

Recipe Manager (0.1.1) Manual

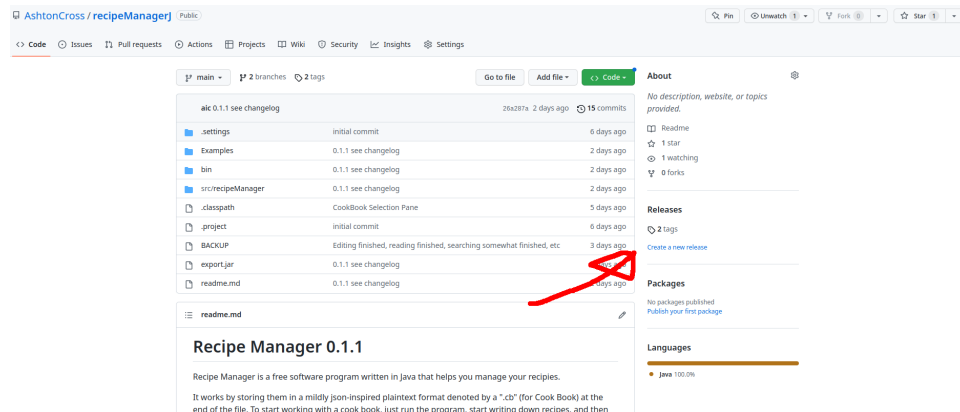
1 Introduction

Recipe Manager is a program written in Java to help you manage your recipes. Recipes are stored within files that end in “.cb”. These files are read and written by the Recipe Manager. To read more about Cookbook files and recipes, see section 2 (Cookbook Files and Recipes).

This guide is 1/4 school project requirement, 1/4 me having fun writing a label for a program I am proud of, 1/4 me getting practice with L^AT_EX, and 1/4 information about how to use the Recipe Manager program.

1.1 Downloading, “Installing”, and Launching

Recipe Manager .jar’s are available for download at the recipe manager github page under the “releases” tab.



After downloading the latest release, unzip the archive and move it to a reasonable location, such as in your Documents folder. Before running the program, make sure that you have the correct version of Java installed. Recipe Manager requires any version above Java 17. After you have the proper Java Runtime Environment installed, you may now run the program.

To run the program, you now need to open a terminal and run the following command:

```
java -jar Recipe-Manager.jar
```

Certain systems may also allow you to double click the .jar file in a file browser. If this is not the case with your operating system, then I recommend creating a run script. On Unix systems such as GNU/Linux, this script should look something like this:

Listing 1: run.sh

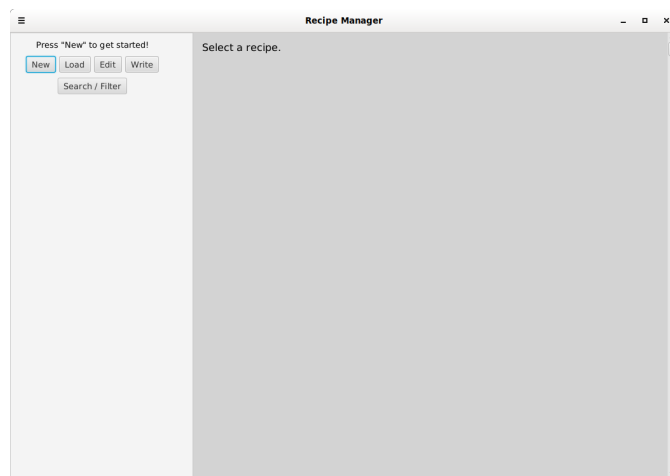
```
1  #!/usr/bin/sh
2
3  java -jar /absolute/path/to/Recipe-Manager.jar
```

After creating, give the file run permissions with the command “`chmod +x run.sh`”, and then run by entering `./run.sh` into the terminal. (`sh run.sh` works fine as well, and does not require executable permissions). This script should also function on MacOSX or BSD, as well as most other Unix operating systems with Java installed.

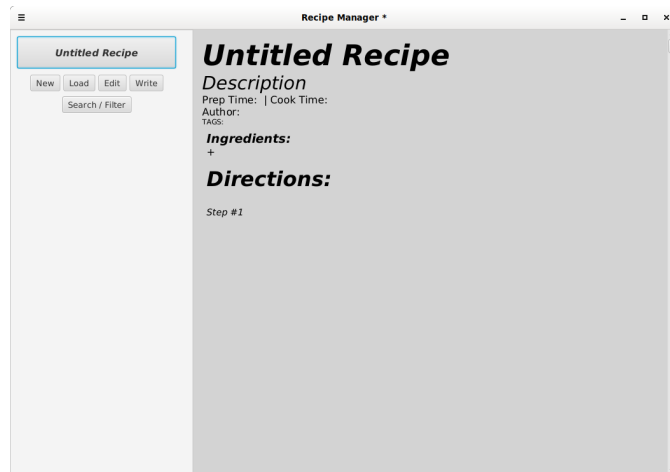
For Windows, you can use this exact same execution script (minus the shebang at the first line) stored within a .bat file, so long as you have your Java Runtime Environment in your PATH environment variable.

1.2 Creating Your First Recipe

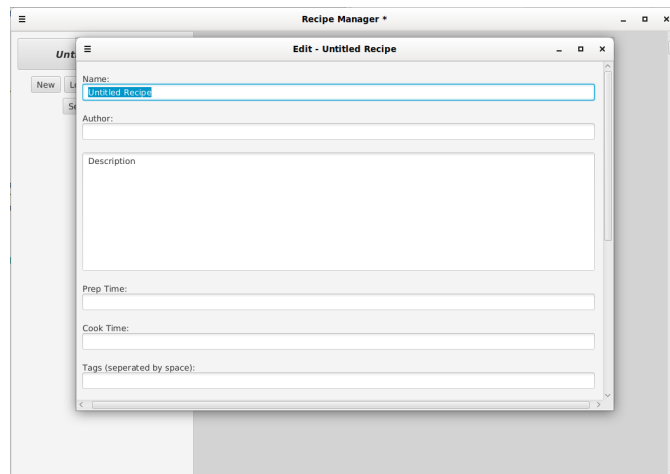
When you first launch Recipe Manager, the screen should look something like this.



To create an empty recipe, click the “New” button. A new recipe will appear on the Cookbook Select panel on the left. Click onto the recipe to load it into the Information panel.



You should notice that the currently selected recipe will change its name to bold and italic. With this recipe selected, you can now hit the button labeled “Edit” to bring open the Recipe Editor.



This panel is where you edit a recipe. Go ahead and modify the name, author, and description. Tags allow you to organize your recipies, for more information, see section 2.2. Incredients and directions can be added and removed using the “+” and “-” buttons respectively, and then once you are done, you can click on the “Save Without Writing” button. As the button implies, this will save the recipe so that the edits are viewable from within the information panel. You may now click on the “Write” button, and if you have not written to a file yet, then it will prompt you for a location and name for the file to be stored in.

Congratulations! You have now created a brand new recipe!

1.3 Loading Recipes

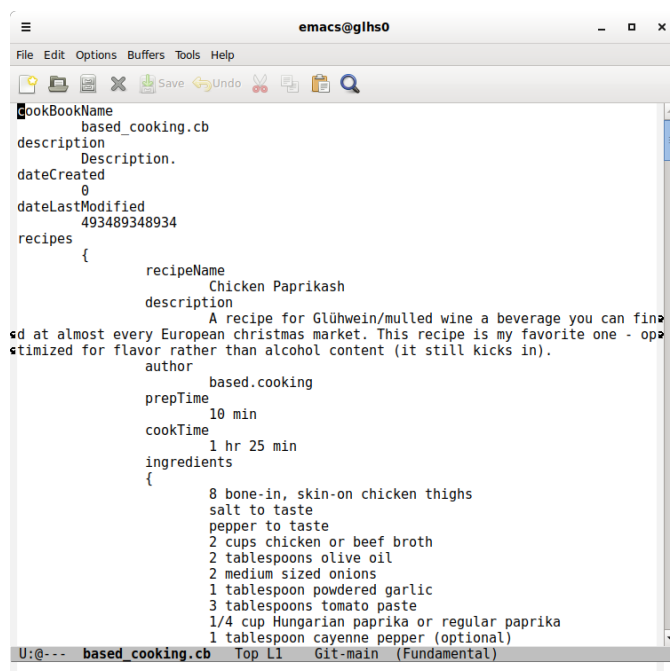
To load a pre-existing recipe, click the “Load” button on the Control Panel. The program will prompt you for a location of a *.cb file. After navigating to a file and selecting it, the file will then be loaded.

1.4 Deleting Recipes

To delete a recipe, select the recipe that you want to delete, and then enter the Editor by hitting the “Edit” button on the Control Panel. After entering the Editor, scroll down to the bottom of the Editor, and click the “Delete” button. The program will ask you if you are sure, and after confirmation, the recipe will be removed from the list. This recipe will not be written the next time that the “Write” button is invoked.

2 Cookbook Files and Recipes

Cookbook files, denoted by the file postfix “.cb”. These files are plaintext, and are editable via any text-editor.



```
emacsglho
File Edit Options Buffers Tools Help
[Icons] Save Undo [Icons]
{
  cookBookName
    based_cooking.cb
  description
    Description.
  dateCreated
    0
  dateLastModified
    493489348934
  recipes
    {
      recipeName
        Chicken Paprikash
      description
        A recipe for Glühwein/mulled wine a beverage you can find
        ed at almost every European christmas market. This recipe is my favorite one - op
        timized for flavor rather than alcohol content (it still kicks in).
      author
        based.cooking
      prepTime
        10 min
      cookTime
        1 hr 25 min
      ingredients
        {
          8 bone-in, skin-on chicken thighs
          salt to taste
          pepper to taste
          2 cups chicken or beef broth
          2 tablespoons olive oil
          2 medium sized onions
          1 tablespoon powdered garlic
          3 tablespoons tomato paste
          1/4 cup Hungarian paprika or regular paprika
          1 tablespoon cayenne pepper (optional)
        }
    }
}
```

U:~ -- based_cooking.cb Top L1 Git-main (Fundamental)

It is best not to tamper with the data within the plaintext file, as the Recipe Manager interpreter is very sensitive to indentation.

2.1 Recipes

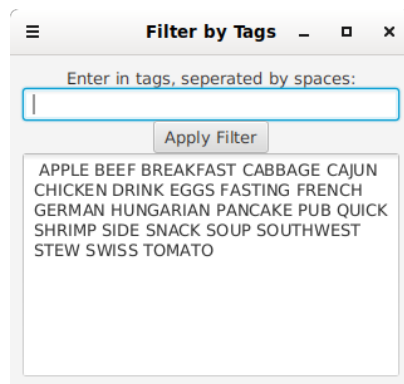
Recipes contain the following information:

- Name of the recipe.
- Author name.
- Description of recipe.
- Prep time.
- Cook time.
- Tags ^{2.2}.
- Ingredients.
- Directions.

The first few elements of the recipe do not have any special formatting and are displayed and stored as-is. The ingredients and directions are expandable, and you may add as many additional directions and ingredients as needed.

2.2 Tags

Tags are used for sorting recipes. Recipes are given tags, and when searching by tags, only recipes with that tag will show up on the Control Panel. To search, simply click the intuitively labeled “Search / Filter” button on the Control Panel. From there the Filter Menu will open up.



The text area showcases every avilable tag currently used by recipes currently loaded into the frame.

3 Compiling Source

For building, I recommend compiling the project through Eclipse. Start by importing the project into Eclipse, then right click on the project in the Package Explorer, and click “Export”. Continue on with all the default values, then it should create a jar file called “Export.jar” located in the project files.

Run this file using the instructions in section 1.1.