Specification for Recipe Roundup (working title), a recipe lookup utility
CpSc 3720: OO Software Development
Chris Ragan
July 11, 2014

Motivation:

Many times I will find myself in the kitchen with no idea of what I should cook. I simply know what ingredients I have, and that I'd rather not spend money on eating out. The solution: Recipe Roundup.

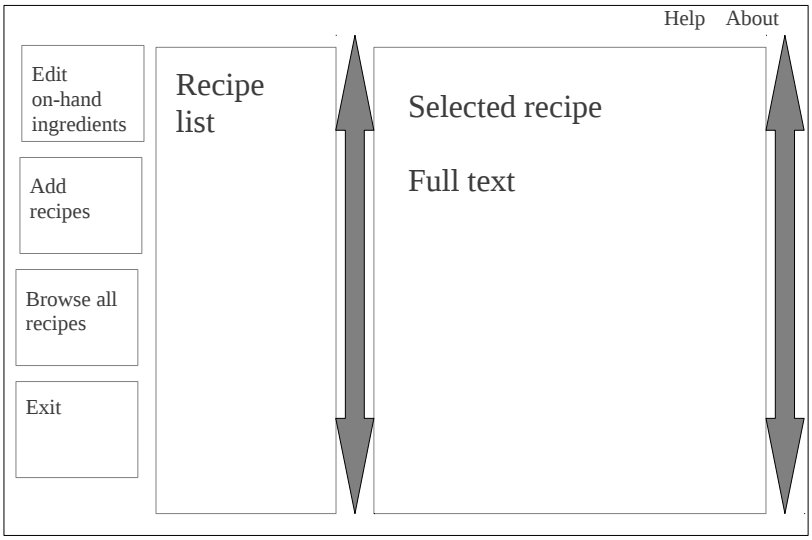


Figure 1: Main window for Recipe Roundup

Main Window:

1. The main window will look similar to Figure 1. Navigation buttons may be replaced with tabbed navigation.
2. Initially, the list and full text will be empty. The user will be required to create a list of on-hand ingredients in order for recipes to be displayed.
3. Loading and saving of the recipe database, as well as the on-hand ingredient list, will be done upon startup and shutdown of the program.

Recipe Roundup start up:

1. No config options are planned at this time. The user will simply type rroundup or something similar.
2. The view will be as shown in Figure 1 when a list of ingredients has already been saved.
3. For a first-time run, the dialog shown in Figure 2 will automatically be shown, prompting the user to select ingredients.

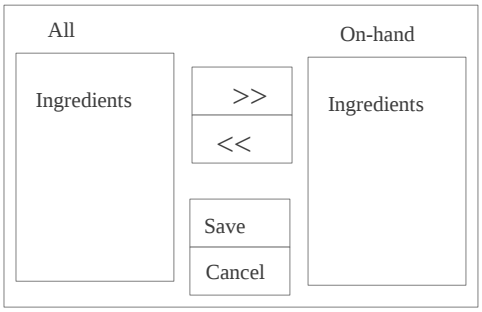

Figure 2: Ingredient selection dialog

Overall functionality:

1. The ingredients available for user selection will be pulled from all the ingredients in the recipe database. This will be updated each time the user enters a new recipe to the database, appending new ingredients not already contained in the list.
2. Once the user has selected on-hand ingredients, all recipes that are possible will be displayed in the recipe list.
3. The full text of the recipe will be displayed in the text area adjacent to the recipe list. This will be updated each time the user selects a different recipe.
4. Functionality will be built-in for the user to browse ALL recipes in the database, regardless of whether they have the ingredients.
5. Recipe entry process is TBD. There will be a database of at least 10 recipes provided with the application, and functionality for additional entries will likely be added after the initial release.

Deliverables:

1. Core functionality: The initial startup window, loading and saving data to an .xml file, and the ingredient selection dialog. This will provide the framework upon which everything else will hinge. MVC is the design pattern I intend to use most heavily, but will revise as the project progresses.
2. Intermediate: At this point search functionality should be possible based on the list of ingredients entered. Recipe format will be standardized to allow ease of expansion. This stage will showcase the actual recipe lookup.
3. Final: At least 10 recipes will be available for search; more as time permits. Lookup by recipe and full browsing capability will be implemented, and new recipe entry functionality should be at least in the prototype stage (may be rough or incomplete). A tabbed window would be ideal to allow rapid switching between functions, but this is a lower-priority item.

Difficult problems to solve:

1. Making the search process efficient and scalable as the recipe database grows.
2. Implementing tabbed windows rather than navigation buttons.
3. Ensuring similar ingredients don't wind up in the list (e.g. “Egg”, “Eggs”, “Egg whites”).
4. Figuring out a simple process for entering new recipes. The app won't get used if expansion is overly cumbersome.