 Thu	25 Fri	26 Sat	27 Sun	28 Mon	29 Tue	30 Wed
Austin_Huizinga1		36h 34m							4h 0m						1h 0m							4h 20m		8h 30m
Daniel Luper		33h 8m				1h 15m								21m						1h 20m			6h 50m	11h 3m
Kurt_Wokoek1		22h 34m																		1h 0m	48m	1h 30m	2h 15m	3h 49m
Patrick_Harris3		25h 30m																		2h 0m			2h 30m	7h 30m
PJ_Wallace1		25h 0m															1h 30m			1h 30m		2h 0m	3h 30m	
Luka_Lelovic1		25h 0m														4h 0m				1h 30m	2h 30m	1h 15m	2h 30m	
			0m	0m	0m	1h 15m	0m	0m	4h 0m	0m	0m	0m	0m	21m	1h 0m	4h 0m	1h 30m	0m	0m	7h 20m	3h 18m	9h 5m	17h 35m	30h 51m

## Overview

1 Active pull request

8 Active issues

1

Merged pull request

0

Open pull requests

1

Closed issue

7

New issues

Excluding merges, **6 authors** have pushed **65 commits** to main and **65 commits** to all branches. On main, **73 files** have changed and there have been **1,425 additions** and **2 deletions**.



patrickeharris Add test coverage plan

568f2b5 2 minutes ago 129 commits



FoodPantsApp

Final DEMO UI for presentaiton :D

6 minutes ago



Iteration1

add return numPurchased

14 minutes ago



Iteration2

Add test coverage plan

2 minutes ago



.gitignore

Remove .idea

9 days ago



README.md

Update README.md

12 hours ago

# Git Insights








# Issue Tracking System


☐ 7 Open ✓ 5 Closed


Author ▾ Label ▾ Projects ▾


☐  **System LookAndFeel within swing interferes with images due to improper scaling on HighDPI resolutions** bug  
#13 opened 14 seconds ago by austin127


☐  **Document Issues/Communication**  
#12 opened yesterday by kfwokoek

☐  **Update SD's**  
#11 opened yesterday by kfwokoek

☐  **Consistency** help wanted  
#10 opened 2 days ago by austin127

☐  **Iteration 2**  
#9 opened 8 days ago by lukalelovic

☐  **Page Forms** enhancement  
#8 opened 8 days ago by lukalelovic

☐  **Controllers** enhancement help wanted  
#7 opened 8 days ago by lukalelovic



Q/A

... ? ? ? ? .. ?

# Credits

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by [SlidesCarnival](#)
- Photographs by [Unsplash](#)

