CookHelper

Το πρόγραμμα CookHelper παίρνει συνταγές και εμφανίζει χρήσιμες πληροφορίες για αυτές.

User manual:

Μπορείτε να τρέξετε την εντολή ‘java -jar target/recipes.jar -help’ για να δείτε το user manual.

```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -help
User manual:
  This program only accepts .cook files.

    Mode 1 (GUI):
      Usage 1 (Empty): Opens GUI with no pre-selected files.
        Run: java -jar target/recipes.jar
                        or
             java -jar target/recipes.jar -gui

      Usage 2 (With files): Opens GUI with one or more pre-selected files.
        Run: java -jar target/recipes.jar -gui recipe1.cook recipe2.cook ... recipeN.cook

    Mode 2 (Terminal):
      Usage 1 (Display Recipe): Select one recipe and display it.
        For one person run: java -jar -term target/recipes.jar recipe.cook
        For multiple people (e.g 3) run: java -jar -term target/recipes.jar recipe.cook 3

      Usage 2 (Shopping list): Select one or more recipes and print a shopping list.
        For one person run: java -jar target/recipes -term -list recipe1.cook recipe2.cook ... recipeN.cook
        For multiple people (e.g 3) run: java -jar target/recipes.jar -term -list recipe1.cook recipe2.cook ... recipeN.cook 3

  For further information please check the README and report files.

alex@pc:~/IdeaProjects/CookHelper (main)$
```

GUI

Το GUI έχει 6 κουμπιά που μπορεί να χρησιμοποιήσει ο χρήστης:

Select Recipe: Ανοίγει τα αρχεία του χρήστη για να διαλέξει με mutly-select τα αρχεία που θέλει να φέρει στην λίστα.

Display Recipe: Παίρνει ένα αρχείο και εμφανίζει την συνταγή.

Shopping List: Παίρνει ένα ή πολλά αρχεία και εμφανίζει την λίστα για ψώνια.

Execute Recipe: Παίρνει ένα αρχείο και εκτελεί την συνταγή.

Remove Recipes: Παίρνει ένα ή πολλά αρχεία και αφαιρεί τις συνταγές από την λίστα.

Exit: Τερματίζει το πρόγραμμα.

Προστασία του χρήστη

Το πρόγραμμα προστατεύει τον χρήστη και στο terminal αλλά και στο GUI για να μην τερματίζει το πρόγραμμα ανώμαλα στα λάθος inputs.

Παραδείγματα προστασίας terminal:

GUI Mode:

```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar
Opening GUI
```

```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar abc def
To add arguments you have to use modifiers!

For user manual run: java -jar target/recipes.jar -help
alex@pc:~/IdeaProjects/CookHelper (main)$
```

```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -gui ~/recipes/eggs.cook ~/recipes/eggs.cook /etc/passwd non_existing_file
File 'eggs.cook' is already in the list.
File 'passwd' is not a '.cook' file!
File 'non_existing_file' doesn't exist!
Opening GUI
Selected Files: 'eggs'
```

Terminal Mode:

```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -term ~/recipes/bread.cook ~/recipes/eggs.cook
Too many files!

For user manual run: java -jar target/recipes.jar -help
alex@pc:~/IdeaProjects/CookHelper (main)$
```

```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -term /etc/passwd
File '/etc/passwd' is not a '.cook' file!

For user manual run: java -jar target/recipes.jar -help
alex@pc:~/IdeaProjects/CookHelper (main)$
```

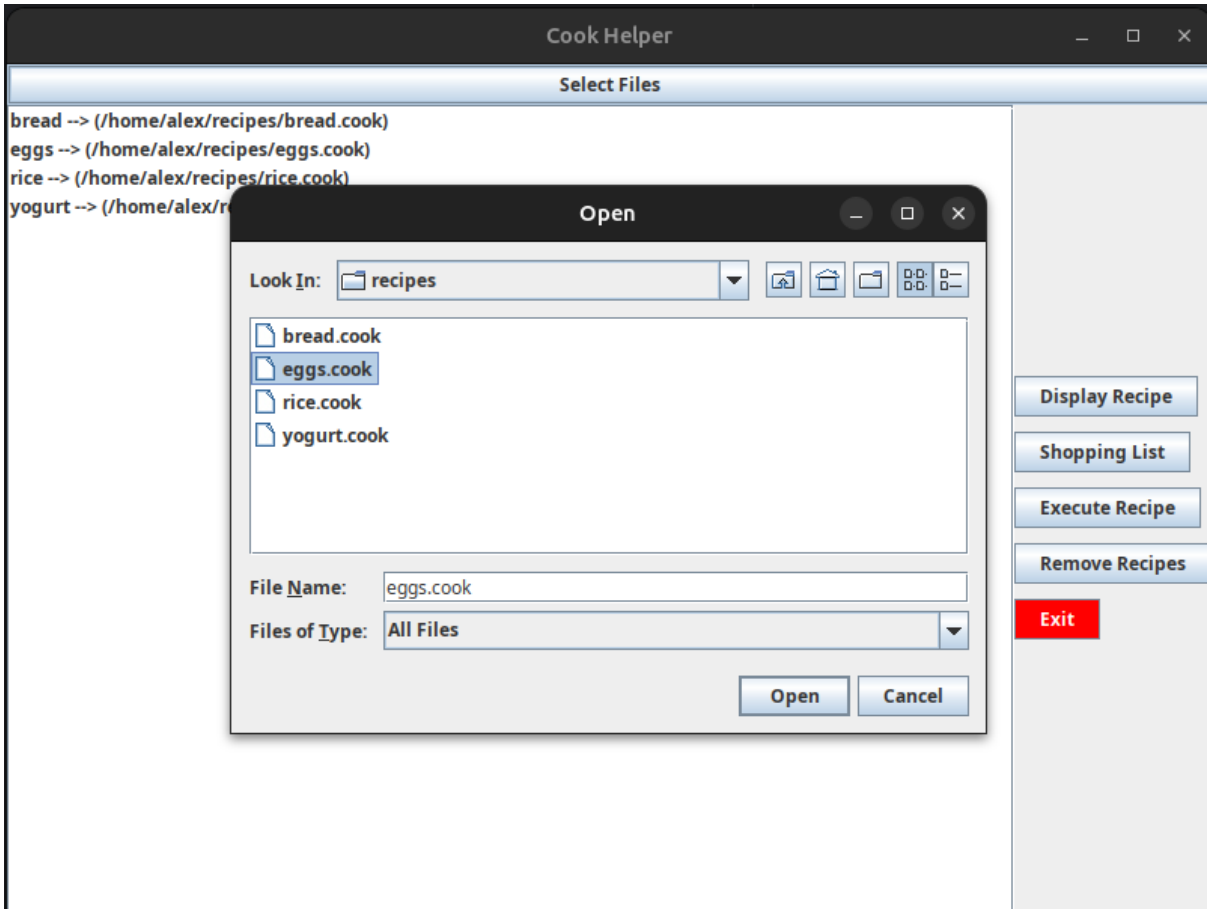
```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -term non_existing_file.cook
File 'non_existing_file.cook' doesn't exist!

For user manual run: java -jar target/recipes.jar -help
alex@pc:~/IdeaProjects/CookHelper (main)$
```

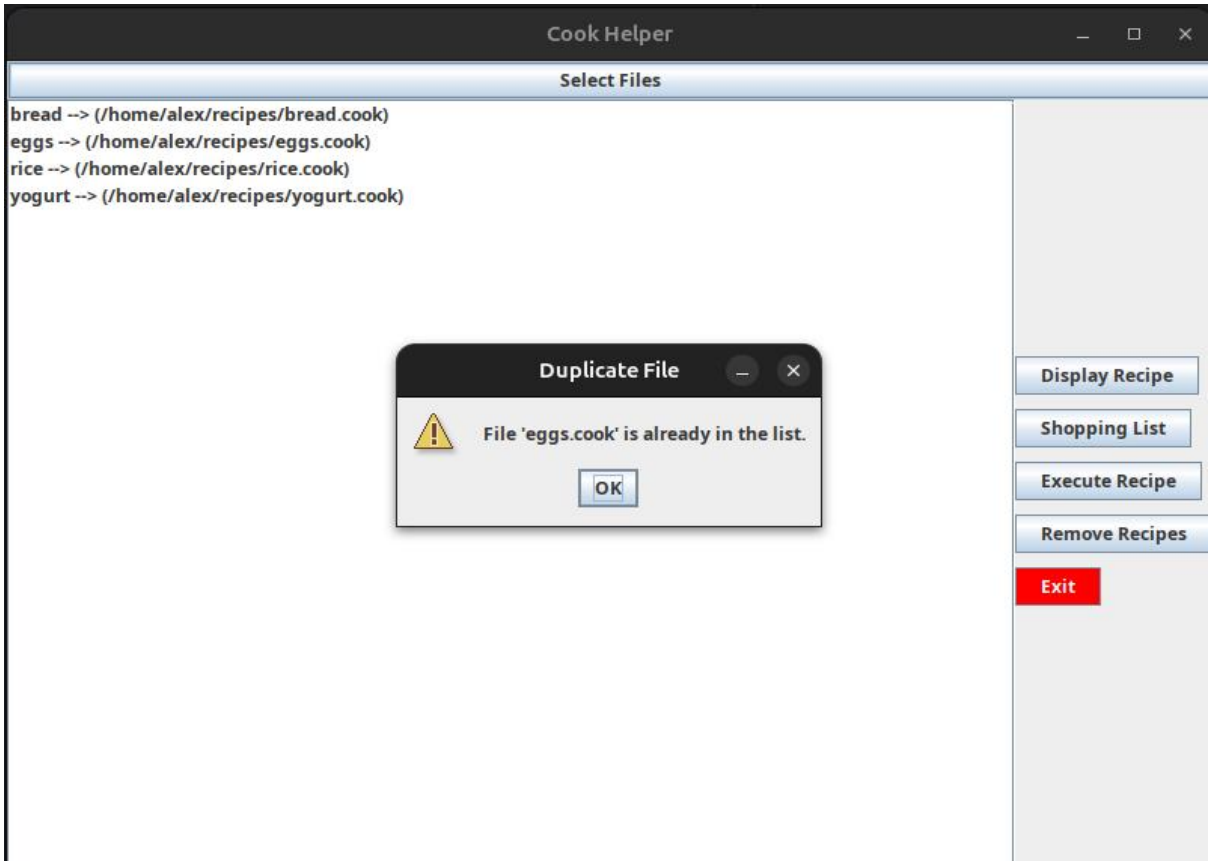
```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -term
No files provided!

For user manual run: java -jar target/recipes.jar -help
alex@pc:~/IdeaProjects/CookHelper (main)$
```

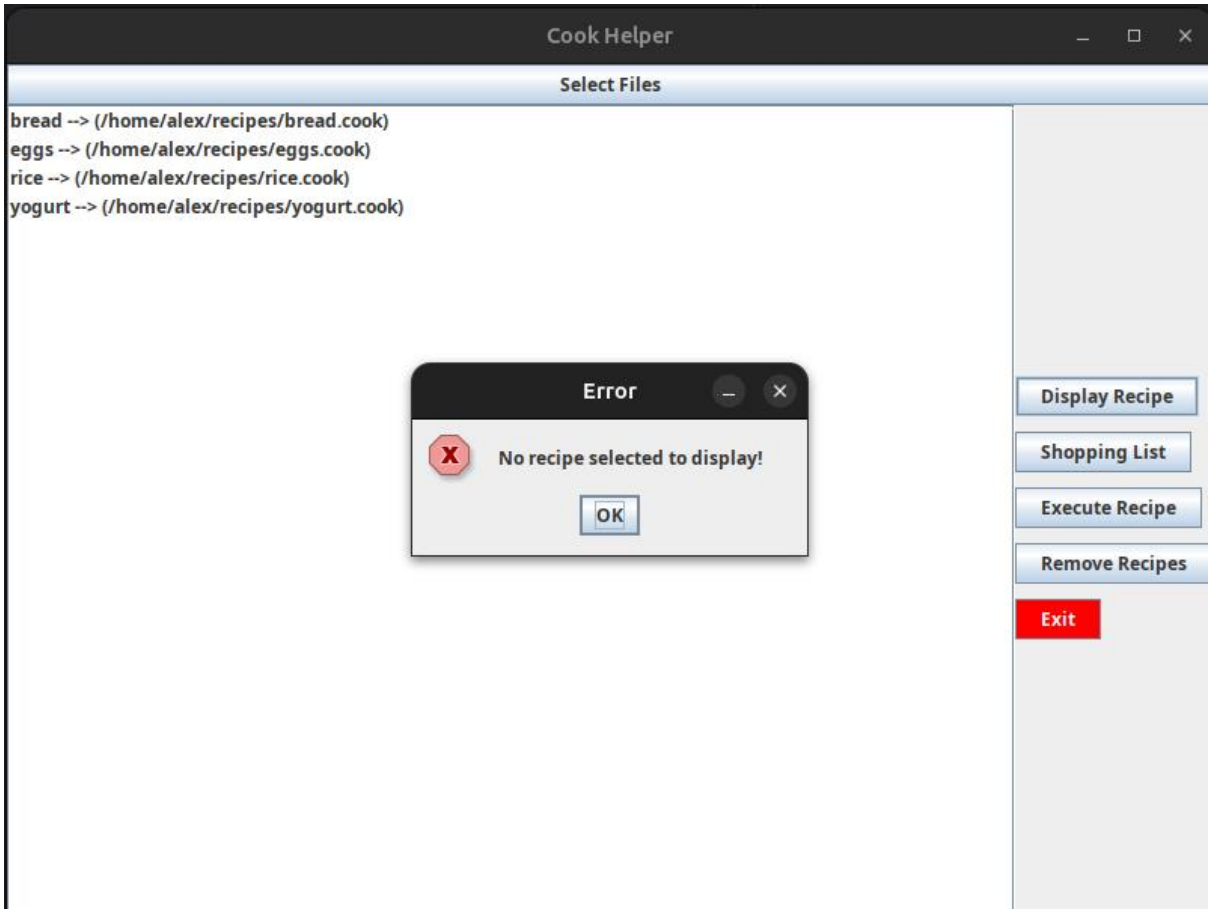
```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -term -list
No files provided!

For user manual run: java -jar target/recipes.jar -help
alex@pc:~/IdeaProjects/CookHelper (main)$
```

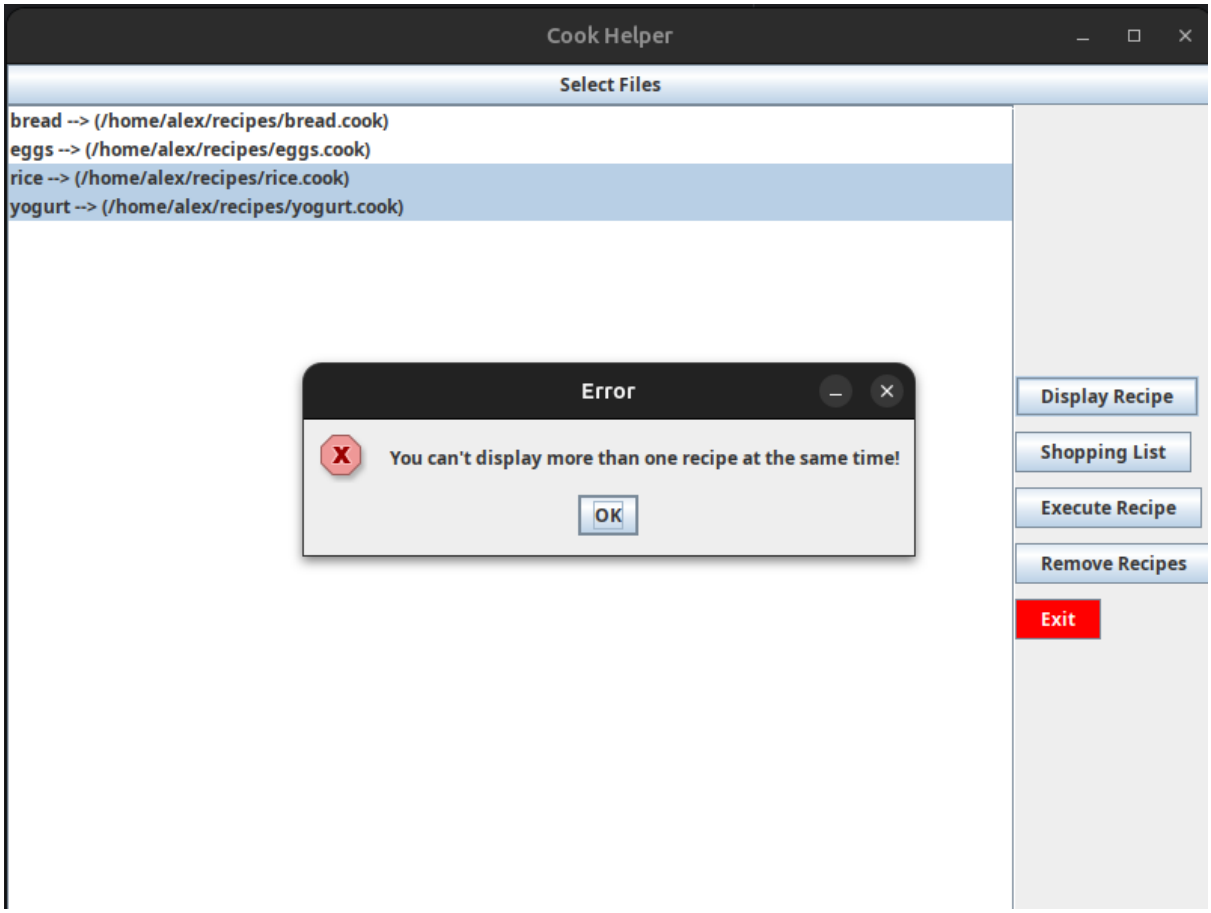
```
alex@pc:~/IdeaProjects/CookHelper (main)$ java -jar target/recipes.jar -term -list ~/recipes/eggs.cook non_existing_file.cook
File 'non_existing_file.cook' doesn't exist!

For user manual run: java -jar target/recipes.jar -help
alex@pc:~/IdeaProjects/CookHelper (main)$
```

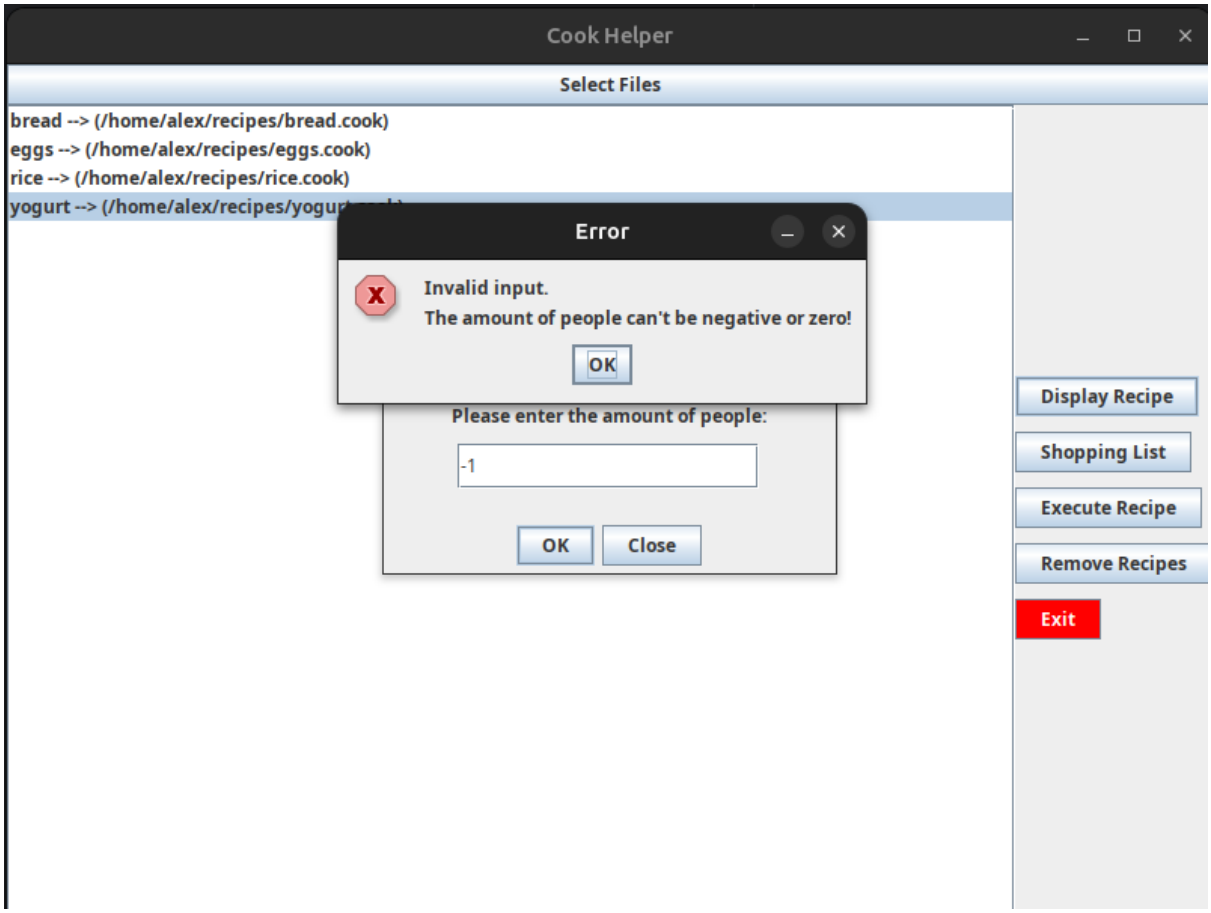
Παραδείγματα προστασίας GUI:

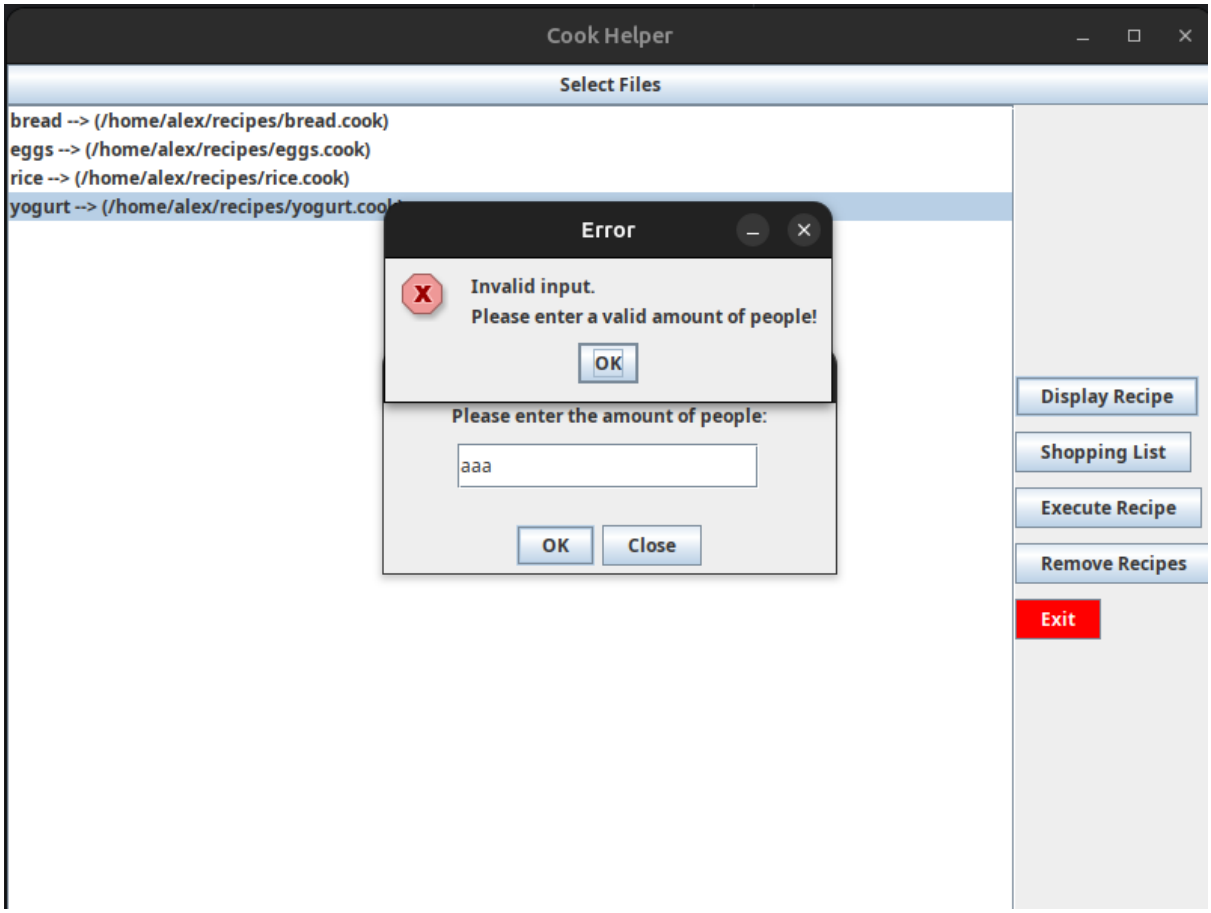



Cook Helper
Select Files
bread --> (/home/alex/recipes/bread.cook)
eggs --> (/home/alex/recipes/eggs.cook)
rice --> (/home/alex/recipes/rice.cook)
yogurt --> (/home/alex/recipes/yogurt.cook)
Duplicate File
File 'eggs.cook' is already in the list.
OK
Display Recipe
Shopping List
Execute Recipe
Remove Recipes
Exit


Cook Helper
Select Files
bread --> (/home/alex/recipes/bread.cook)
eggs --> (/home/alex/recipes/eggs.cook)
rice --> (/home/alex/recipes/rice.cook)
yogurt --> (/home/alex/recipes/yogurt.cook)
Error
No recipe selected to display!
OK
Display Recipe
Shopping List
Execute Recipe
Remove Recipes
Exit


Cook Helper
Select Files
bread --> (/home/alex/recipes/bread.cook)
eggs --> (/home/alex/recipes/eggs.cook)
rice --> (/home/alex/recipes/rice.cook)
yogurt --> (/home/alex/recipes/yogurt.cook)
Error
You can't display more than one recipe at the same time!
OK
Display Recipe
Shopping List
Execute Recipe
Remove Recipes
Exit


Cook Helper
Select Files
bread --> (/home/alex/recipes/bread.cook)
eggs --> (/home/alex/recipes/eggs.cook)
rice --> (/home/alex/recipes/rice.cook)
Error
Invalid input.
The amount of people can't be negative or zero!
OK
Please enter the amount of people:
-1
OK
Close
Display Recipe
Shopping List
Execute Recipe
Remove Recipes
Exit



Τρόπος υλοποίησης GUI

Το μόνο σημαντικό που αξίζει να αναφέρουμε για το GUI είναι ότι στις λειτουργίες Display Recipe, Shopping List και Execute Recipe χρησιμοποιούμε τις κλάσεις DisplayRecipeModel, ShoppingListModel και ExecuteRecipeModel που αποφασίζουν τι πρέπει να εμφανιστεί κάθε φορά και χρησιμοποιούμε τις κλάσεις DisplayRecipeView, ShoppingListView και ExecuteRecipeView που καθορίζουν το πως θα εμφανιστούν τα δεδομένα που υπάρχουν στο αντίστοιχο model.

Τρόπος υλοποίησης κώδικα

Αρχικά ελέγχει αν τα arguments είναι αποδεκτά αλλιώς εμφανίζει κατάλληλο μήνυμα. Ανάλογα τα arguments χρησιμοποιείτε διαφορετική λειτουργία.

Ο τρόπος που διαβάζονται τα αρχεία είναι μέσω της κλάσης StepsReaderImpl (υποκλάση της διεπαφής StepsReader) που με την βοήθεια της BufferedReader διαβάζει από το αρχείο τα βήματα και κάνει την κατάλληλη επεξεργασία για ξεχωρίσει τα ingredients, τα utensils και το time. Τα στοιχεία αυτά τα παίρνει η κλάση RecipeImpl (υποκλάση της διεπαφής Recipe) και εμφανίζει τα ζητούμενα της εκφώνησης σε αυτήν την συνταγή.

Η λίστα δεν είναι τίποτα παραπάνω από την προηγούμενη εξήγηση με την διαφορά ότι αυτή η διαδικασία γίνετε όσες φορές όσες με τις συνταγές που δόθηκαν και τις αποθηκεύει σε μία λίστα. Η λίστα εμφανίζει μόνο τα υλικά των συνταγών.

Και στις 2 λειτουργίες ο χρήστης μπορεί να δώσει στο τέλος των arguments έναν ακέραιο αριθμό που συμβολίζει τον αριθμό ατόμων οπότε η μόνη διαφορά είναι ότι η ποσότητα των υλικών πολλαπλασιάζεται με τον αριθμό των ατόμων. Αν δεν δοθεί αριθμός ατόμων τότε υποθέτουμε ότι ο αριθμός ατόμων είναι 1.

Ο τρόπος που αποθηκεύουμε τα υλικά σε μία συνταγή είναι με την κλάση IngredientImpl (υποκλάση της διεπαφής Ingredient) με name, amount και measure που αποθηκεύονται σε ένα HashMap με value ένα String που είναι το όνομα του υλικού και σαν value ένα αντικείμενο τύπου IngredientByMeasure που έχει μέσα ένα άλλο HashMap για να κρατάει τα διαφορετικά measures για το ίδιο υλικό.

Ο τρόπος που αποθηκεύουμε τα σκεύη σε μία συνταγή είναι με την κλάση UtensilImpl (υποκλάση της διεπαφής Utensil) με name που αποθηκεύονται σε ένα HashMap.

Ο τρόπος που αποθηκεύουμε τον χρόνο σε μία συνταγή είναι με την κλάση TimeImpl (υποκλάση της διεπαφής Time) με time (ο χρόνος σε δευτερόλεπτα) και αποθηκεύονται σε μια μεταβλητή τύπου Time.