Refactorings and Addressing Bug Reports Document Guide

Perfect Pantry App
Group 7, Section Z

Refactorings

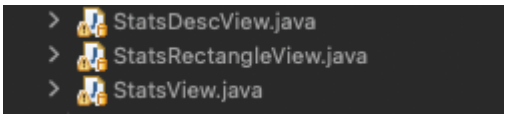
1. StatsView

Problem/ Code Smell: Repetitive Code

The stats view consists of two bar charts which indicate the composition of items in all containers, and within specific containers. When I began to design this user story, I created the UI directly in the StatsView class initially to display the composition of all containers. However, this meant that to display the composition of a specific container, I would need to duplicate my original code but with slightly different values.

Thus I utilised the ***Extract Class method*** to divide the StatsView into StatsRectangle and StatsDesc. So now new bar charts can be created for each container as well as for all pantries, whilst utilising the same code.

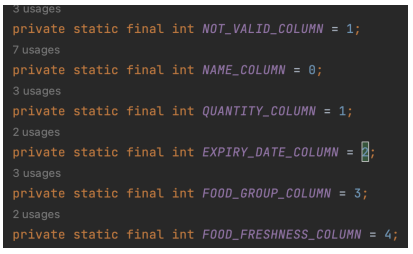
StatsView is the main view, StatsRectangleView is the bar chart and StatsDesc are the labels/ descriptions of the bar chart.



2. ItemListView - Nina

Problem/ Code Smell: Magic Numbers

There were many numeric literals used as column indexes without any contexts. At the start of this user story, the order of columns from item name to their freshness tag was not decided. When adding more features to this class, sometimes I forget which column index is responsible for which item field (name, expiry date, etc).



I replace those column index literals with a **Symbolic Constant.** It now adds meaning and context to those values.

In a later refactor (**4. ItemsListView),** I moved these constants into a separate class, CustomTableModel, as those fields were more relevant to that class. CustomTableModel determines the cells under which columns are editable and the class of objects within those columns.

3. StarredRecipeListView - Nina

Problem/ Code Smell: Speculative Generality

The deleteRecipe method has an unused JPanel recipePanel parameter. It was initially intended for putting the delete confirmation message but that was never implemented. Instead, it just uses JOptionPane with HomeView.getFrame() as its parent component.

```
private void deleteRecipe(Recipe recipe, JPanel recipePanel) {
    int opt = JOptionPane.showConfirmDialog(HomeView.getFrame(), message: "D
    if (opt == JOptionPane.YES_OPTION) {
        HomeView.data.removeStarredRecipe(recipe);
        displayRecipes();
    }
}
```

Simply do **Remove Parameter** to remove unnecessary code. There was a chain of unneeded parameters and they all were removed.

4. ItemsListView - Nina
Problem/ Code Smell: Large class
The ItemsListView class has too many responsibilities. ItemsListView deals with table of items, right click feature for JMenuItem (edit quantity, remove item, storage tip), colour coding items, and filter/sort functionality. It is tough to add features to them or maintain them.

I used the **Extract Class** technique by making 3 new classes (ActionListenerManager, CustomTableModel and CustomColorCodedTable) that contain relevant fields and methods. The constructor and the ItemsListView class itself are now less bloated. You can check what it looked like before here: [ItemsListView Before]
ItemsListView's constructor is simplified to these following pictures.

```
tableModel = new CustomTableModel();
table = new CustomColorCodedTable(tableModel);
```

```
ActionListenerManager actionListenerManager = new ActionListenerManag

actionListenerManager.attachRemoveItemListener(removeItem);
actionListenerManager.attachGenerateTipListener(generateTip);
actionListenerManager.attachEditQtyListener(editQty);
actionListenerManager.attachCustomTagListener(customTag);
```

You can check them out here:
[ActionListenerManager]
[CustomTableModel]
[CustomColorCodedTable]
[ItemsListView]

SettingsView - Sarah
Problem: Large Methods
The method setSettingsViewVisibility was too large with different aspects setting up gui. Hence I separated smaller tasks such as addFonts and addActionListeners to its own method.

Bug Report Addressings

Explain how you addressed each reported issue.

1. ItemsListView - Nina
Bug: Colour Code for FoodGroups is not shown
https://github.com/EECS2311/EECS2311/issues/121

I included colour code for FoodGroups by allowing the toggle colour button to switch colour coding mode Enums. The prepareRenderer has a switch statement for each mode and colours the row entries by their respective value.

2. domain.logic.recipe - Nina
Bug: Recipe ingredients mismatch
https://github.com/EECS2311/EECS2311/issues/118
https://github.com/EECS2311/EECS2311/issues/119
The available and missing ingredients may not properly reflect the food items in the containers. I found out that this is the limitation of Spoonacular API: https://spoonacular.com/food-api/docs#Search-Recipes-by-Ingredients

It gets recipes and their needed ingredients from website/blog sources. The available and missing ingredients are determined by the API function itself. It would be available if the ingredients mentioned in the API query are related to the recipe's ingredients. It is also why it guesses "goat cheese" from "cheese" like in “Bug Report: Recipe ingredients list mismatch #118”. Sometimes, some of the available ingredients should be in missing ingredients and vice versa. But this ingredient sensitivity is not in my control. The recipe page would be treated as recommendations based on the provided ingredients (and it won't guarantee that it will exactly match the food items you have or don't have).

3. ItemListView - Nina
Bug: Table columns are moveable
https://github.com/EECS2311/EECS2311/issues/112
The columns should not be movable and I followed the suggested fix by adding table.getTableHeader().setReorderingAllowed(false); in the ItemListView constructor.

4. DB - Nina
Bug: Deleting an item removes any items with matching name in database
https://github.com/EECS2311/EECS2311/issues/112
Deleting an item in some container should not delete other items with a matching name in different containers. Following the suggested fix, I changed the query to account for container so deleting an item needs to both match the requested item name and container name.

5. AddItemView - Nina
Bug(s): Expired Items can be added + Incorrect error messages for AddItemView
https://github.com/EECS2311/EECS2311/issues/107
https://github.com/EECS2311/EECS2311/issues/106
Already expired items should not be added to containers and there were a few incorrect error messages to the user when adding an invalid item. For both of these bugs, I updated the verifyAddItem method logic in ItemUtility class. When there are multiple invalid or empty fields, it gives a generic error message “There are empty and/or invalid inputs, no item added." It now gives separate error messages when item name is empty or item name is over 50 characters. It also checks if the quantity is an integer (and is at least 1), if expiry date is entered in a correct format (dd-MMM-yyyy), and if the expiry date has already passed via a helper function validateExpiryDate.

6. DB - Edison
Bug report: Lowercase food item does not show storage tip
https://github.com/EECS2311/EECS2311/issues/122
Storage tips were case-sensitive and would not show up unless the item followed a predesigned case. In the getStorageTip method, I altered the input string to follow a format where the first character was in upper-case and the rest was in lower-case. Any new storage tip added to the database should follow such format.

7. ItemsListView - Michel
Bug Report(s): Performing operations while filtering or sorting, selects the wrong item
https://github.com/EECS2311/EECS2311/issues/116
https://github.com/EECS2311/EECS2311/issues/115
https://github.com/EECS2311/EECS2311/issues/114
https://github.com/EECS2311/EECS2311/issues/113

Storage tips, and deleting items would trigger on the wrong item on the table if sorting or filtering was in effect. The simple solution was to ensure that the right click menu options would occur on the main table (not the sorted or filtered ones).

8. Calendar - Sarah
https://github.com/EECS2311/EECS2311/issues/117
Bug Report(s): Navigating the calendar - incorrect year display
Pressing “Previous Month” when its February will make the year go back, making the date wrong. The fix to this was the month value should be 0 instead of 1 (0 or Janauary)

9. Settings - Sarah
Pressing “settings” button on the homeview will cause incrementations to increase, and the notifboolean to not work. The fix to this was to remove action listeners before reading them in setSettingsGUI method