**Project Overview**

**Nicole Burns**

# Overview

As a relatively disorganized person, it can be challenging to keep track of what foods I can make with which ingredients. As such, I needed a program that could store some of my favorite recipes and allow me to search through them to see what recipes I could make with the ingredients I have on hand (i.e. without having to go to the grocery store).

This Recipe Management Software has been developed as a solution to existing recipe management needs. At a high level, the following are in scope for this project:

* Storage of recipes
* Querying of recipes
* Addition of new recipes to storage
* UI to enable ease of use for non-technical people

Anything not explicitly called out in the requirements section of this document is considered out of scope.

# Glossary

* RM = Recipe Management Software, the software being developed as part of this effort.
* User = a user of the recipe management software

# Requirements

|  |  |
| --- | --- |
| Req. ID | Requirement |
| 1 | The RM shall be able to store a recipe’s name |
| 2 | The RM shall be able to store a recipe’s ingredients |
| 3 | The RM shall be able to store a recipe’s directions |
| 4 | The RM shall be able to store a recipe’s total cook time |
| 5 | The RM shall be able to retrieve a filtered list of recipes based on ingredients in a query list |
| 6 | The RM shall be able to retrieve a filtered list of recipes based on a recipe name |
| 7 | The RM shall be able to retrieve a filtered list of recipes based on both a recipe name and partial ingredient list |
| 8 | The RM shall provide the user with the ability to add new recipes |
| 9 | The RM shall not allow a user to create a new recipe without a name |
| 10 | The RM shall not allow a user to create a new recipe without ingredients |
| 11 | The RM shall not allow a user to create a new recipe without the total cook time |
| 12 | The RM shall not allow a user to create a new recipe without directions |
| 13 | Recipe names should be descriptive; the RM shall not allow a user to create a recipe if it has the exact same name as an existing recipe |
| 14 | For a given recipe, the RM shall provide the user with a list of all ingredients |
| 15 | For a given recipe, the RM shall provide the user with all associated directions |
| 16 | For a given recipe, the RM shall provide the user with the total cook time |
| 17 | The RM shall include a UI for users to interact with |

# Tests

There are a handful of test classes set up: TestRecipes.java, TestRecipeSearch.java, and TestRecipeController.java – one for each of the Domain Layer classes. All methods related to a single execution of the program are tested in the test classes, however, methods related to reading and writing from the recipes2.txt file are not tested as any tests that store new recipe data would become invalid each time the test is run.

The UI Layer classes are not tested through Junit testing. Rather testing of the UI Layer was done through an improvised and informal UAT where I opened up the program and tried to break things. A handful of controls were added in as a result of the informal UAT of the Application UI; these controls are discussed in more depth in the presentation.

# Project Disclaimer

Initial project design was based on code taken from here: <https://gist.github.com/cuneyt76/29e9b502aaa5e7ea5adbdffe11c100f0>

While the code from that site was used as a starting point, it was significantly altered to accommodate the requirements outlined above. This is discussed further in the recorded presentation. Specific points of overlap are the Recipe and RecipeSearch classes. Of those, Recipe.java is simple and very similar to the code present in the link above. The RecipeSearch.java class has been entirely redone (new methods, new structure, etc.) to fit what I wanted to accomplish in the project.