TAYLOR'S UNIVERSITY

Wisdom • Integrity • Excellence

ITS61004 Object-Oriented Programming Using Java

ITS63704 Advanced Object–Oriented Programming

ITS66704 Advanced Programming

Project

Cover Sheet

DUE DATE : Week 14 (Presentation Slot)

WEIGHTAGE: 30%

SEMESTER : March 2019

We declare that:

- ***We understand what is meant by plagiarism***
- ***The implication of plagiarism has been explained to us by our lecturer***
- ***This assignment is our own work.***

No	Student Name	Student ID	Contribution**	Overall Score	Your Score
1	Ee Sun Jun	0332103	100%		
2	Sarah Chong Giet Yee	0336289	100%		
3	Ong Sze Yun	0332474	100%		
4	Tan Xuan Ni	0332107	100%		
5	Timothy Louis Scott	0332841	100%		

** Evaluate by your group members of your member contribution to the assignment, 0% - 100%

* Minimum 3 members and maximum 4 members * Your score = Overall Score * Contribution = 4 * 90% = 3.6

Marking Rubrics (Lecturer's Use Only) – *attached as 2nd page in the report*

Criteria	Score				
	Excellent 10	Good 7.5 – 9.5	Fair 5 – 7	Marginal 2.5 – 4.5	Inadequate 0 – 2
Object Oriented Programming Design & Principles					
1. Exception Handling					
2. IO Stream					
3. JavaFX					
4. Event Driven Handlers					
5. User Interface Controls					
6. User Interface Design (Feel & Look)					
Demo & Team Works					
7. Ability to answer questions					
8. Team Collaboration					
Proposal & Coding & Documentation					
9. Code Solution & application Logic					
10. Program Documentation (UML Diagram & Coding & Screen Shots)					
Sub-Total					
Additional Features (Beyond specification) Bonus (3 / 2 / 1)					
Total Marks (/100)					

Remarks:

Softcopy Submission	

List of Features

Overview

The “Taylor’s Pantry Management” program is a software designed to help users manage their pantry or kitchen inventory to prevent any loss or wastage in the kitchen. This program allows users to keep track of their kitchen equipment, utensils, ingredients and even recipes. The UI of this program consists of several elements, including a login page, a register page, a main menu, a submenu for each of the four categories as well as several forms for the users to key in the details of the item that they want to add into the system. Below is the list of the features of this system. The program is entirely coded in pure Java with the use of JFoenix library WITHOUT using and drag-and-drop tools.

File Based Storage System

“Taylor’s Pantry Management” keep tracks of user data by saving them into files, which allows the data to be stored and accessed the next time user reopens the program and login with the same account. Every time the program is opened, it will check if the folder “Program Data” exists, if it doesn’t, then the program will generate one automatically on the same path where the program file is located. Inside the folder, the program will again check for the presence of the file “Login.txt” and “Remember.txt” which are crucial for the login function and the “remember me” function respectively. If they do not exist, then the program will generate one automatically. Aside of those two files, a folder with the name “admin” will also be generated automatically, which contains the file “Equipment.txt”,”Utensil.txt”,”Ingredient.txt”, “Recipe.txt” that contain hardcoded data that will be used for demonstration purposes as well as an “asset” folder that is used to store images. The content of these files basically are data that users key into the system, which will be explained in the next few features. This storage system is very crucial for the program to function properly because this program supports access by multiple users and that’s why a database is needed for each user to save and manipulate their data.

User Registration

Before the program can be used, the user must create an account by clicking the “Register now” hyperlink. This will open up a registration form for the user to key in his or her details. If the user signs up successfully, his or her username and password will be saved into the “Login.txt” file , which allows the user to logon to the system and start using the program. Also, a folder with the same name as the username will be created, along with the files “Equipment.txt”,”Utensil.txt”,”Ingredient.txt”, “Recipe.txt” and the “asset” folder in this folder. Basically, the folder is dedicated to store the data of a particular user. There are a few scenarios which will cause the registration to be unsuccessful. Firstly, if the username is already taken by another user, then the program will prompt the user to re-enter a new username. This is achieved by going through the “Login.txt” for the same username. Second, if the password in both “Password” field, and “Re-type Password” field do not match, then the user will be prompted again to re-enter the password. Third, the sign up will also fail if the user does not agree to the terms and conditions by ticking the checkbox. Lastly, if any of the fields are left blank, the sign up will also fail. Basically, various form validation checks will be performed by the system when the user clicks the “Register” button to ensure that all the details are correct. Dialog is used to alert the users in the event of unsuccessful registration.

Remember Me

Remember me is a handy feature that help “remembers” username and password, so the user need not to key in his or her details again the next time program is reopened. Remember me will be activated when the user ticks the Remember Me box, which triggers the system to store the username and password in the file “Remember.txt”. One account information can only be stored at a time so if a new user login with Remember Me activated, the existing account info will be overwritten. Similarly, if the user decides not to use the feature anymore, he or she can untick Remember Me when logging in so the details will be deleted from “Remember me.txt”.

Forgot password

This is yet another useful feature that helps recover password in case the user cannot recall the password. First the user will be asked to enter his or her username and if the username is found in the “Login.txt” file, the associated password will be shown to the user. User will have to click and hold the mouse to see the password.

Login

If the login credentials are present in the Login.txt file, user will be granted access to the program. If not, user will be prompted to try again.

Main Menu

Main menu is another feature in the program that helps user to navigate to different function of the program. It consists of a background image, four giant buttons each leading to a different submenu, a clock and a quick tip at the bottom. Firstly, let’s talk about the clock. The clock on top of the menu shows the current time, and is achieved by using the Animation class in Javafx . The clock animation is running infinitely so the digits are updated as the time goes by. Secondly, the buttons. The buttons are decorated with exquisite image, and they move and change the background image on hover. The quick tip at the bottom on the other hand is randomly shown every time the user logs in. We have a list of hardcoded tips to be randomly shown to the user. It is also interesting to note that the background music will play in the main menu page and is stopped when the user logs out or go to other submenus.

Equipment, Utensil, Ingredient & Recipe submenus.

Each of the four categories has its own submenu dedicated to access and manipulate data in the category of equipment, utensil, ingredient and recipe respectively. Most of the interface and functions are similar across all four submenus. Below is a list of features and functions found in each submenu.

1. **Tile-based display pane.**

   At the center of each submenu is a display pane that shows all the existing items in a user database arranged in tiles. Similar to popular e-commerce app like Shopee and Lazada, the items are shown in rows and columns of tiles which can also be scrolled if the display pane cannot accommodate all items at once. This feature is realized using the combination of tile pane and scroll pane placed at the center of a border pane. Each time a submenu opened, the program will read the corresponding data file (e.g. Recipe.txt if recipe submenu is opened) and generate a tile for every item found. The tile is then added and shown through the display area. Each tile contains the image of the item, the name of the item, a delete button and an edit button.

2. **Add (item)**

   The ability to make entries is the core function of any management system. When the user clicks the "Add [item]" button, a stage that contains a form will be shown for user to key in the details of the item. When the user is finish, the data in each data field will be stored line by line in the data file. For example, if the data consists of brand, name, measurement and color, brand is written in the first line, name is second, measurement is third and so on. The program is coded in such a way when it reads from the file, the data is identified through the position of the line that the data is located in. For example, the program will know that the data in the first line is brand, second line is name, so on and so forth and when all of the four data have been read, the program will know that from fifth line onwards the data belong to a new item or entry.

   i. **Ability to insert image**

      In the adding form, users are able to insert related image of the item being added by dragging the image file into a specific area. The image added will be shown in the tile and if no images are added, the tile will show "No image available" by default. When the user drags the image into the form, the path to the image will be located. Then, the program will copy the image over to the asset folder so the image won't be lost .If an image shares similar file name with an existing image in the asset folder, the program will modify the file name by adding a “1” in front, or a bigger number if the previous number exists.

3. **Show item**

   When the user clicks on a tile, a new stage showing the full information of that particular item is displayed to the user. So how does the system locate the corresponding data from the data files? When a tile is generated, an index number is assigned to it based on first come first serve basis. For example, say if all the items belong to a category have 5 data fields each, and the first 5 line of the data file belong to the first item, then a tile will be generated for this item first with an index number of 1. Then the next 5 line will have an index number of 2 and so on. Based on the index number, the system is able to locate the data of a particular item by multiplying it with the number of data field and minus by the number of data field. For example, index 3 * 5 is 15, minus 5 is 10 so the program will know the data from 10th line to 15th line belongs to the item with index 3. Thus, all the data can be retrieved correctly.

4. **Edit item**

   By clicking the edit button, users will be able to edit the data of a particular item. Using similar mechanism as the show item function, the program first locates and display the data to the user in another stage. Then. The user will be able to make changes to the data simply by modifying the data in the text field. When a data is changed, the program will first create a temporary file and copy over the data file line by line until it reaches the data of the item being edited. These data will not be copied over but instead, newly keyed in data will be inserted to replace the existing data instead. Then the original file is deleted, and the temporary file is renamed to the original file.

5. **Delete item**

   By clicking the delete button, user will be able to delete the item. The delete function is similar to the edit mechanism, a temporary file is created will all the data being copied over line by line from the original one until the program reaches the data of the item being deleted. These data will be omitted, and the program continues copying the remaining data. Then the original file is deleted, and the temporary file is renamed to the original file. The delete button is also assigned an id that is the same as the index number so the program know when the line containing the data of the item being deleted is reached.

6. **Filter**

   In each category, the items are further classified into different types. For example, in the Equipment submenu, items are classified based on equipment type such as skillet, blender or oven. These classifications are needed in order for the filtering to works. On the left panel of each submenu exists a list of checkboxes used for filtering the items to be shown in the display pane. When a checkbox is ticked, the program will check for the type of the item before generating a tile to be shown in the display pane. The tile will only be generated if there is a match. The filter support multiple filtration.

   i. **Country Filter**

      In the Ingredient submenu, an extra country filter is added to filter the items from certain country of origin.

7. **Search**

   Search function works similarly to filter function. Items will only be shown if there is a match between the keyword entered by the user and the brand/name of the item.

8. **Table view**

   If the user prefers a more systematic and neater way to see the data, the program is also capable of showing the data in table. In table, all the details will be shown in columns in contrast to the tile view where user will have to click the tile in order to see the detailed information. This feature is realized by using the Table View class in JavaFX.

9. **Data Corruption Precaution Measure**

   Because the program relies heavily on the structure of the file, the files are therefore not to be tampered with. In the event of inappropriate modifications made to the data file, the file will be replaced with a new empty file and the program will restart.

10. **Various form validation check**

    Form validation check such as only number can be accepted in certain fields, cannot leave blanks and many more.

11. **Ingredient Expiry Notice**

    User will be notified if any ingredients in the database are found to be expired.

Work Breakdown Structure/ List of coding contribution

Name	Student ID	Work Breakdown
Timothy Louis Scott	0332841	Coding: 1. Login page & Login function 2. Remember Me function Documentation: 1. Screenshot of the program
Sarah Chong Giet Yee	0336289	Coding 1. Tableview function 2. Filter function 3. Main Menu Documentation 1. Problems encountered 2. List of features
Ong Sze Yun	0332474	Coding 1. Tableview function 2. Search function 3. Main Menu Documentation 1. UML diagram
Tan Xuan Ni	0332197	Coding 1. Register page and function 2. Validation check 3. Testing Documentation 1. Enter Dummy data into the program
Ee Sun Jun	0332103	Coding : 1. Submenus 2. Add, Edit, Delete function 3. Forgot password Documentation : 1. List of features 2. Assumptions Made

Assumptions made

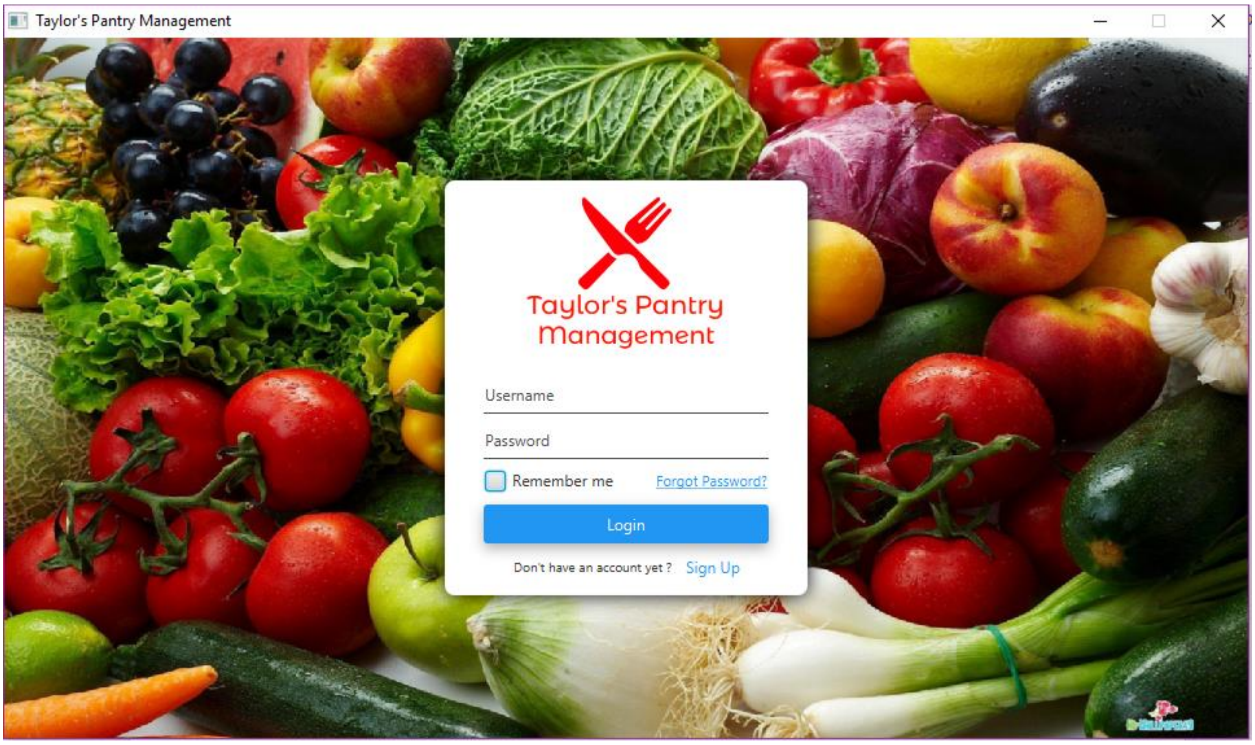
1. We assume that the resolution of the end user’s display/monitor is the same as the resolution of our display/monitor in which we use for the development of this program. This is because the layout of the program will be out of the place when viewed in a display with high resolution (1080p). Therefore, this assumption is made unless the user knows how to adjust the scaling of his or her display.

2. We assume that the end user’s machine has Java 1.8 installed, which is the environment that we used for the development of this program since other versions might have problem running our program. For example , Java 12 does not have JavaFX pre-installed and therefore is not capable of running our program which relies heavily on JavaFX.

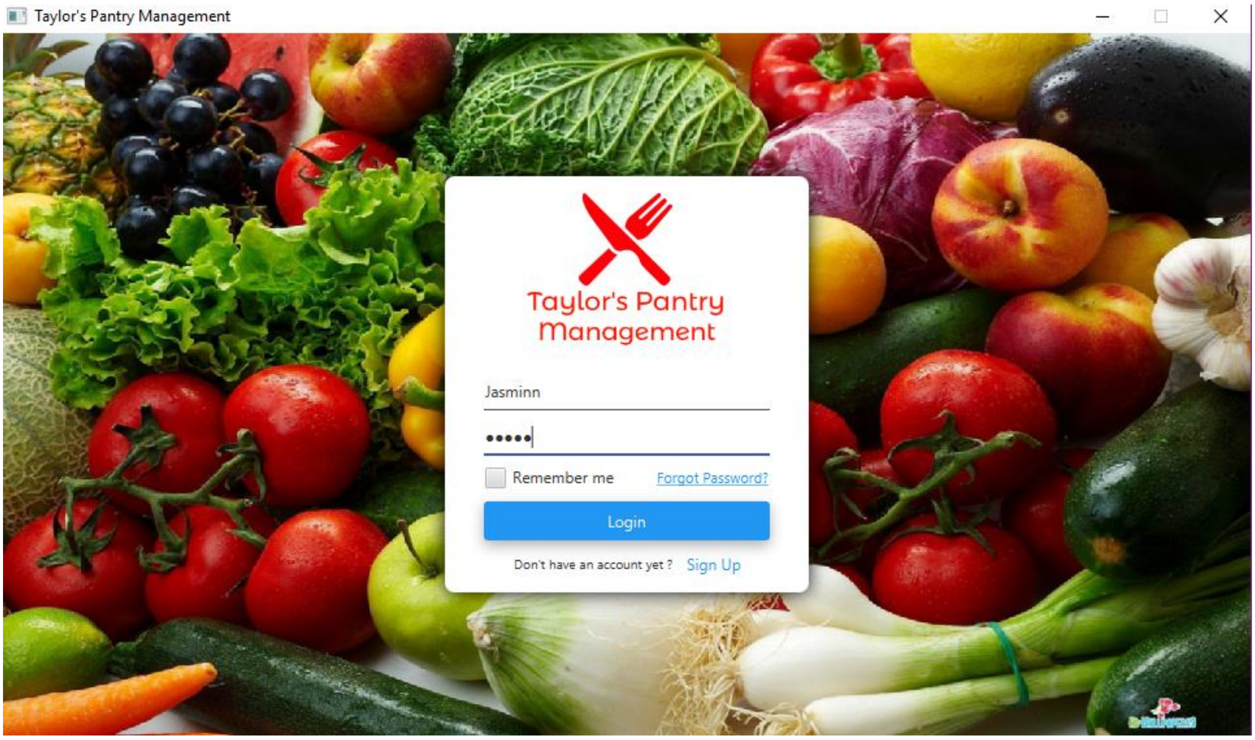
UML Diagram

Screenshots of the Program

Login


Taylor's Pantry Management
Taylor's Pantry Management
Username
Password
□ Remember me
Forgot Password?
Login
Don't have an account yet ?
Sign Up

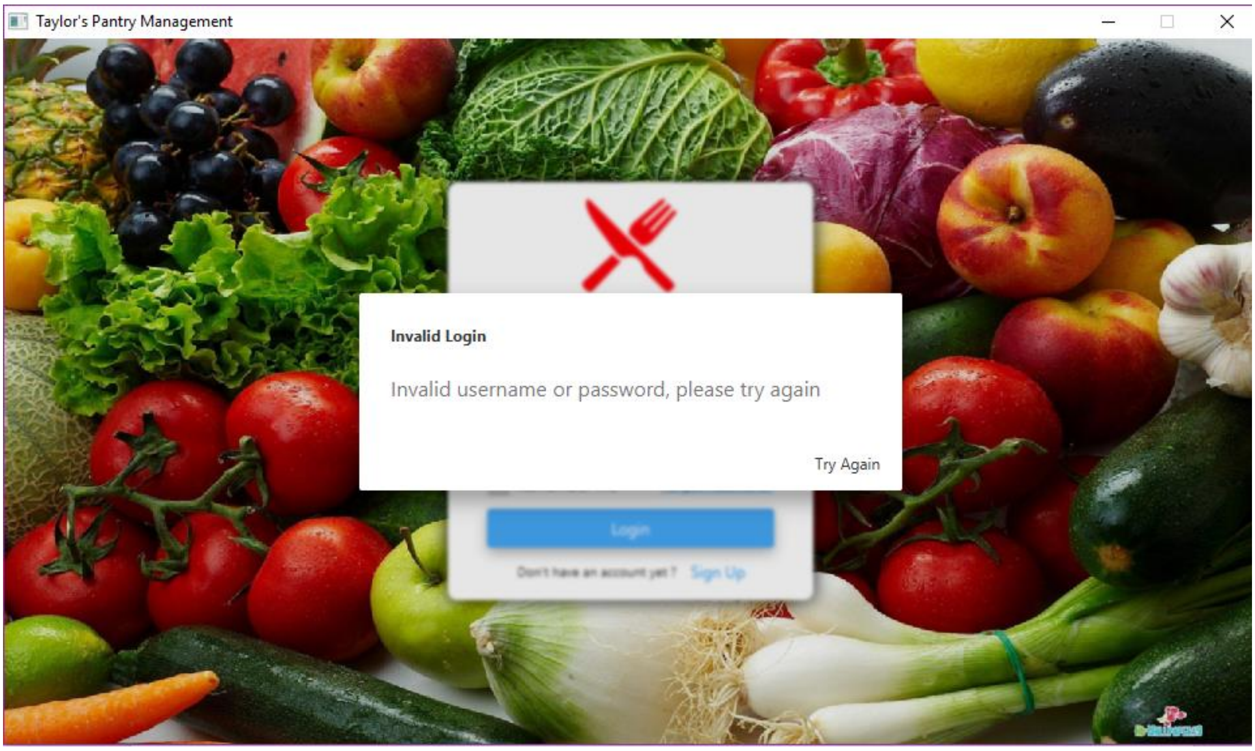
This is the first page that appears when the user runs the program. If the user has an existing account, they can enter their login details, as shown below.


Taylor's Pantry Management
Taylor's Pantry Management
□ Remember me
Forgot Password?
Login
Don't have an account yet ?
Sign Up

Jasminn

•••••

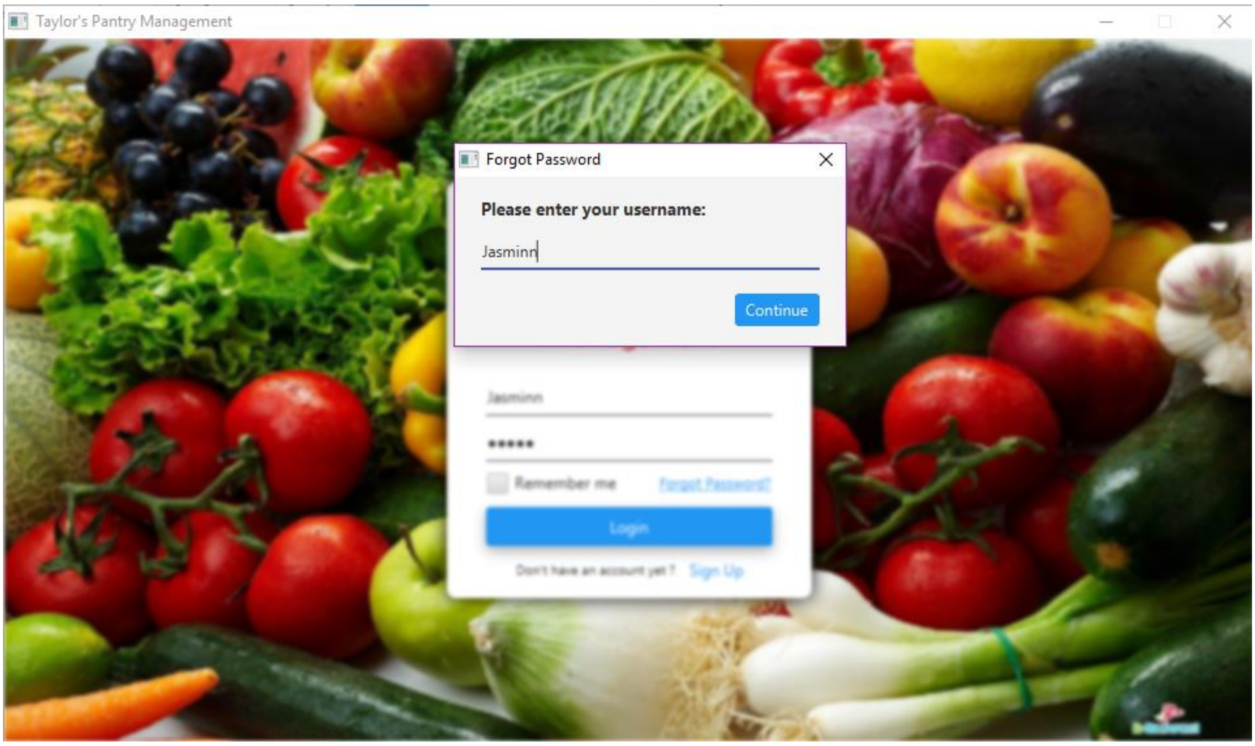
However, there is a possibility that the user may have inputted incorrect login details because they have forgotten them.


Taylor's Pantry Management
Login
Don't have an account yet ? Sign Up

Invalid Login

Invalid username or password, please try again

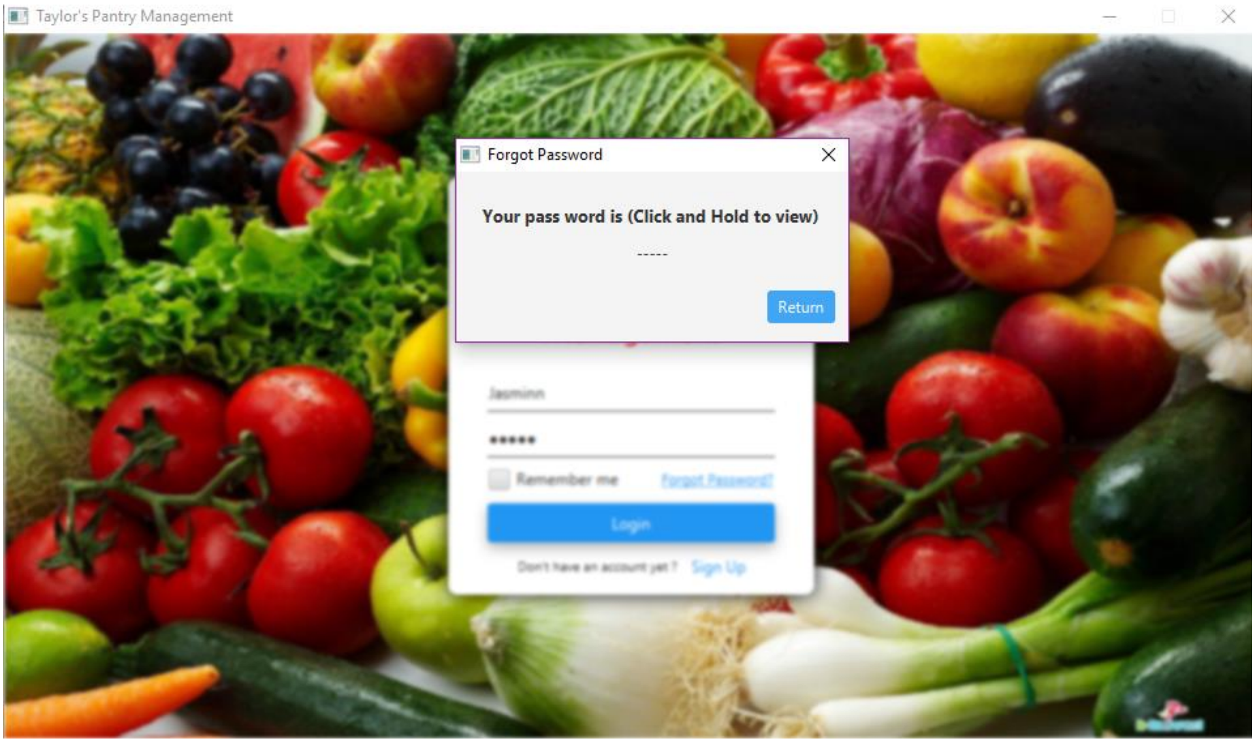
If the user has forgotten their password, they can click ‘Try Again’ and then click ‘Forgot Password’. They will be prompted to enter their username.



Please enter your username:

Jasminn

After this, the system will remind them of their password.

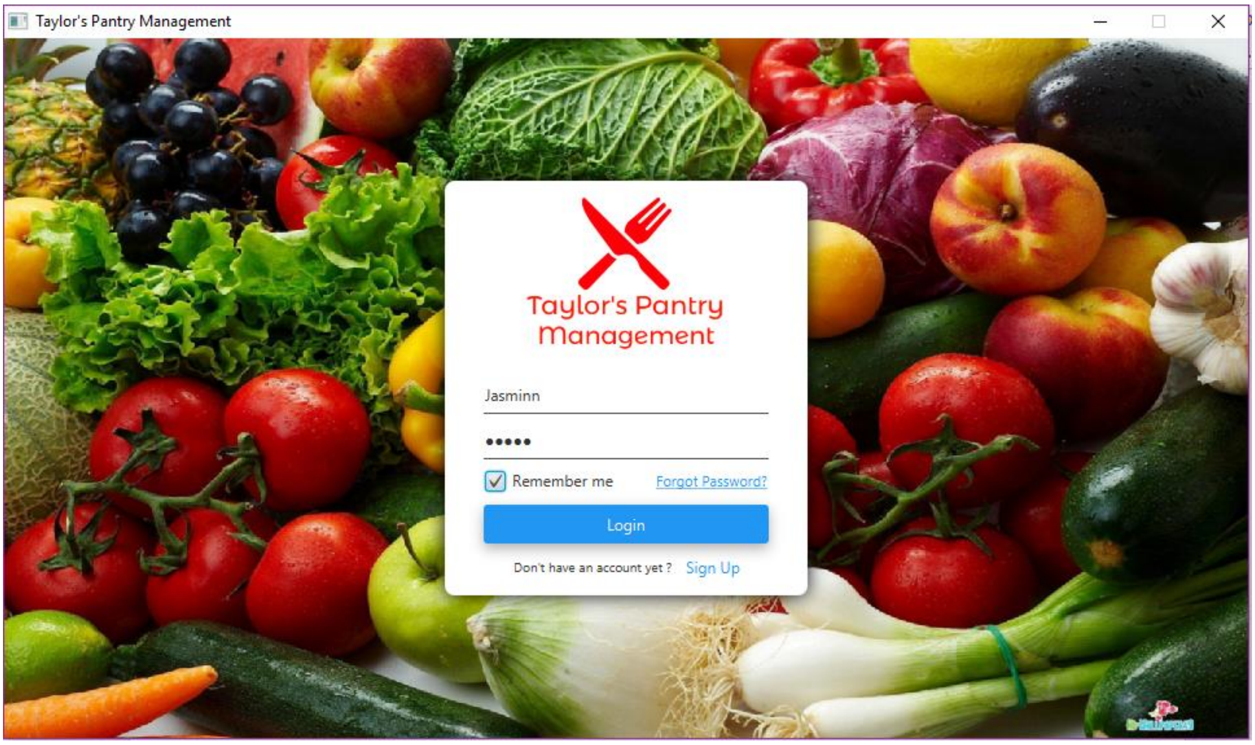


Your pass word is (Click and Hold to view)

Jasminn

•••••

The user can then enter their correct password and tick the 'Remember Me' box so their details will automatically be typed out the next time they open the program.

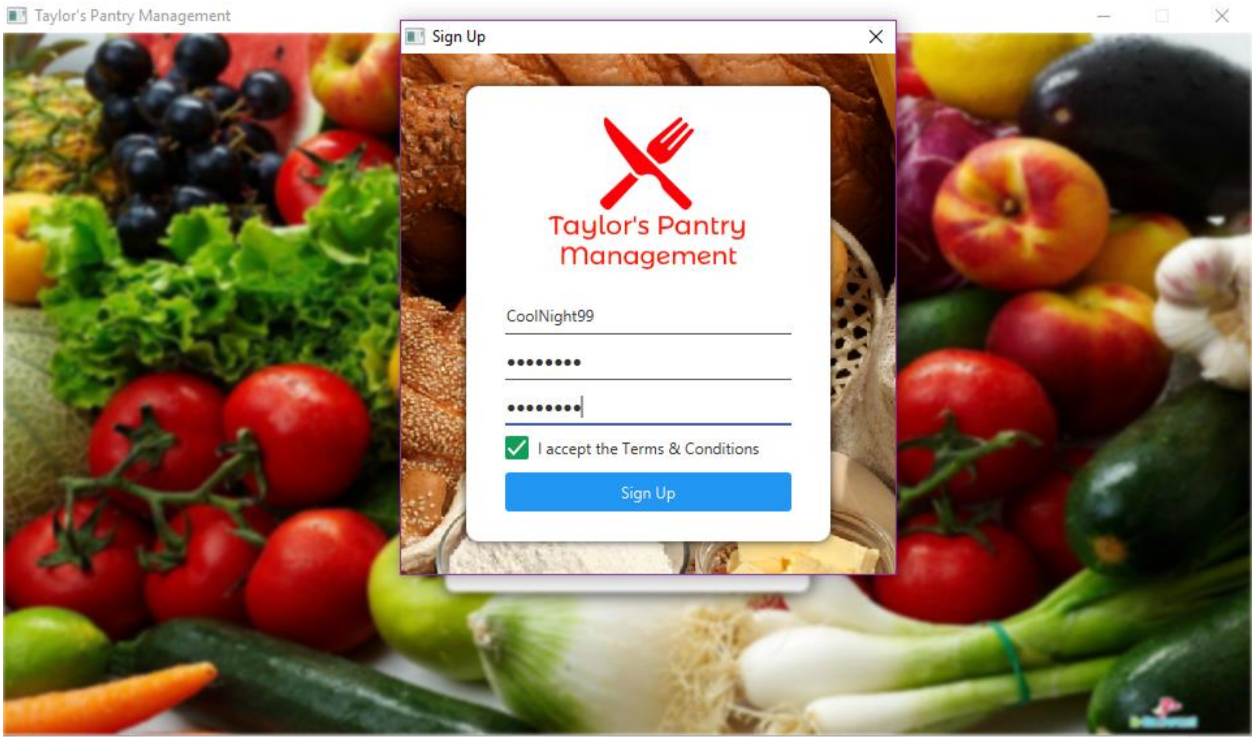

Taylor's Pantry Management
Taylor's Pantry Management
☑ Remember me
Forgot Password?
Login
Don't have an account yet ?
Sign Up

Jasminn

•••••

Sign Up

If a new user is using the system, they can click the 'Sign Up' button on the Login page. They will be asked to enter their details.



CoolNight99

•••••••••

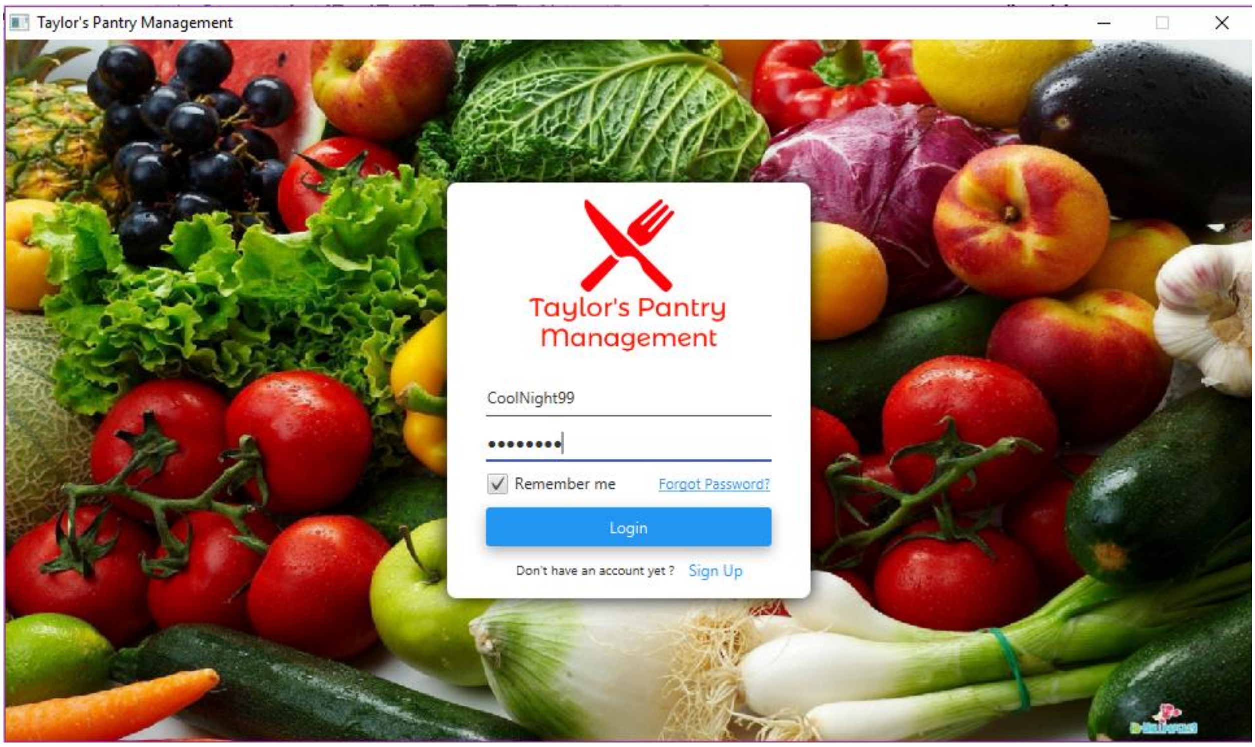
•••••••••

☑ I accept the Terms & Conditions

They can then use their details to access the system.

Main Menu

After logging in successfully, the user will be taken to the main menu.

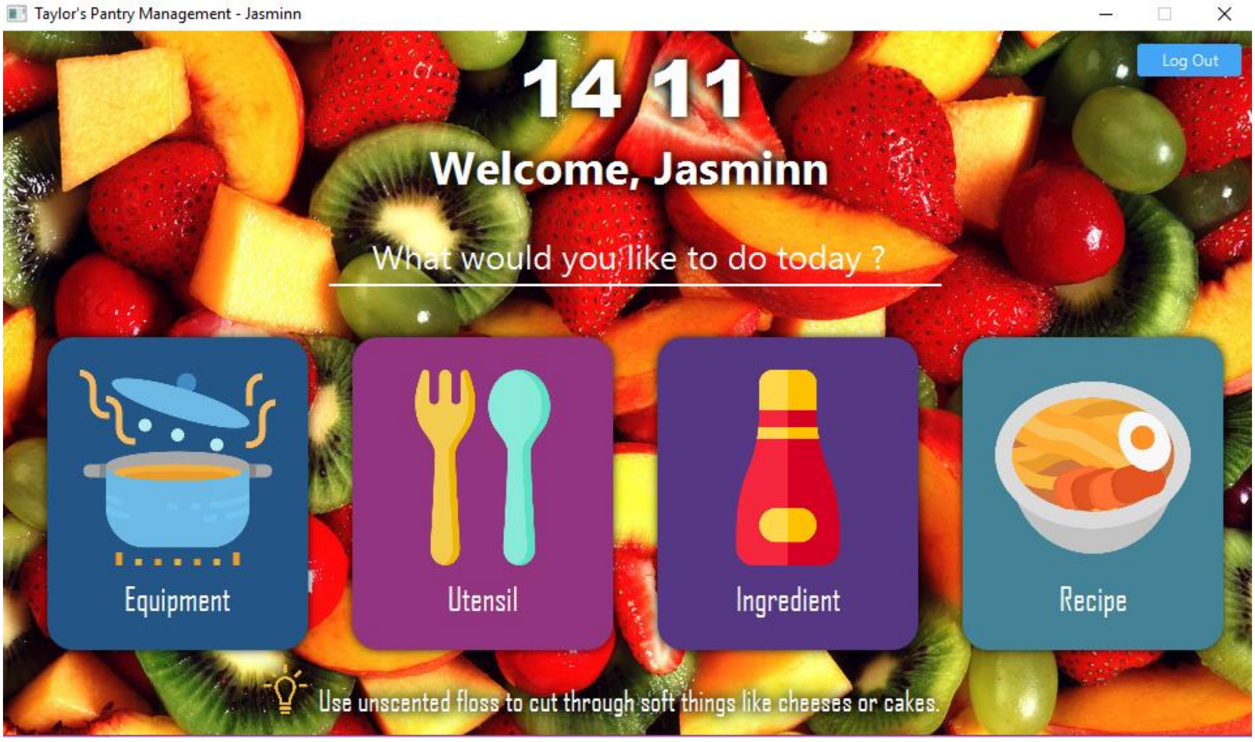

Taylor's Pantry Management
Taylor's Pantry Management

CoolNight99

•••••••••

☑ Remember me

Forgot Password?
Login
Don't have an account yet ?
Sign Up


Taylor's Pantry Management - Jasminn

14 11

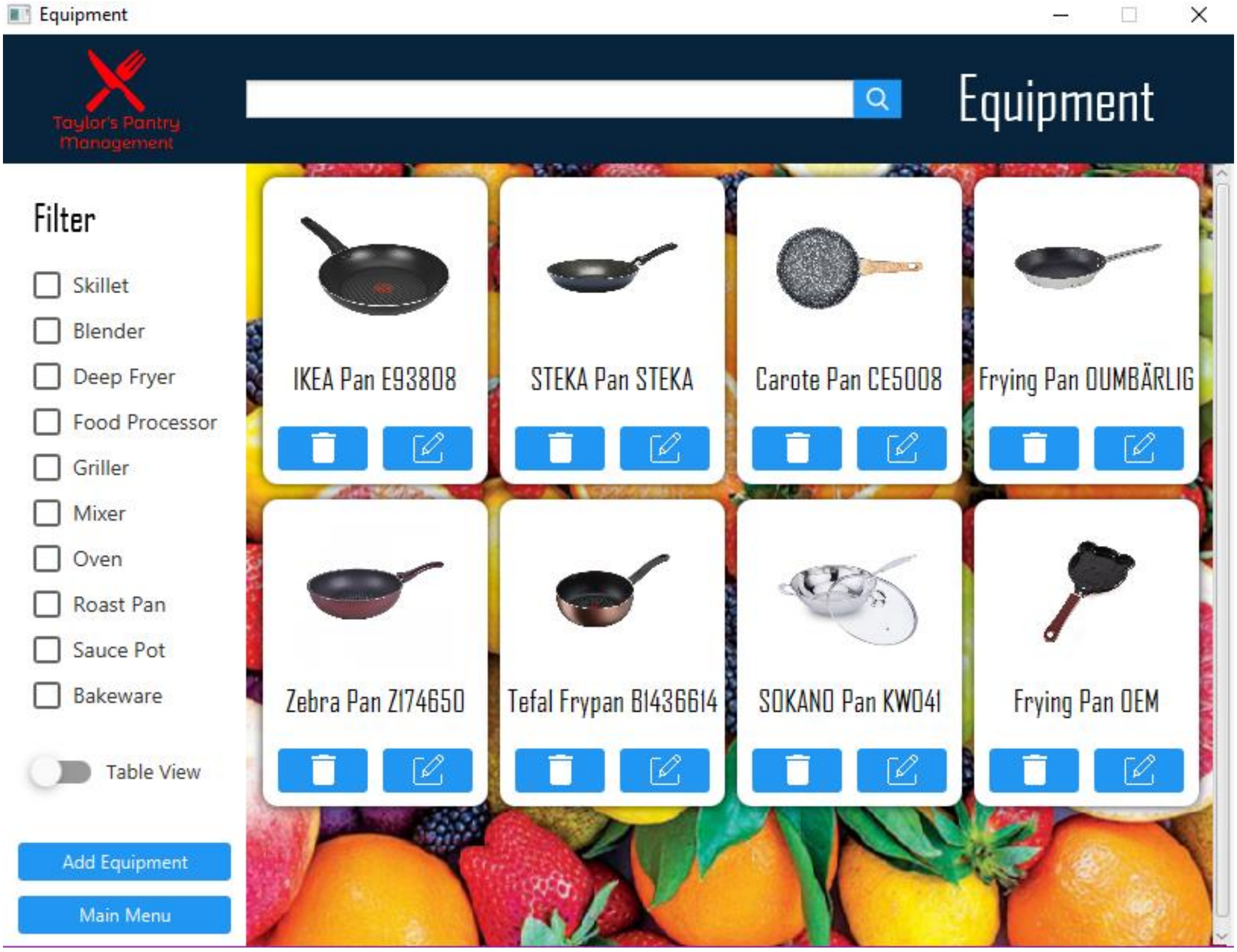
Welcome, Jasminn

What would you like to do today ?

Use unscented floss to cut through soft things like cheeses or cakes.

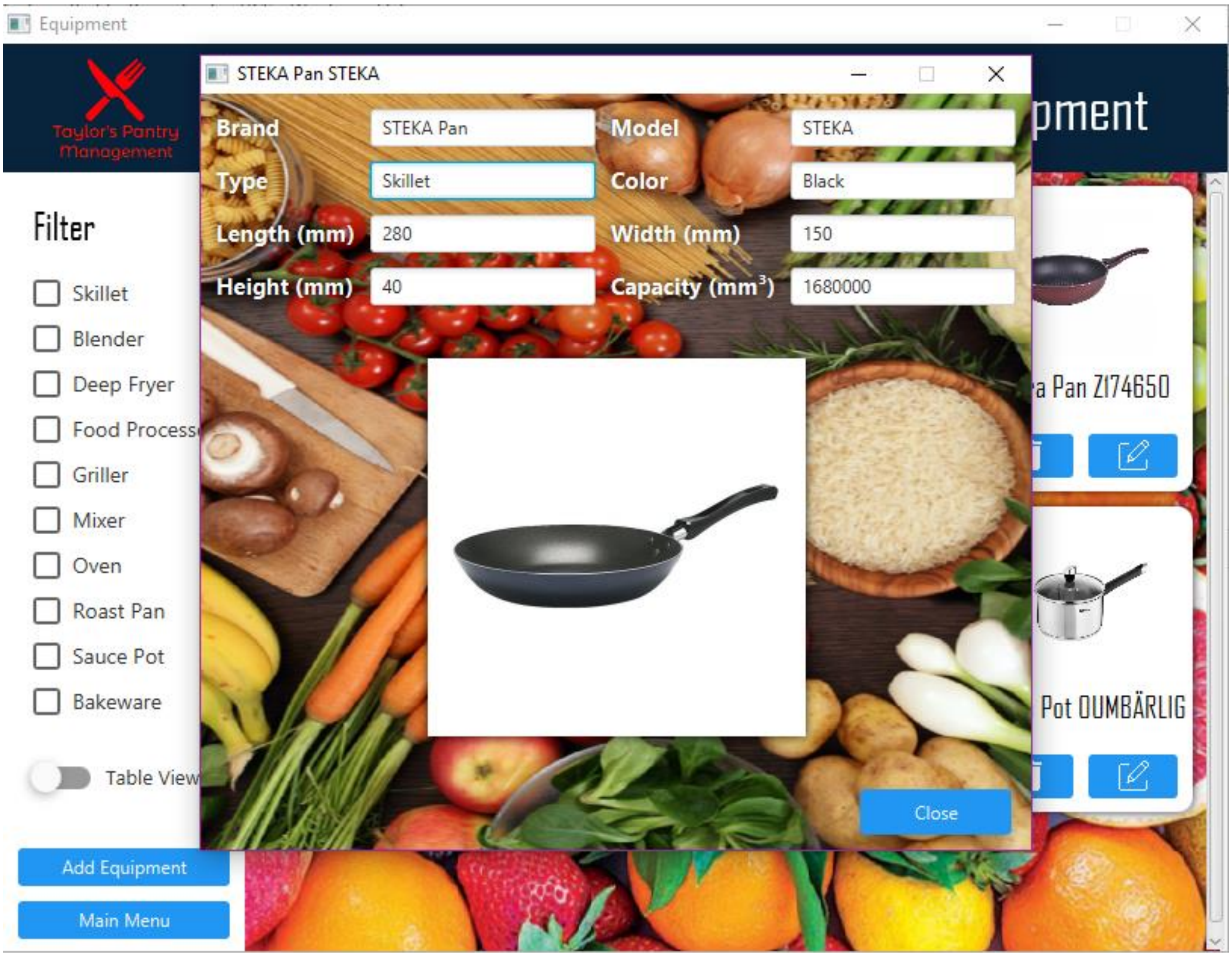
The main menu displays four sub-menus, the current time in 24-hour format, a welcome message, a useful tip that is randomized and a ‘Log Out’ button. The user can click on any of the sub-menus to see what is currently stored in the pantry.

Equipment



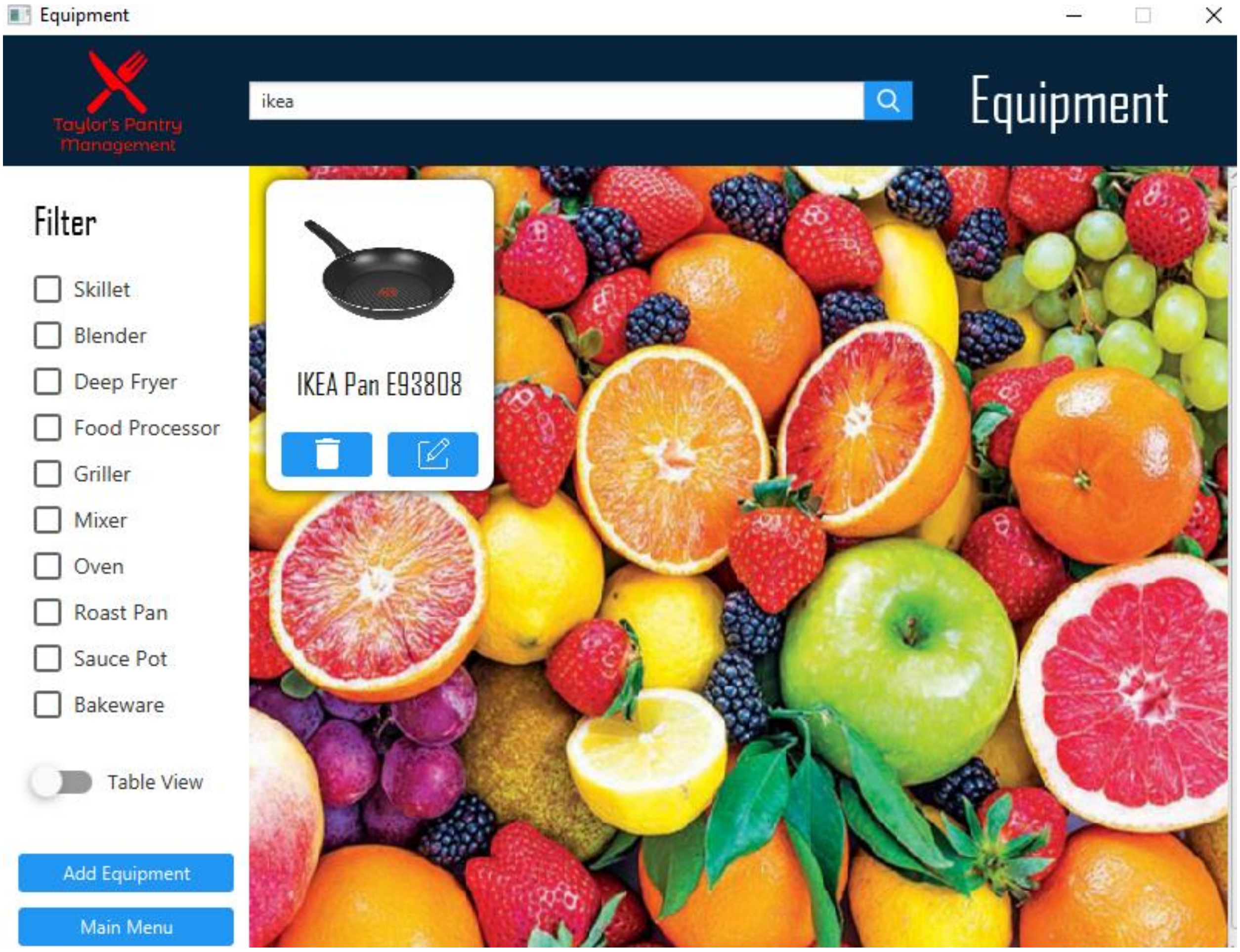
The Equipment page contains a list of equipment in the pantry that is shown in grid view by default.

By clicking on one of the boxes, the user can view it in more detail.

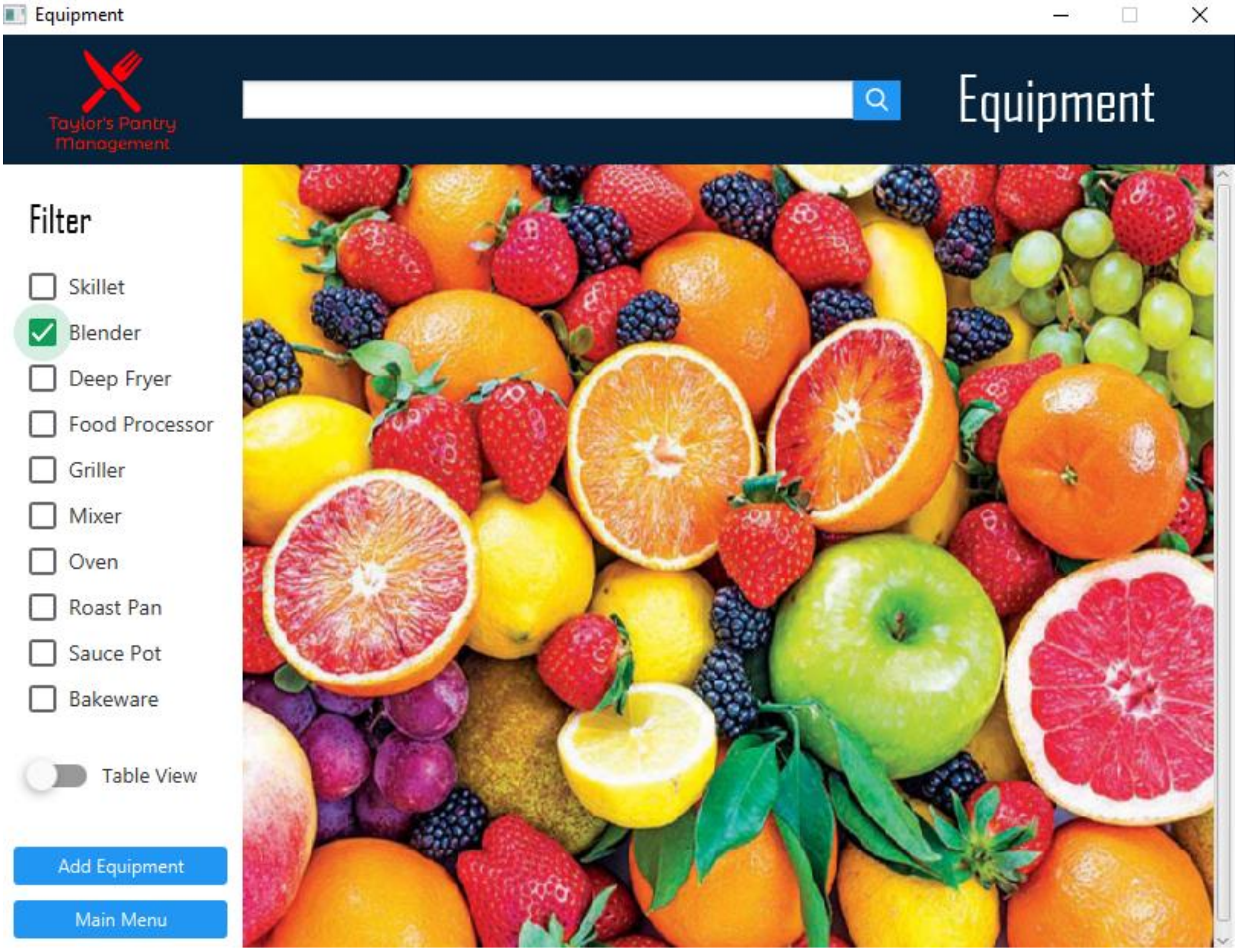

STEKA Pan STEKA

Brand	STEKA Pan	Model	STEKA
Type	Skillet	Color	Black
Length (mm)	280	Width (mm)	150
Height (mm)	40	Capacity (mm^3)	1680000

The user can also search for what they are looking for by typing in the search bar.



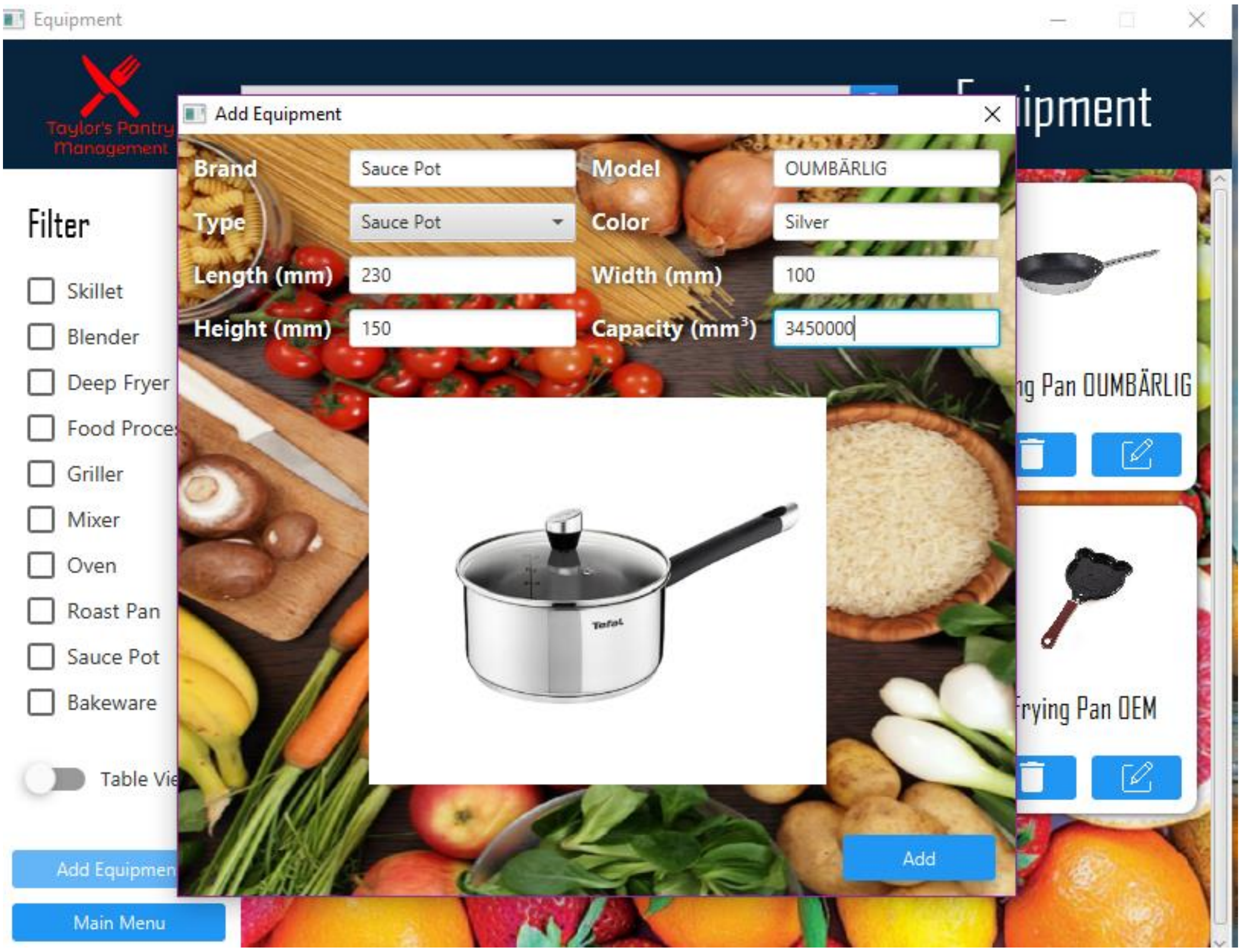
Not only that, the user can filter the equipment by type to narrow down their search. When the user ticks the ‘Blender’ checkbox, it shows no results, which means the pantry currently does not have any blenders.



□ Skillet
☑ Blender
□ Deep Fryer
□ Food Processor
□ Griller
□ Mixer
□ Oven
□ Roast Pan
□ Sauce Pot
□ Bakeware

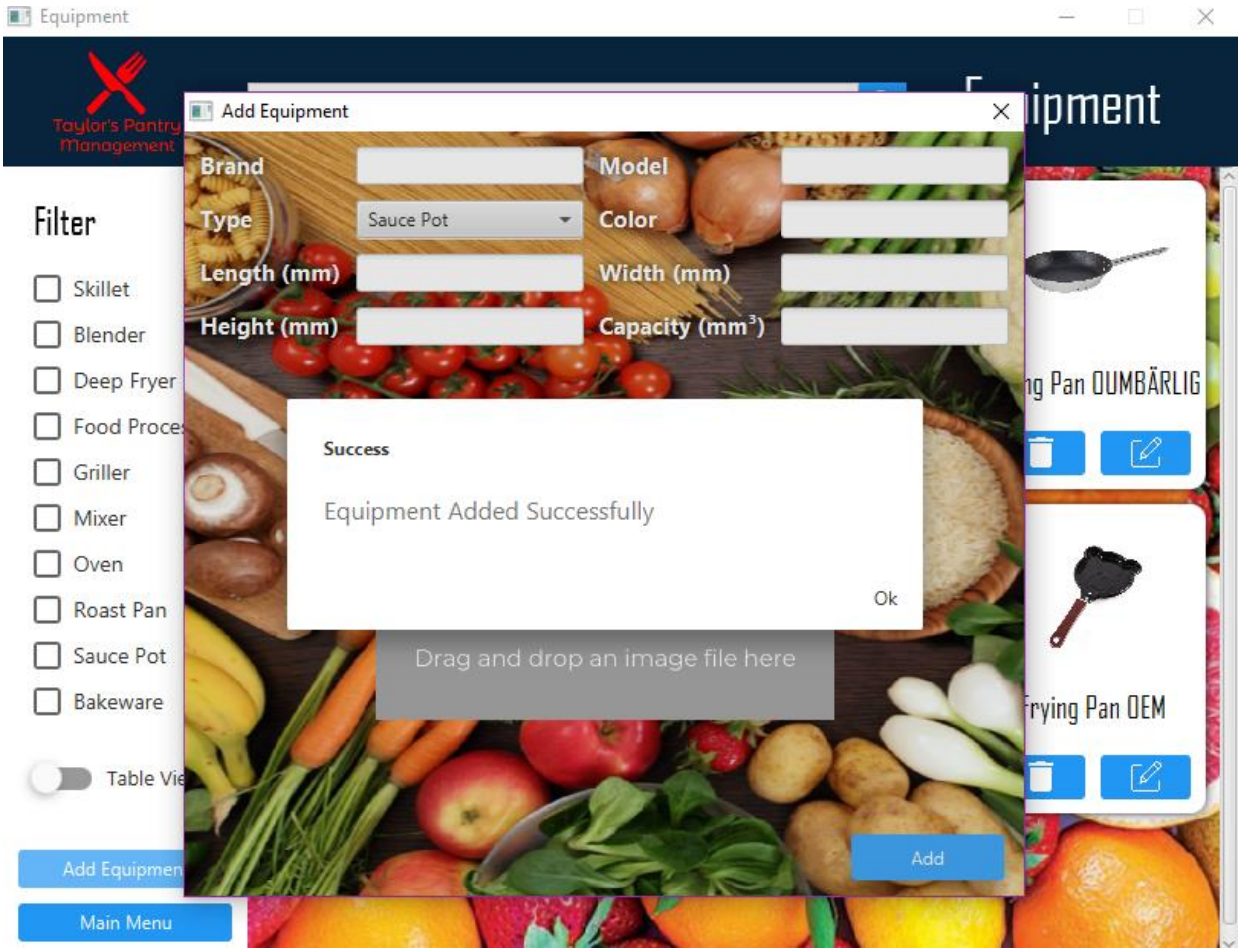
○ Table View

Furthermore, the user can add new equipment to the system.



Brand: Sauce Pot
Model: OUMBÄRLIG
Type: Sauce Pot
Color: Silver
Length (mm): 230
Width (mm): 100
Height (mm): 150
Capacity (mm^3): 3450000

Once they click the ‘Add’ button, a message will pop up to say that the equipment was added successfully.



Brand

Model

Type: Sauce Pot

Color

Length (mm)

Width (mm)

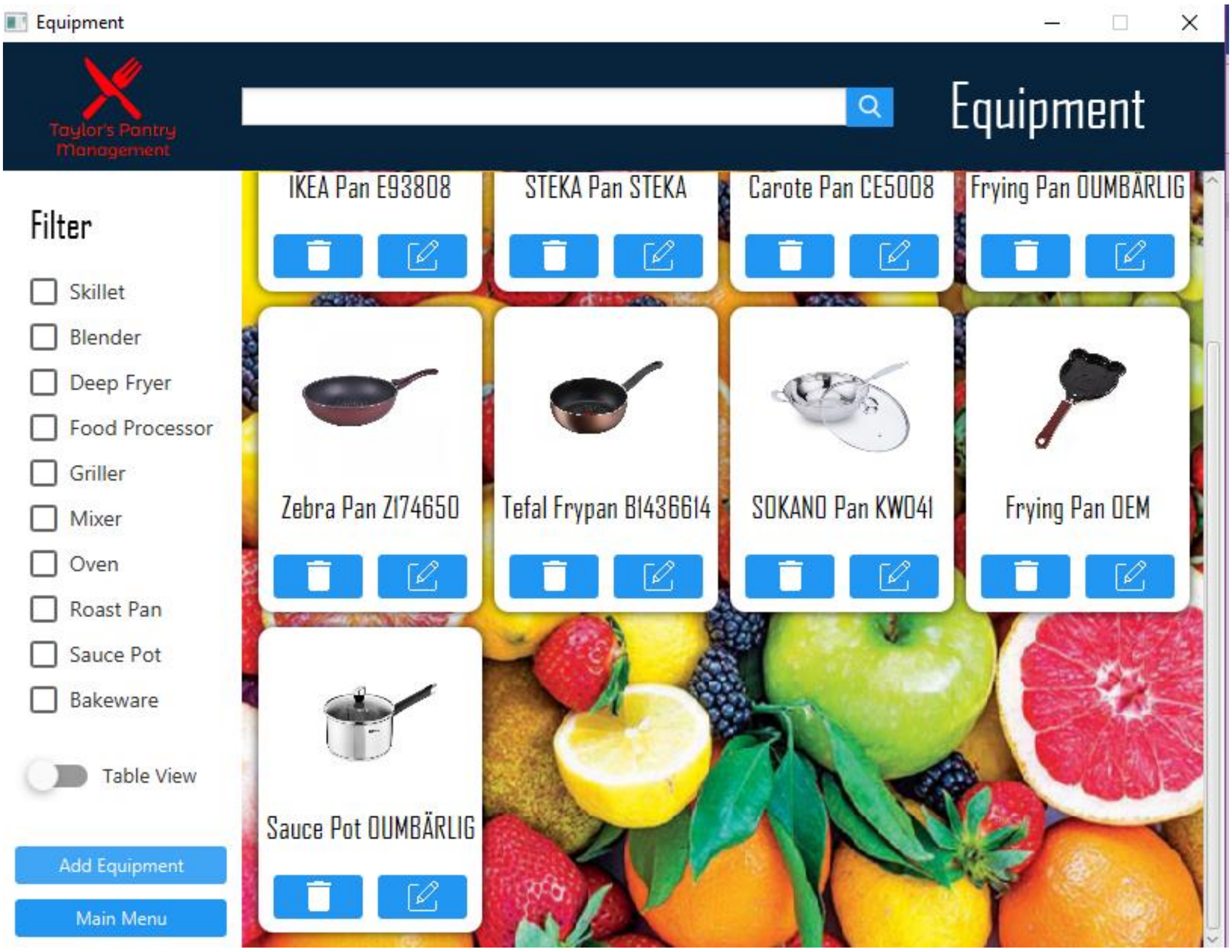
Height (mm)

Capacity (mm^3)

Success

Equipment Added Successfully

The new equipment will be added to the system and they can view it in grid or table view.



□ Skillet

□ Blender

□ Deep Fryer

□ Food Processor

□ Griller

□ Mixer

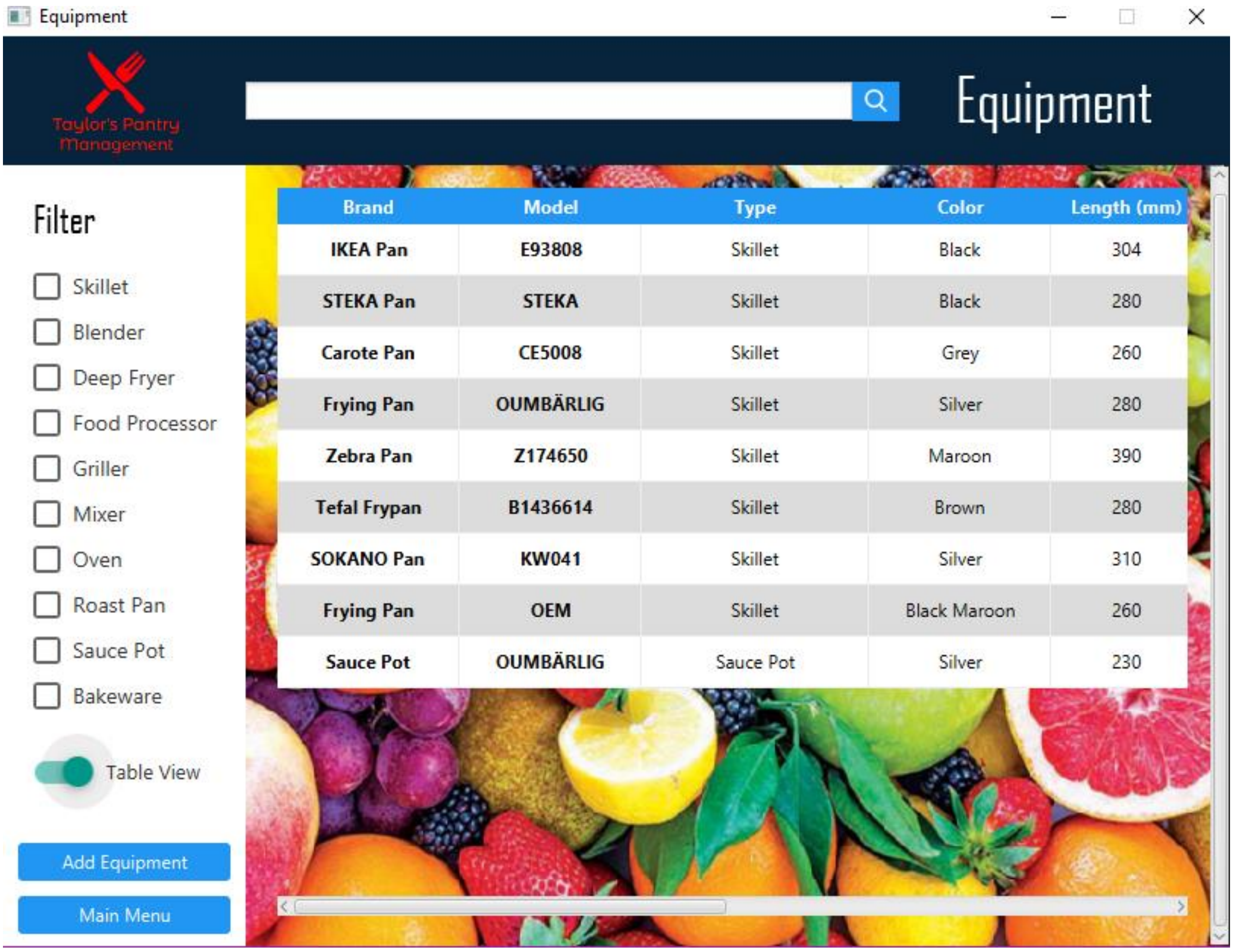
□ Oven

□ Roast Pan

□ Sauce Pot

□ Bakeware

○ Table View



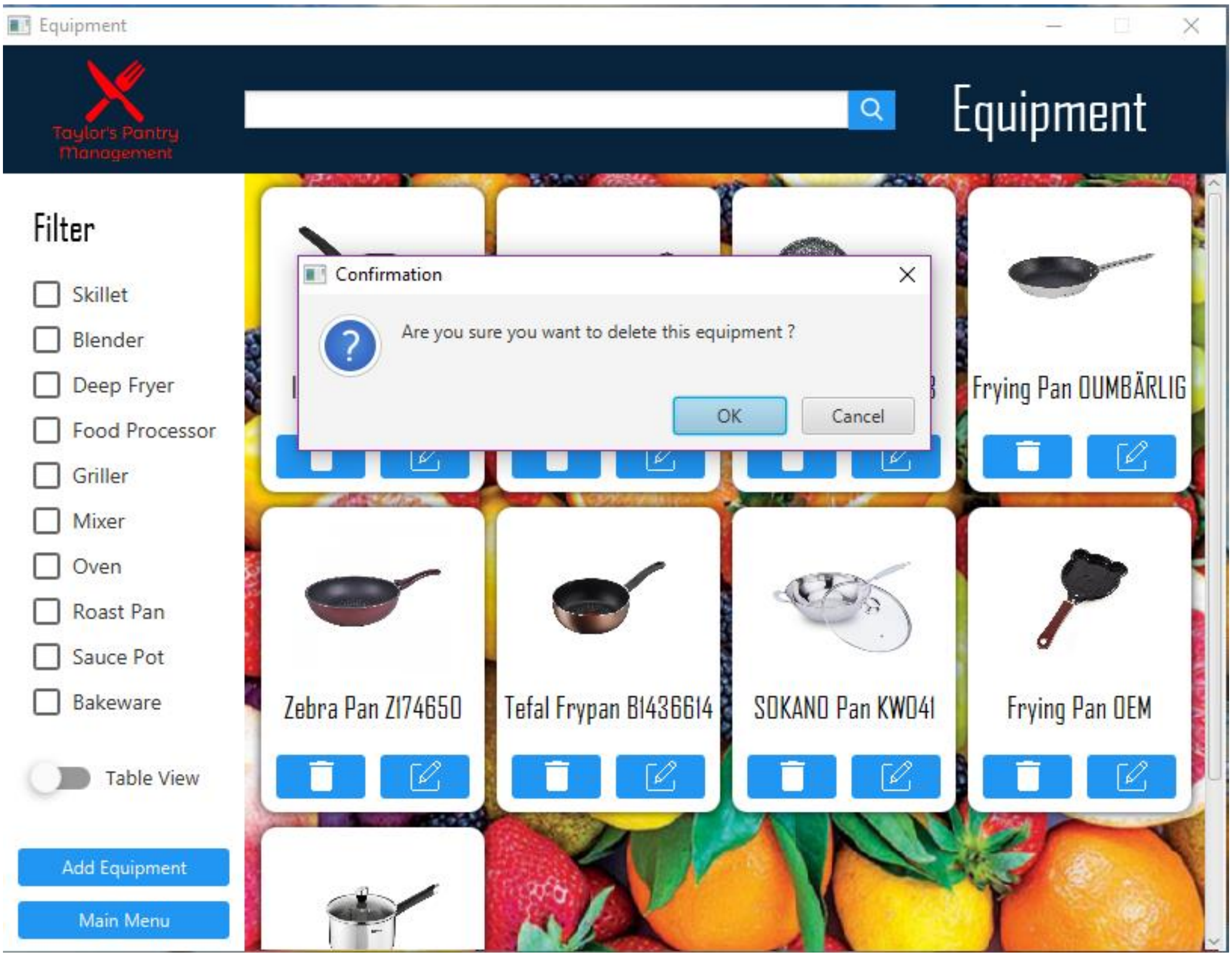
Filter

- □ Skillet
- □ Blender
- □ Deep Fryer
- □ Food Processor
- □ Griller
- □ Mixer
- □ Oven
- □ Roast Pan
- □ Sauce Pot
- □ Bakeware

● Table View

Brand	Model	Type	Color	Length (mm)
IKEA Pan	**E93808**	Skillet	Black	304
STEKA Pan	**STEKA**	Skillet	Black	280
Carote Pan	**CE5008**	Skillet	Grey	260
Frying Pan	**OUMBÄRLIG**	Skillet	Silver	280
Zebra Pan	**Z174650**	Skillet	Maroon	390
Tefal Frypan	**B1436614**	Skillet	Brown	280
SOKANO Pan	**KW041**	Skillet	Silver	310
Frying Pan	**OEM**	Skillet	Black Maroon	260
Sauce Pot	**OUMBÄRLIG**	Sauce Pot	Silver	230

The user can also choose to delete equipment from the system by clicking on the trash can icon. They will then be asked to confirm their decision.



Confirmation

Are you sure you want to delete this equipment ?

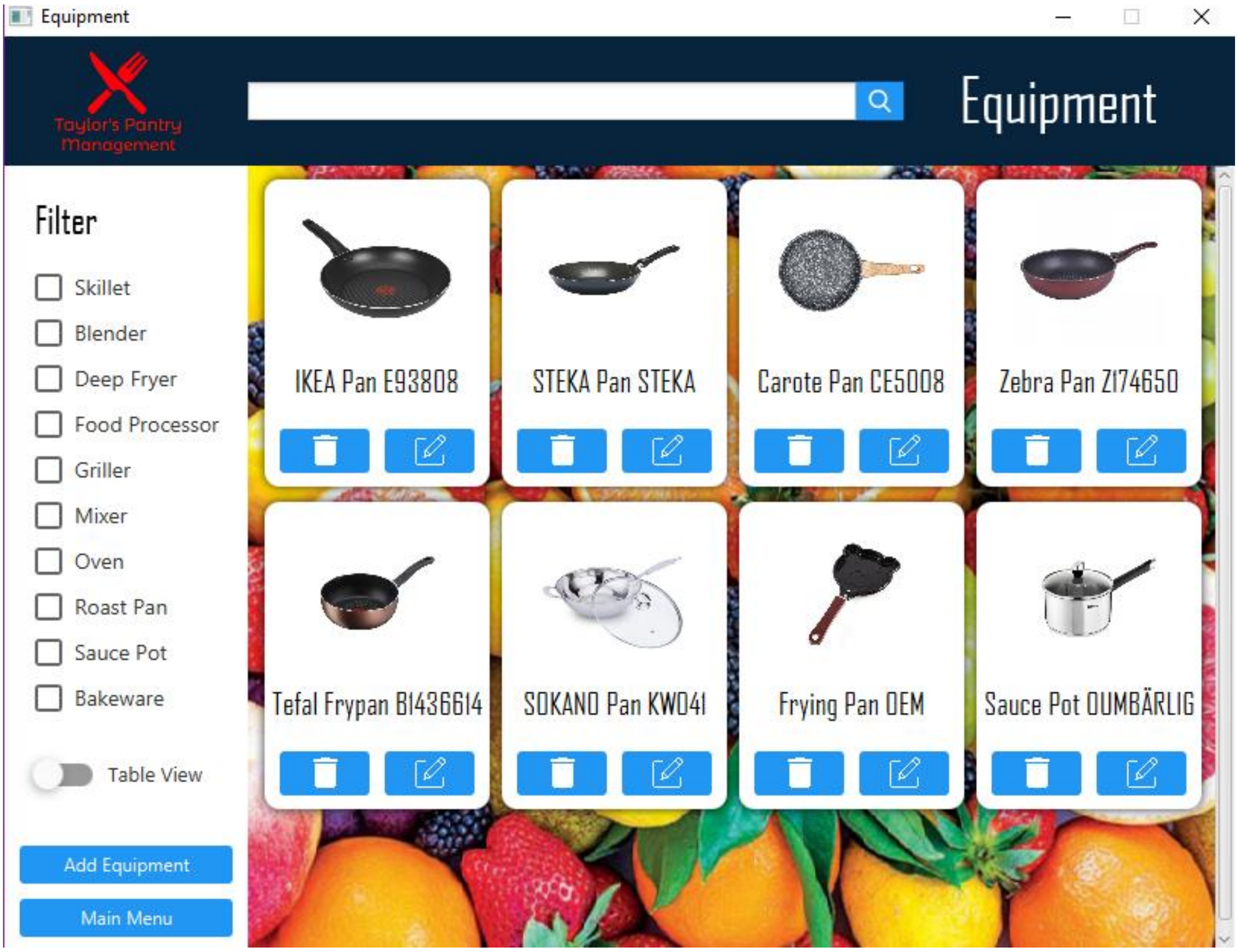
OK Cancel

Filter

□ Skillet
□ Blender
□ Deep Fryer
□ Food Processor
□ Griller
□ Mixer
□ Oven
□ Roast Pan
□ Sauce Pot
□ Bakeware

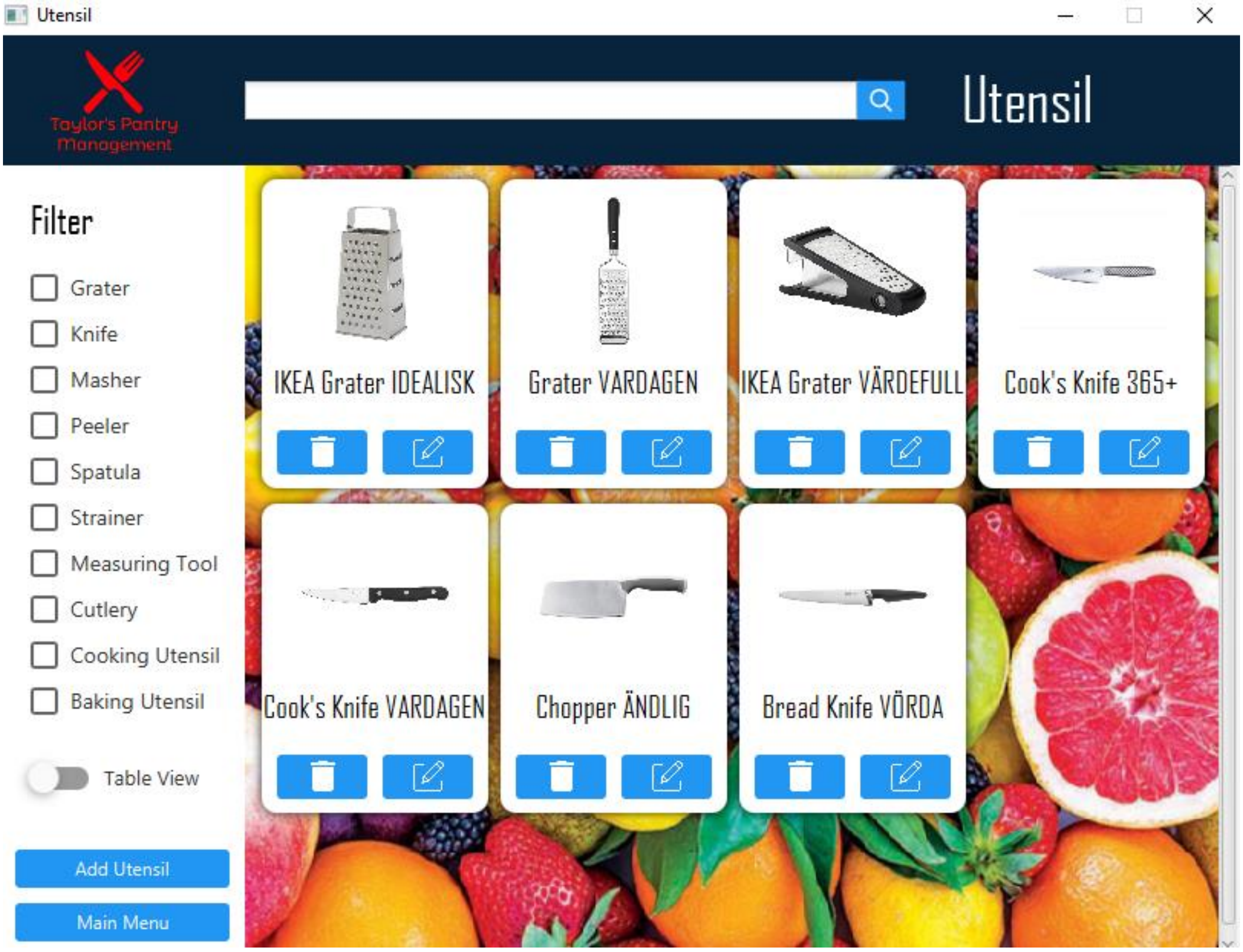
Table View

After deleting the item of their choice (in this case, the Frying Pan OUMBÄRLIG), it will no longer be in the system.



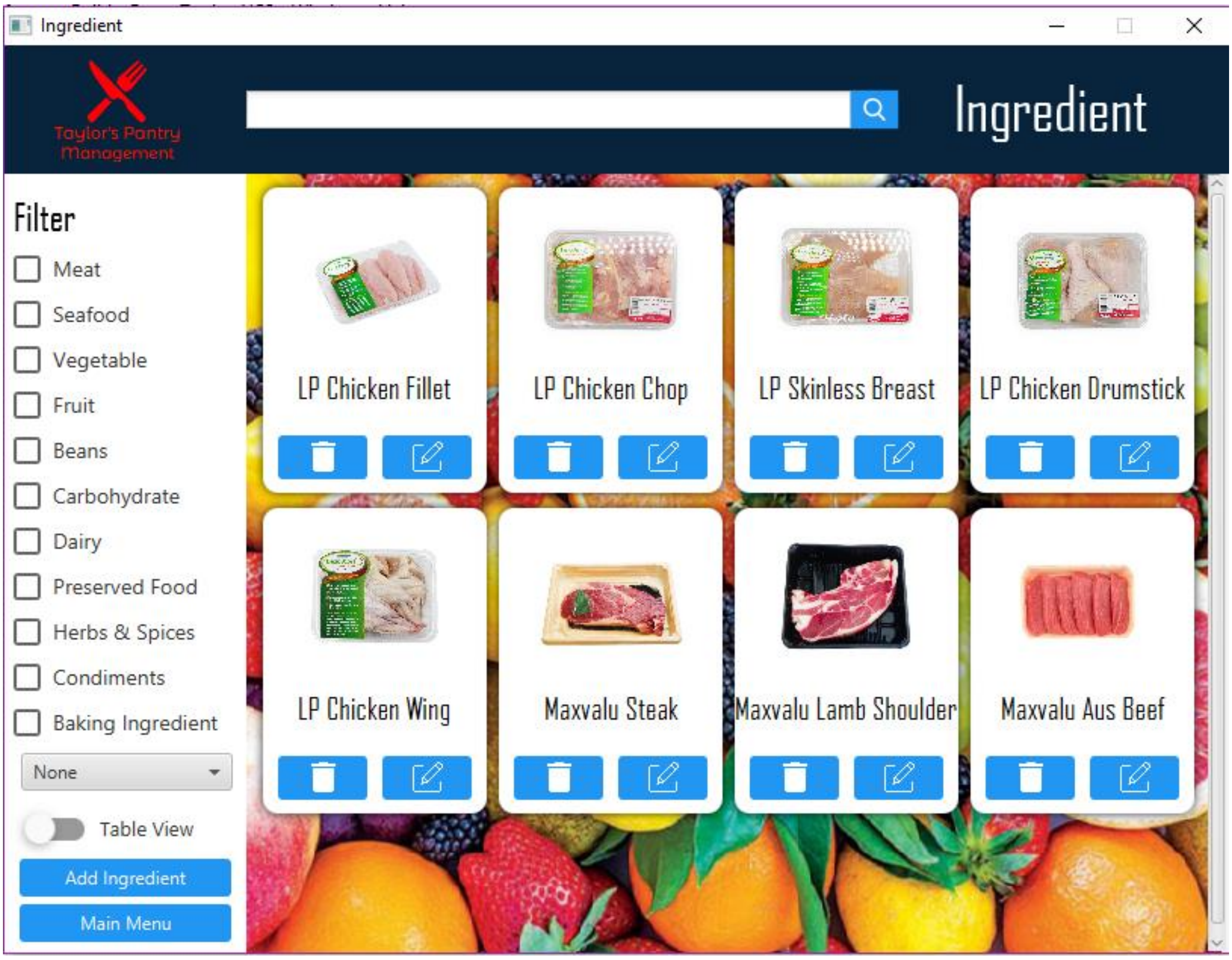
All of the features in the Equipment sub-menu are also available in the other three sub-menus.

Utensil



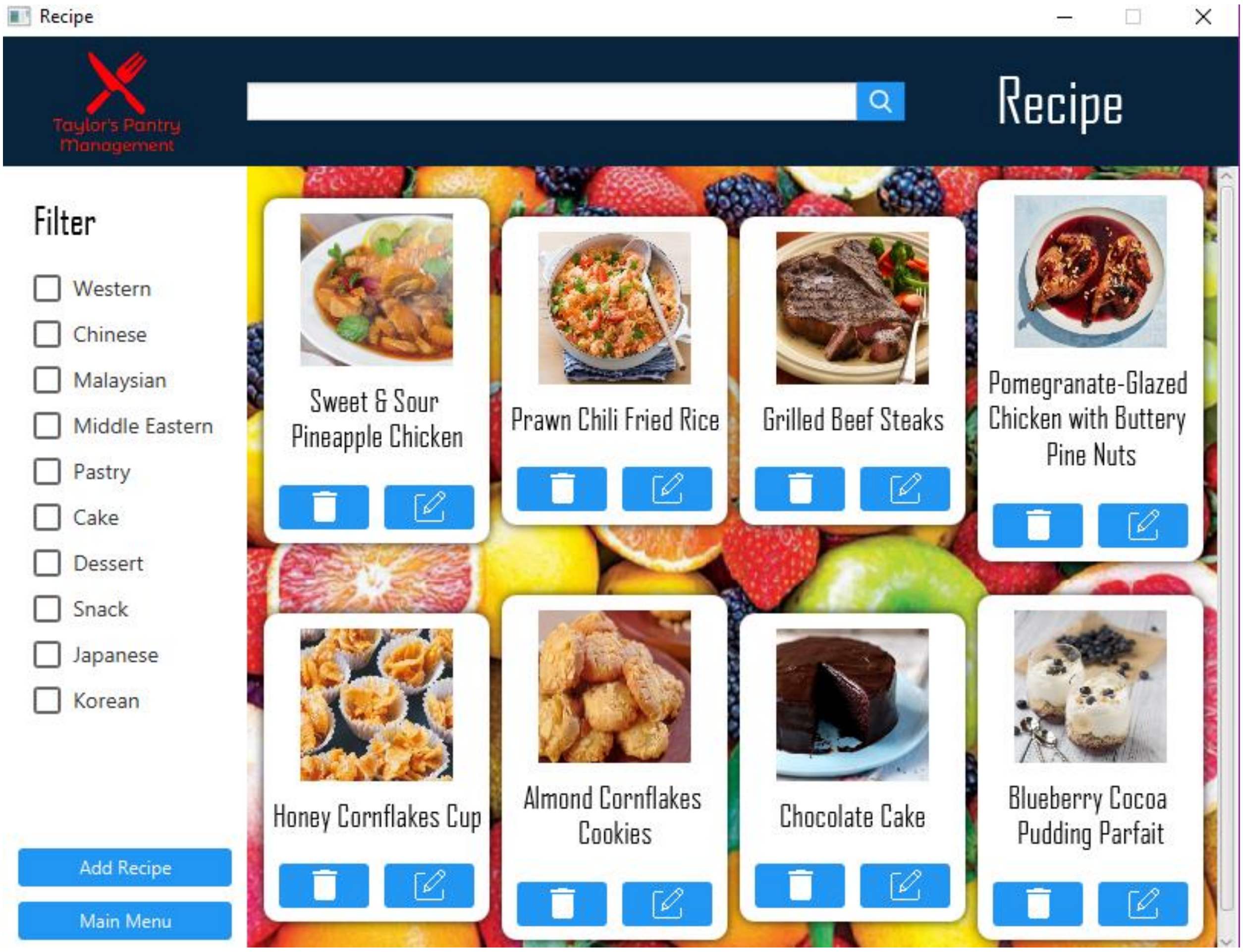
The Utensil page contains a list of utensils recorded by the system, which is displayed in grid view by default.

Ingredient



The Ingredient page contains all ingredients in the system. They are shown in grid view by default.

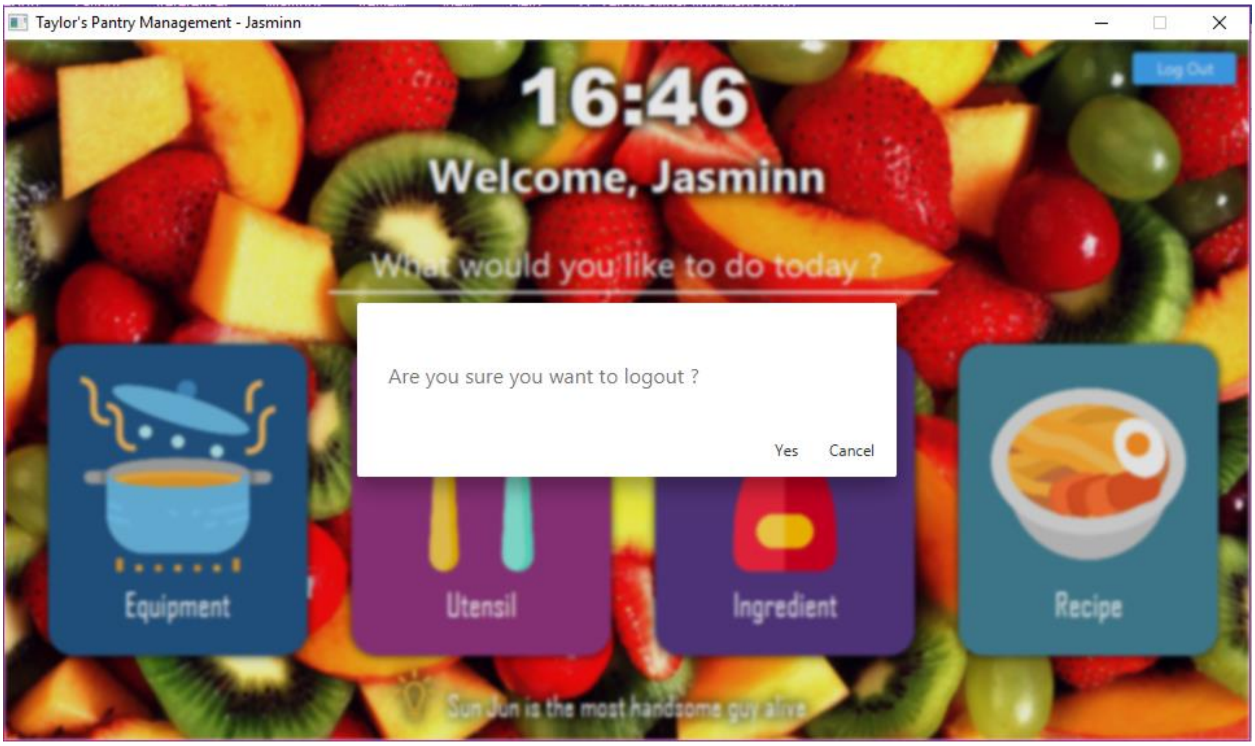
Recipe



The Recipe page contains a list of recipes that the staff can use as guidelines to make food.

Log Out

If the user is finished with what they want to do, they can click the ‘Log Out’ button on the top right-hand corner of the main menu. They will be asked to confirm their decision.



Welcome, Jasminn

What would you like to do today ?

Are you sure you want to logout ?

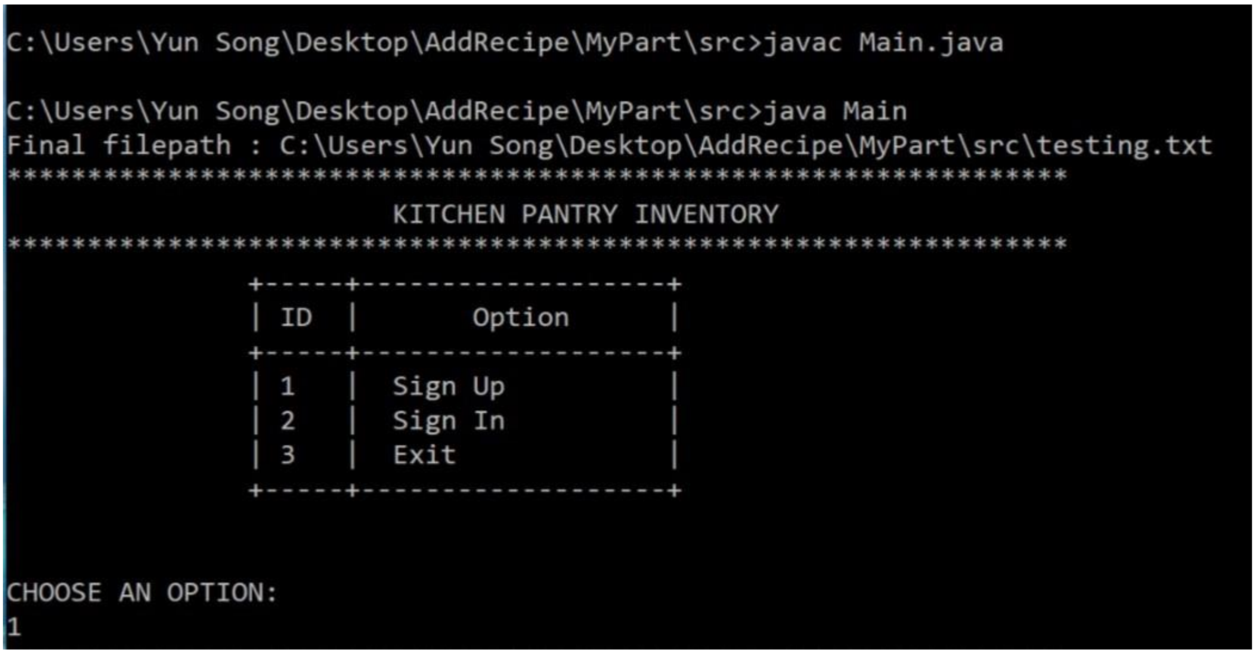
Sun Jun is the most handsome guy alive

After the user has logged out, they will need to login again to access the system.

OOP1 Program vs Current Program

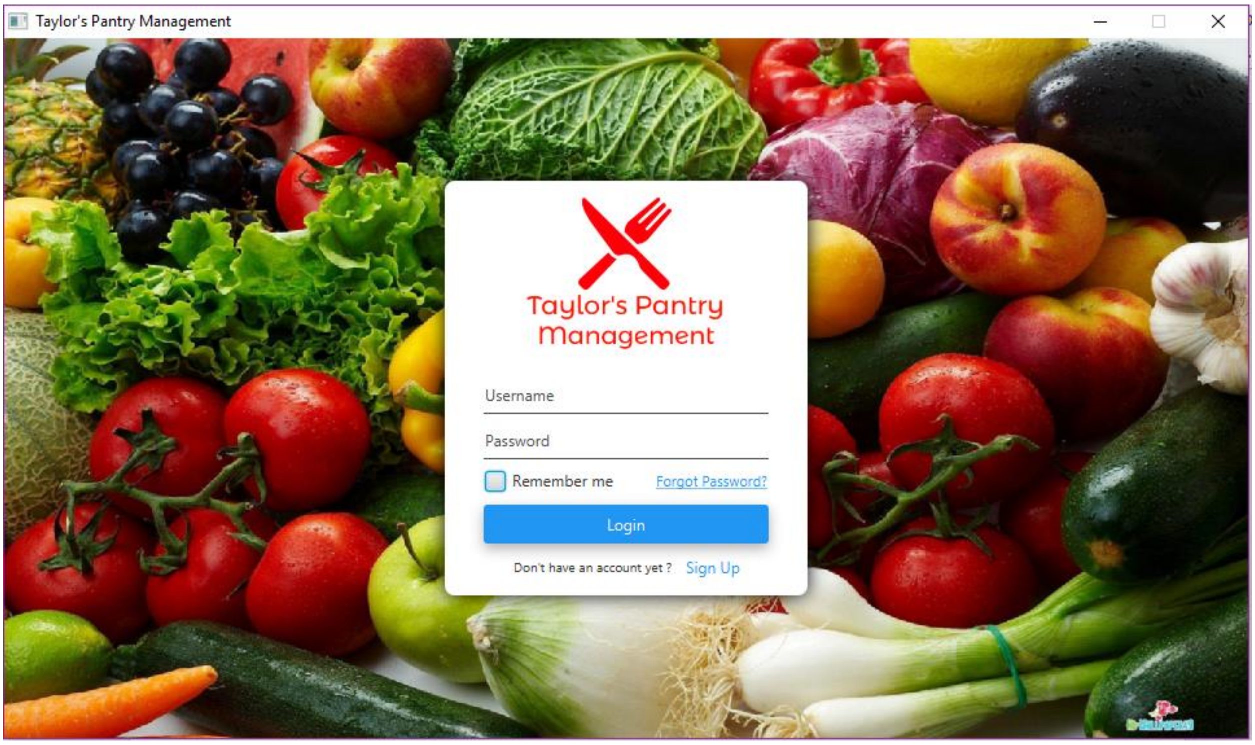
Login

OOP1 Program:



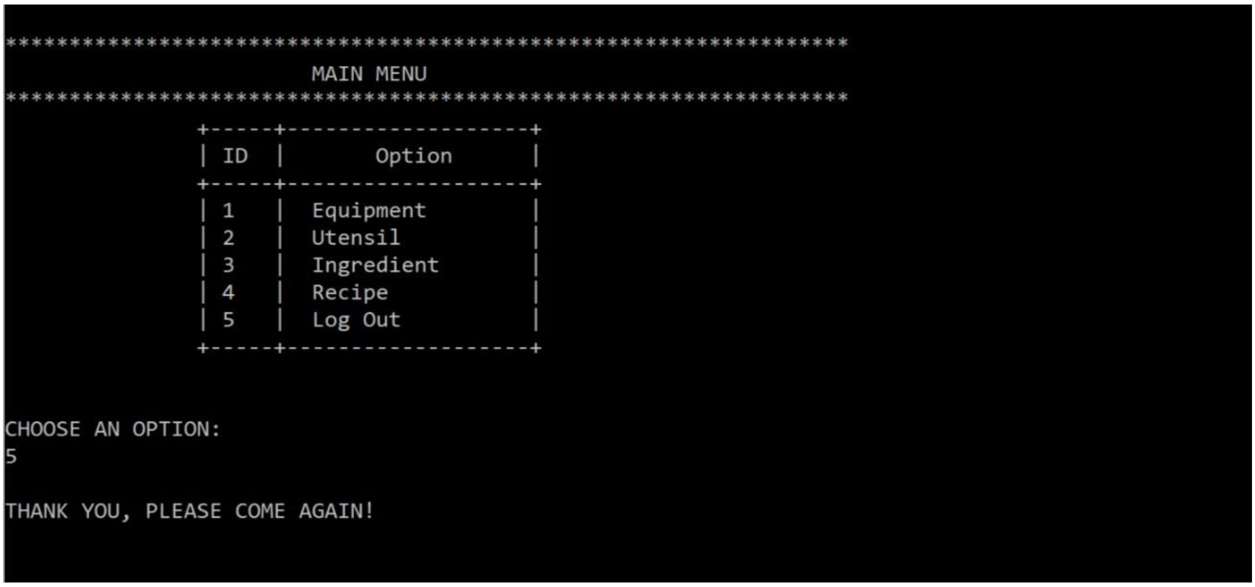
```
C:\Users\Yun Song\Desktop\AddRecipe\MyPart\src>javac Main.java

C:\Users\Yun Song\Desktop\AddRecipe\MyPart\src>java Main
Final filepath : C:\Users\Yun Song\Desktop\AddRecipe\MyPart\src\testing.txt
*******************************************************************
                        KITCHEN PANTRY INVENTORY
*******************************************************************

               +-----+------------------+
               | ID  |      Option      |
               +-----+------------------+
               | 1   |  Sign Up         |
               | 2   |  Sign In         |
               | 3   |  Exit            |
               +-----+------------------+


CHOOSE AN OPTION:
1
```

Current Program:


Taylor's Pantry Management
Taylor's Pantry Management
Username
Password
□ Remember me
Forgot Password?
Login
Don't have an account yet ?
Sign Up

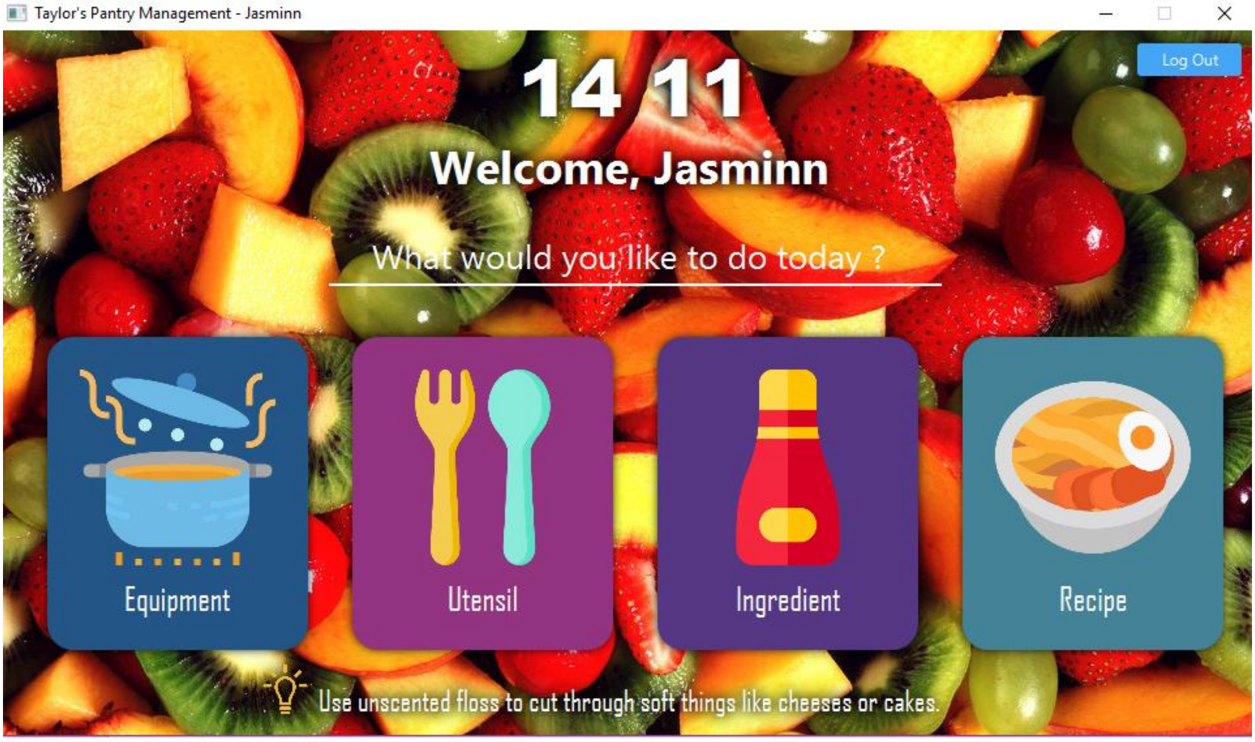
Main Menu

OOP1 Program:



```
******************************************************************
                        MAIN MENU
******************************************************************
               +-----+-------------------+
               | ID  |       Option      |
               +-----+-------------------+
               | 1   |  Equipment        |
               | 2   |  Utensil          |
               | 3   |  Ingredient       |
               | 4   |  Recipe           |
               | 5   |  Log Out          |
               +-----+-------------------+


CHOOSE AN OPTION:
5

THANK YOU, PLEASE COME AGAIN!
```

Current Program:



Taylor's Pantry Management - Jasminn

14 11

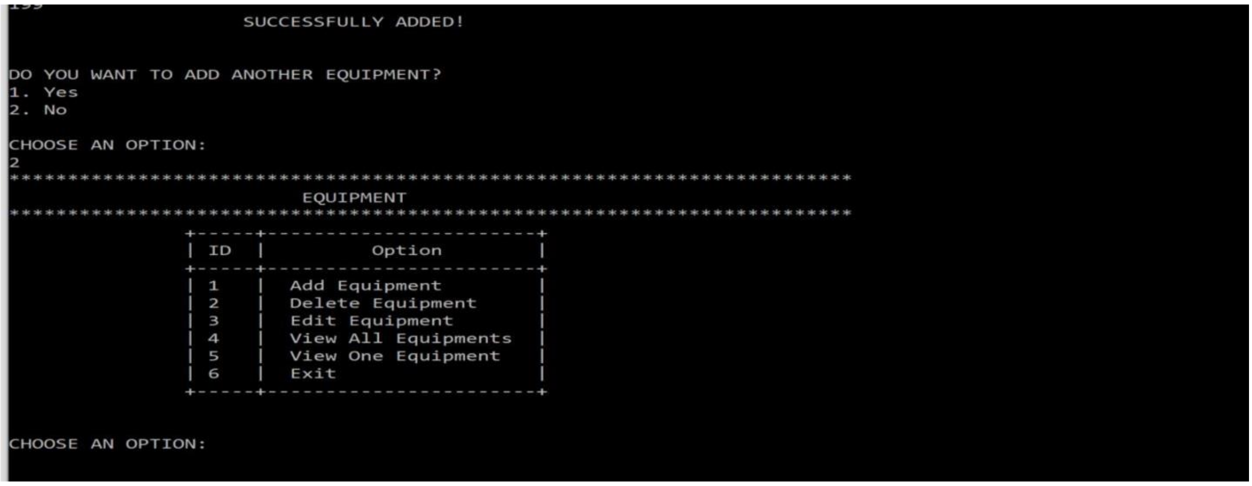
Welcome, Jasminn

What would you like to do today ?

Use unscented floss to cut through soft things like cheeses or cakes.

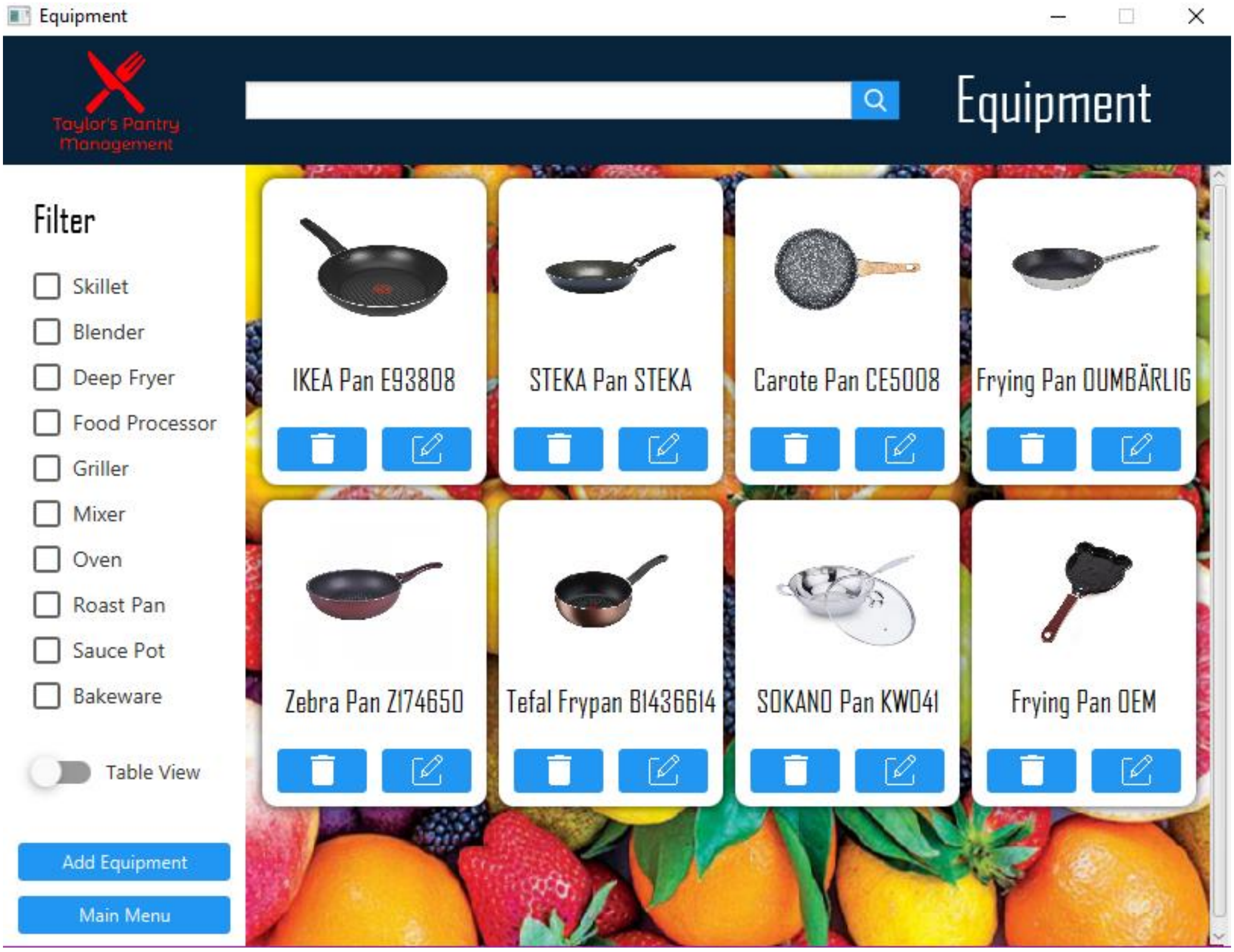
Equipment

OOP1 Program:



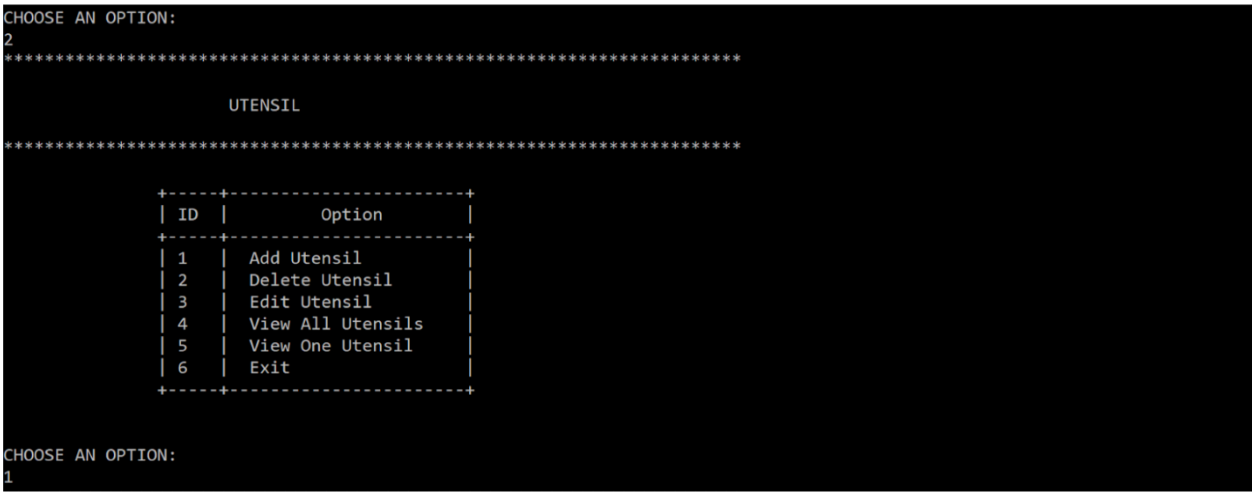
```
199
                    SUCCESSFULLY ADDED!

DO YOU WANT TO ADD ANOTHER EQUIPMENT?
1. Yes
2. No

CHOOSE AN OPTION:
2
************************************************************************
                         EQUIPMENT
************************************************************************
               +-----+-----------------------+
               | ID  |        Option         |
               +-----+-----------------------+
               | 1   |  Add Equipment        |
               | 2   |  Delete Equipment     |
               | 3   |  Edit Equipment       |
               | 4   |  View All Equipments  |
               | 5   |  View One Equipment   |
               | 6   |  Exit                 |
               +-----+-----------------------+

CHOOSE AN OPTION:
```

Current Program:



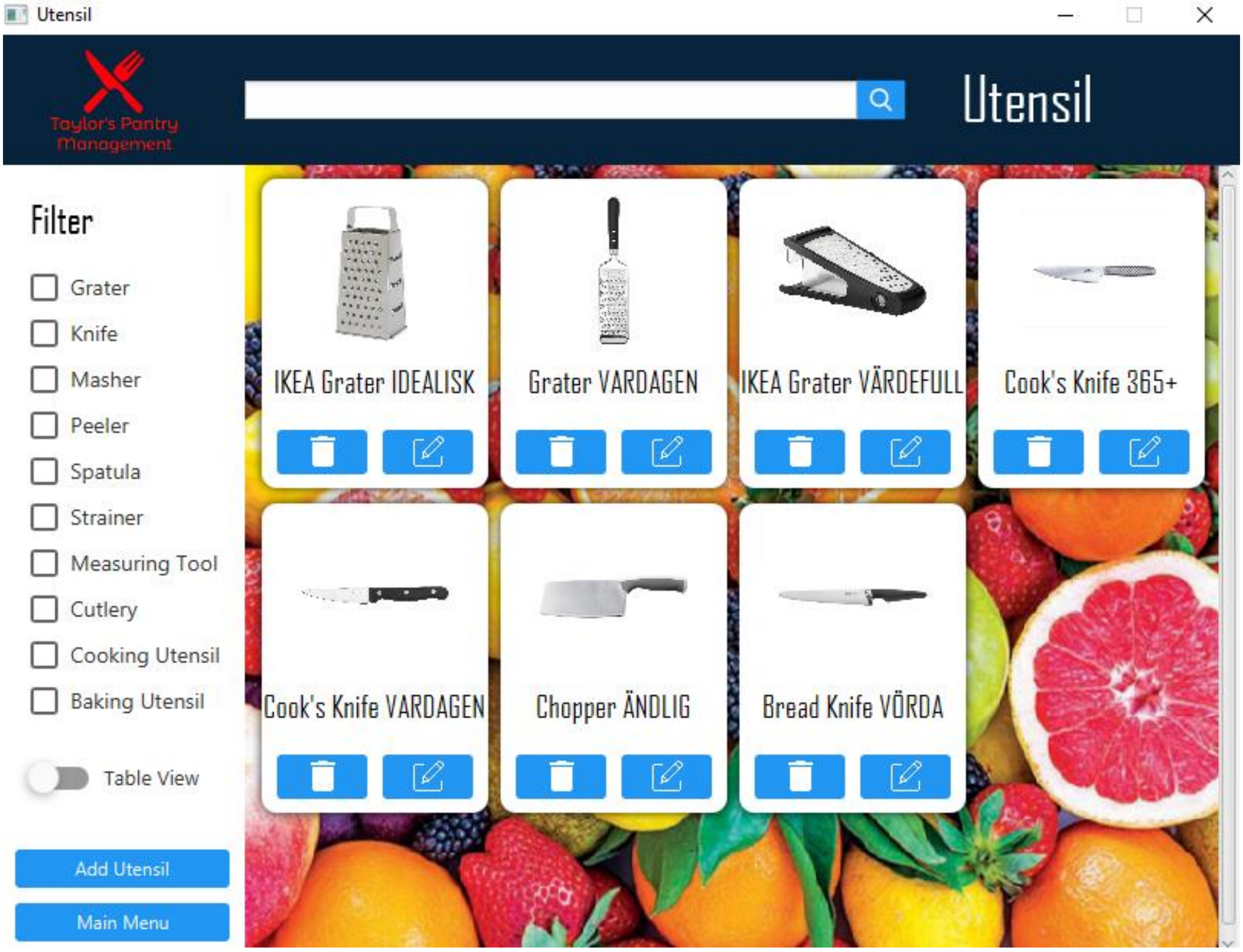
Utensil

OOP1 Program:



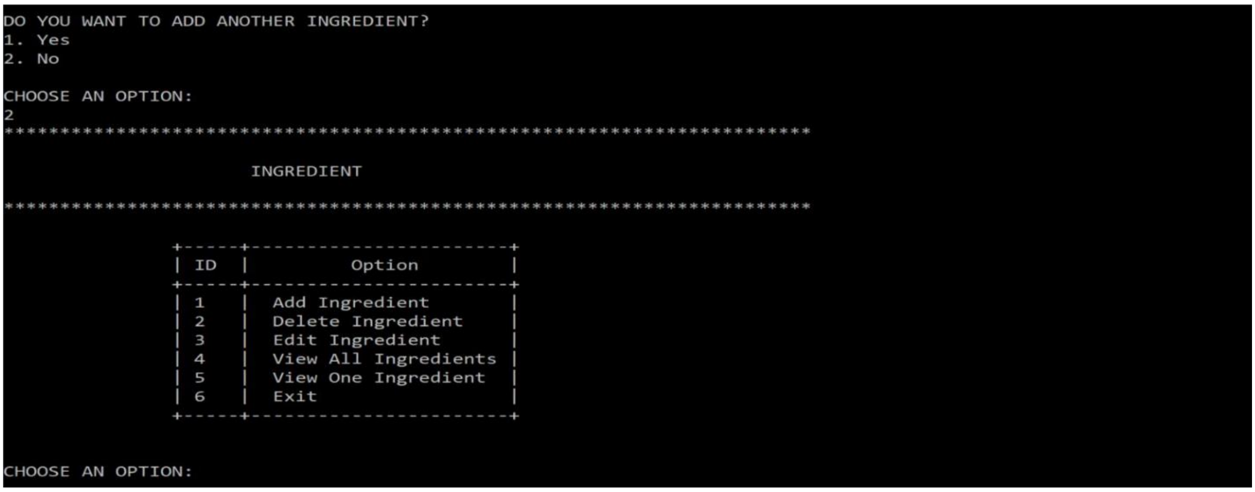
```
CHOOSE AN OPTION:
2
*************************************************************************

                      UTENSIL

*************************************************************************

               +-----+----------------------+
               | ID  |        Option        |
               +-----+----------------------+
               | 1   |  Add Utensil         |
               | 2   |  Delete Utensil      |
               | 3   |  Edit Utensil        |
               | 4   |  View All Utensils   |
               | 5   |  View One Utensil    |
               | 6   |  Exit                |
               +-----+----------------------+


CHOOSE AN OPTION:
1
```

Current Program:



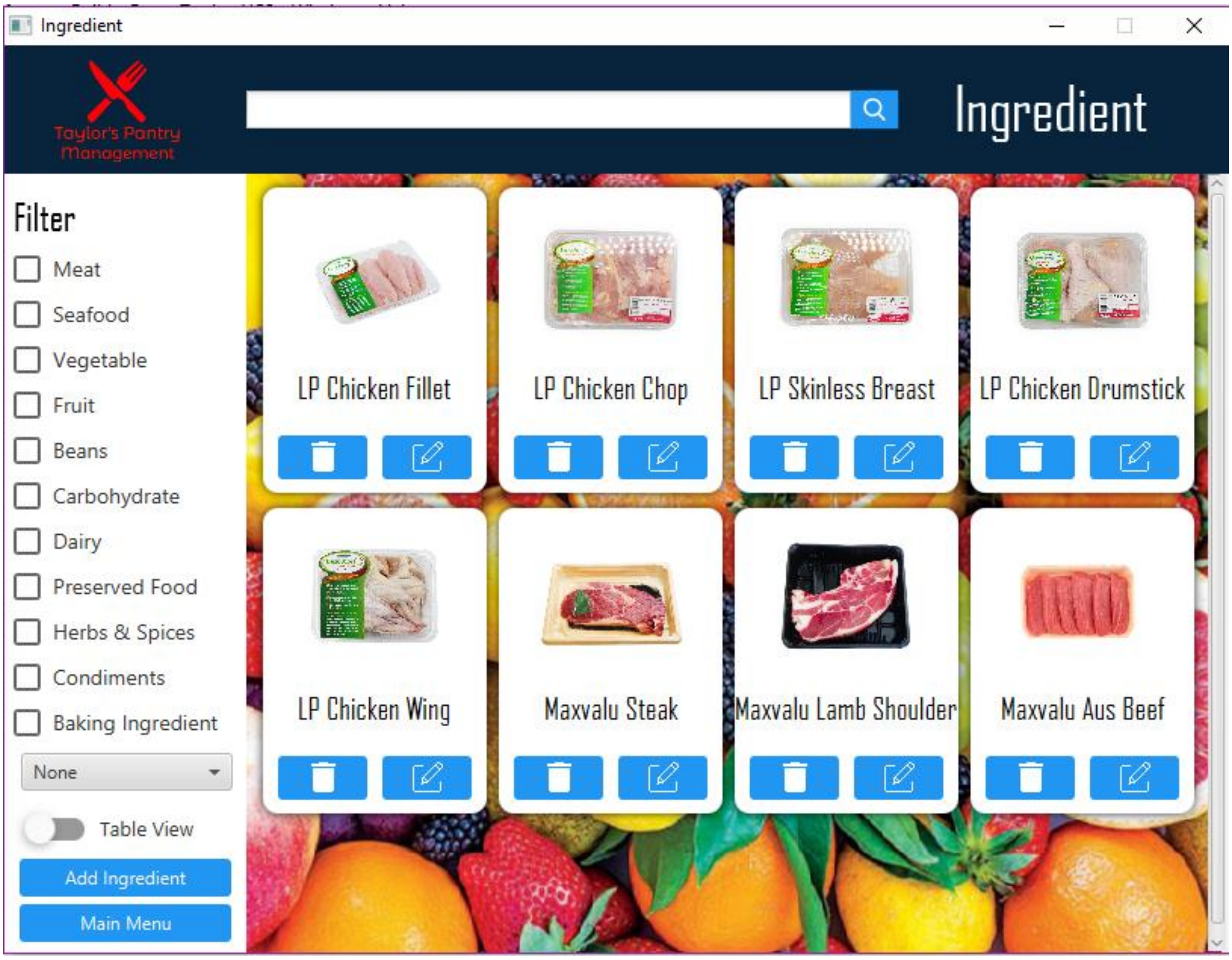
Ingredient

OOP1 Program:



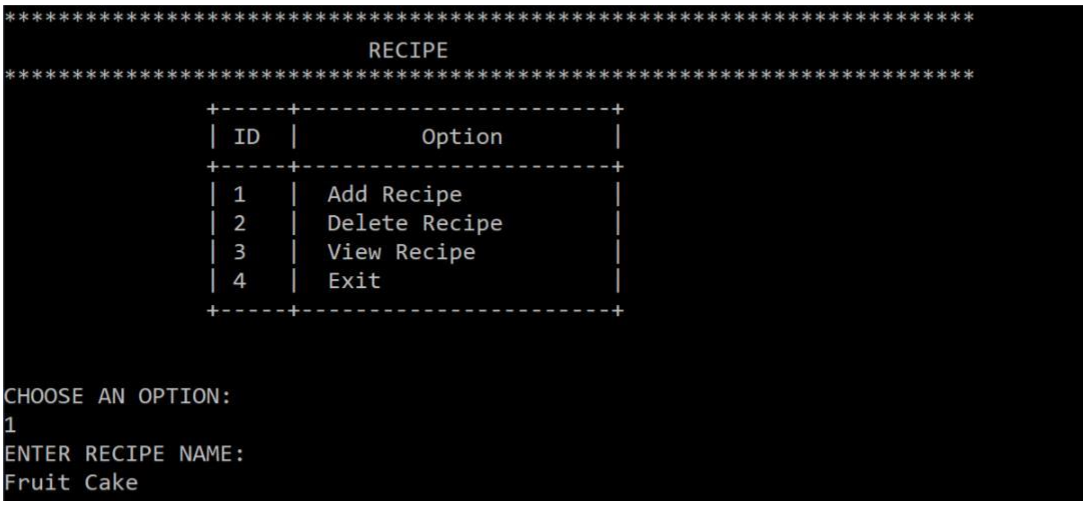
```
DO YOU WANT TO ADD ANOTHER INGREDIENT?
1. Yes
2. No

CHOOSE AN OPTION:
2
************************************************************************

                      INGREDIENT

************************************************************************

               +-----+---------------------+
               | ID  |       Option        |
               +-----+---------------------+
               | 1   |  Add Ingredient     |
               | 2   |  Delete Ingredient  |
               | 3   |  Edit Ingredient    |
               | 4   |  View All Ingredients |
               | 5   |  View One Ingredient |
               | 6   |  Exit               |
               +-----+---------------------+

CHOOSE AN OPTION:
```

Current Program:



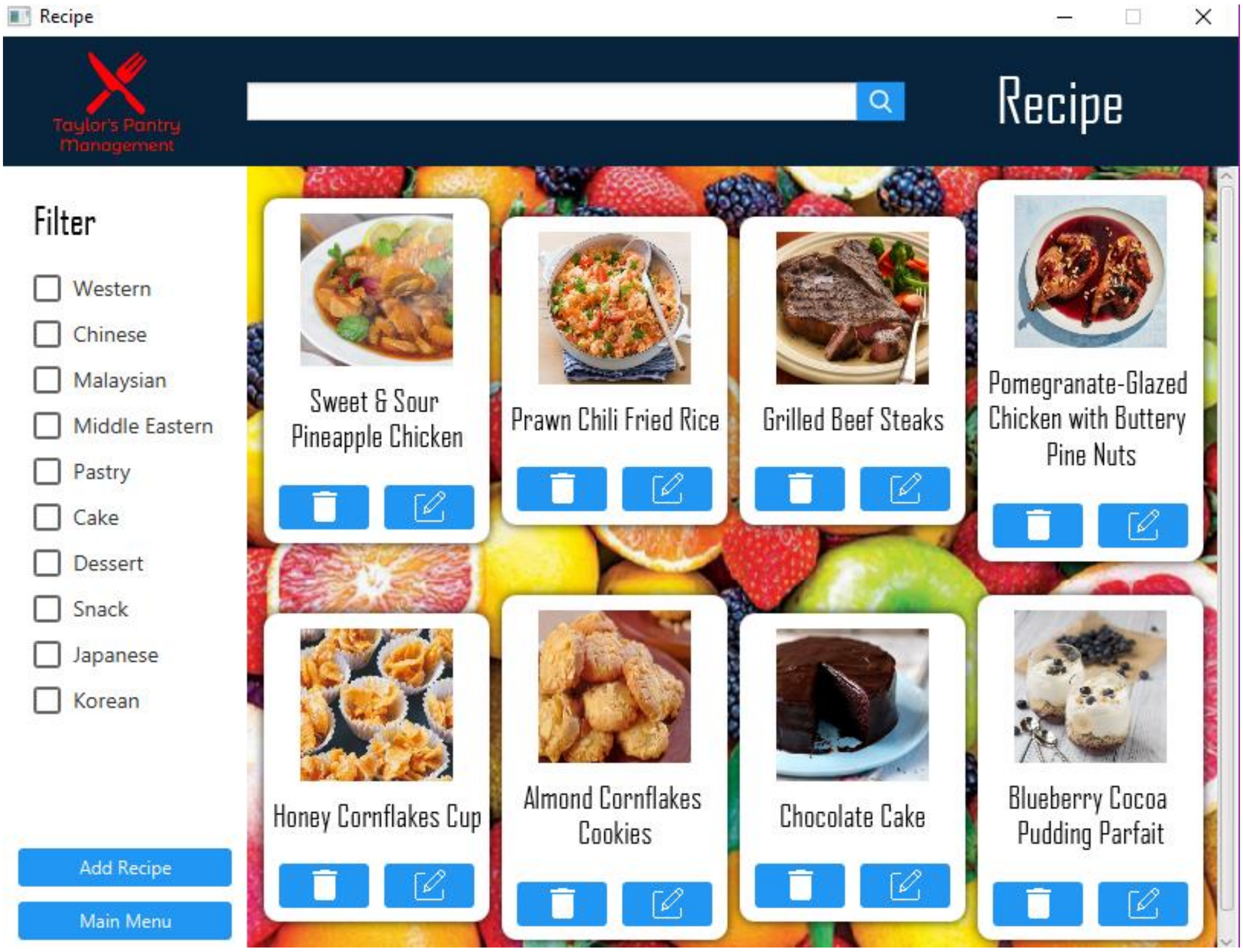
Recipe

OOP1 Program:



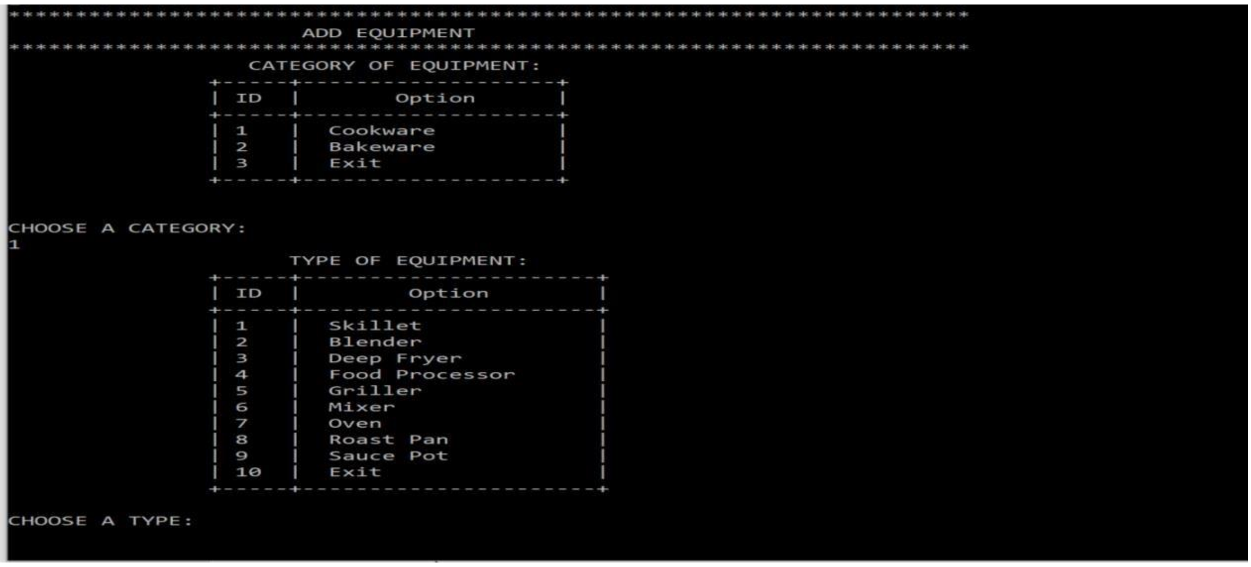
```
************************************************************************
                           RECIPE
************************************************************************
               +-----+----------------------+
               | ID  |        Option        |
               +-----+----------------------+
               | 1   |  Add Recipe          |
               | 2   |  Delete Recipe       |
               | 3   |  View Recipe         |
               | 4   |  Exit                |
               +-----+----------------------+


CHOOSE AN OPTION:
1
ENTER RECIPE NAME:
Fruit Cake
```

Current Program:

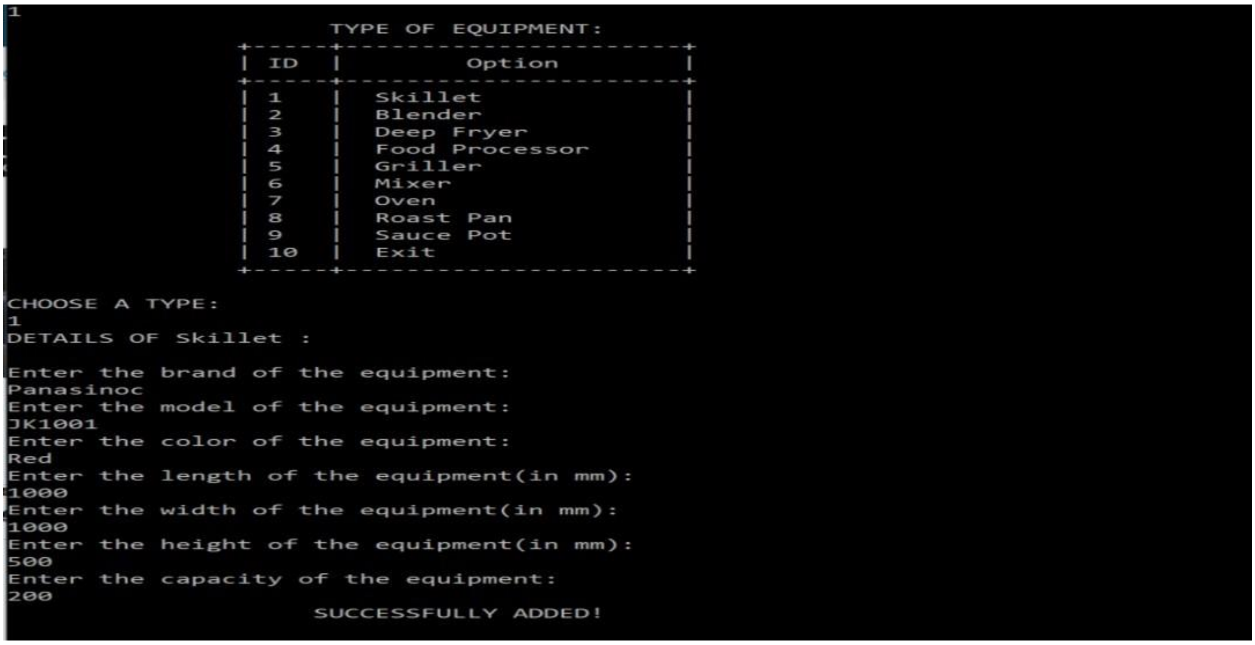


Adding New Data (Equipment)

OOP1 Program:

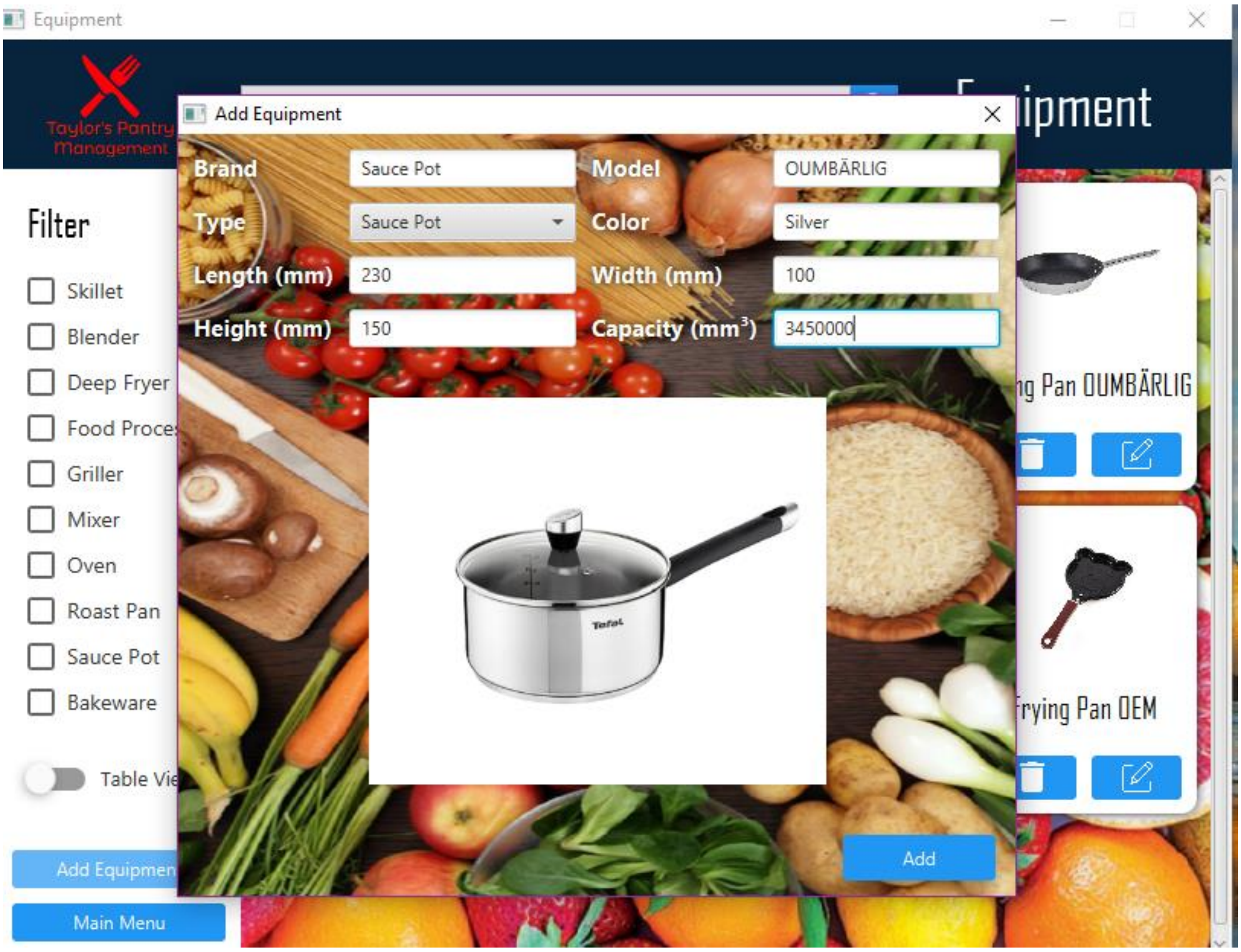


```
************************************************************************
                      ADD EQUIPMENT
************************************************************************
                  CATEGORY OF EQUIPMENT:
               +-----+-------------------+
               | ID  |       Option      |
               +-----+-------------------+
               | 1   |   Cookware        |
               | 2   |   Bakeware        |
               | 3   |   Exit            |
               +-----+-------------------+


CHOOSE A CATEGORY:
1
                     TYPE OF EQUIPMENT:
               +-----+----------------------+
               | ID  |        Option        |
               +-----+----------------------+
               | 1   |   Skillet            |
               | 2   |   Blender            |
               | 3   |   Deep Fryer         |
               | 4   |   Food Processor     |
               | 5   |   Griller            |
               | 6   |   Mixer              |
               | 7   |   Oven               |
               | 8   |   Roast Pan          |
               | 9   |   Sauce Pot          |
               | 10  |   Exit               |
               +-----+----------------------+

CHOOSE A TYPE:
```



```
1
                     TYPE OF EQUIPMENT:
               +-----+----------------------+
               | ID  |        Option        |
               +-----+----------------------+
               | 1   |   Skillet            |
               | 2   |   Blender            |
               | 3   |   Deep Fryer         |
               | 4   |   Food Processor     |
               | 5   |   Griller            |
               | 6   |   Mixer              |
               | 7   |   Oven               |
               | 8   |   Roast Pan          |
               | 9   |   Sauce Pot          |
               | 10  |   Exit               |
               +-----+----------------------+

CHOOSE A TYPE:
1
DETAILS OF Skillet :

Enter the brand of the equipment:
Panasinoc
Enter the model of the equipment:
JK1001
Enter the color of the equipment:
Red
Enter the length of the equipment(in mm):
1000
Enter the width of the equipment(in mm):
1000
Enter the height of the equipment(in mm):
500
Enter the capacity of the equipment:
200
                    SUCCESSFULLY ADDED!
```

Current Program:



Add Equipment

Brand: Sauce Pot

Model: OUMBÄRLIG

Type: Sauce Pot

Color: Silver

Length (mm): 230

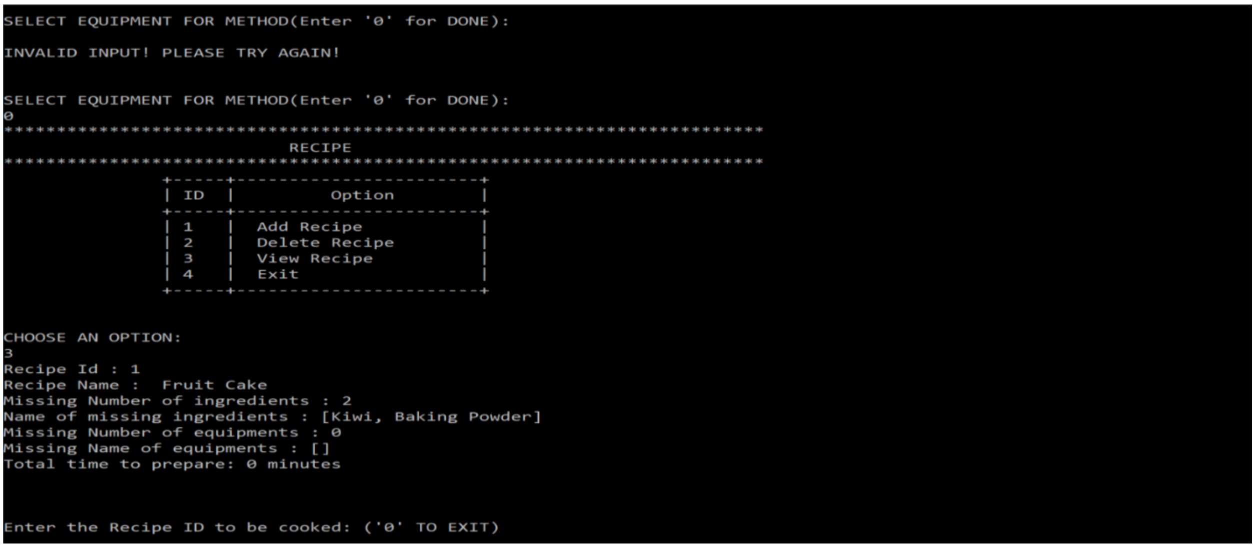
Width (mm): 100

Height (mm): 150

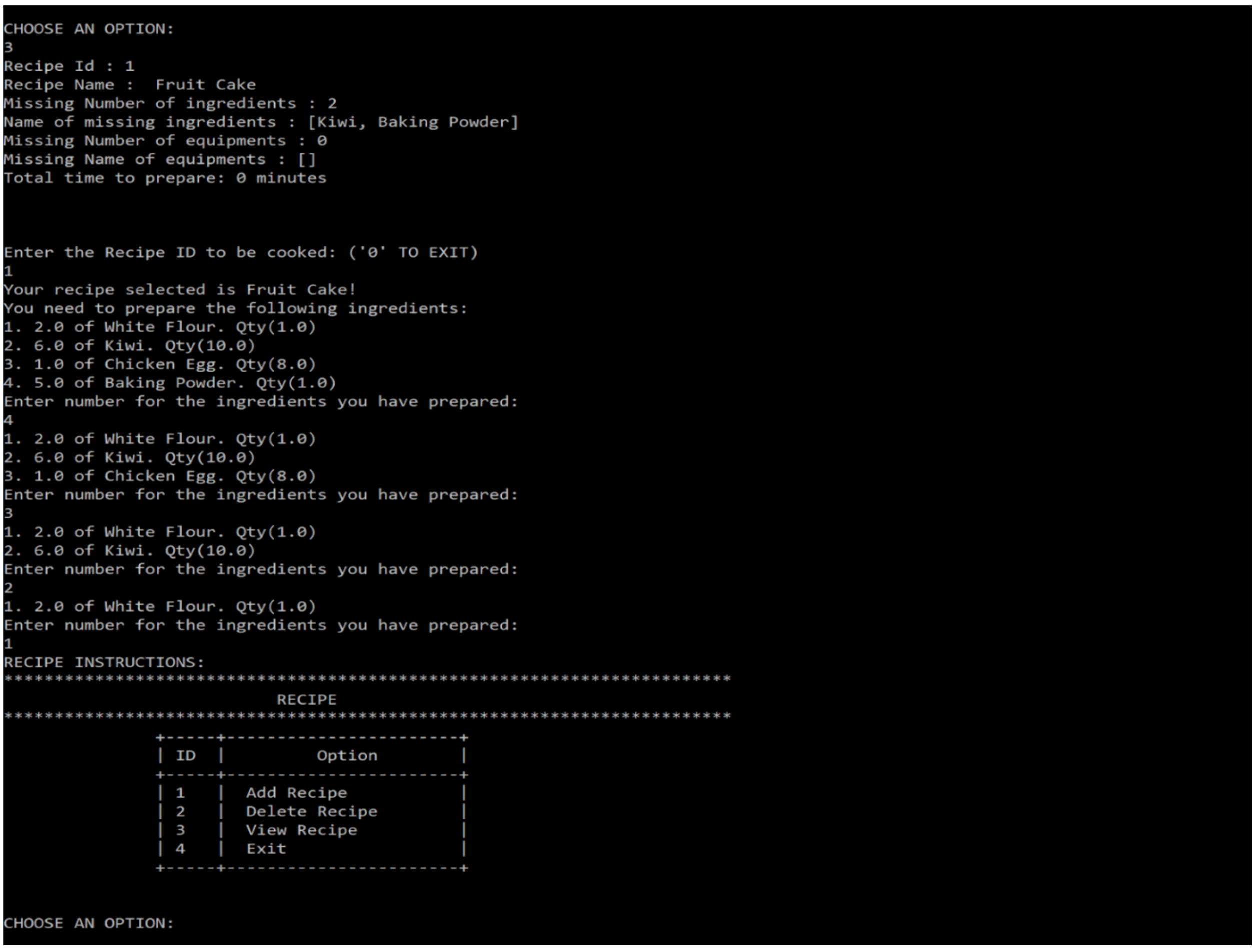
Capacity (mm^3): 3450000

View Recipe

OOP1 Program:

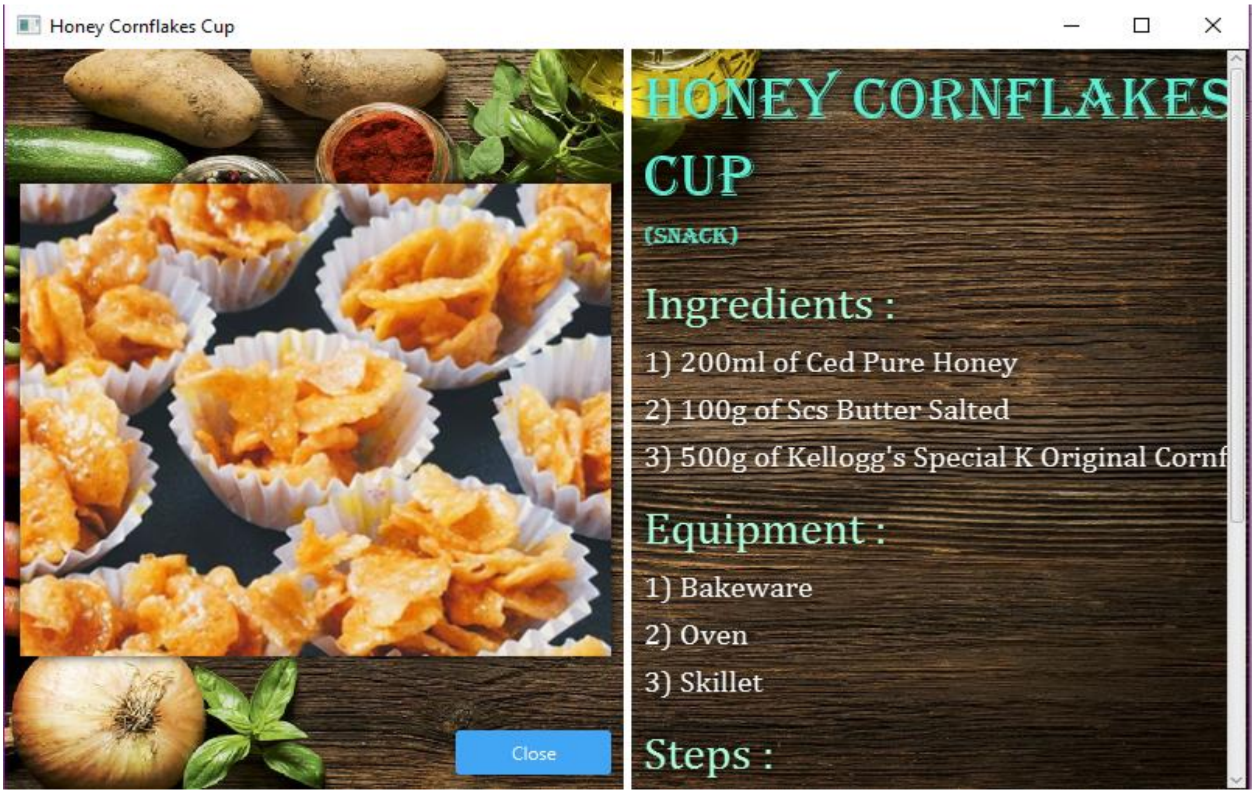


```
SELECT EQUIPMENT FOR METHOD(Enter '0' for DONE):

INVALID INPUT! PLEASE TRY AGAIN!


SELECT EQUIPMENT FOR METHOD(Enter '0' for DONE):
0
*************************************************************************
                           RECIPE
*************************************************************************
               +-----+-----------------------+
               | ID  |         Option        |
               +-----+-----------------------+
               | 1   |  Add Recipe           |
               | 2   |  Delete Recipe        |
               | 3   |  View Recipe          |
               | 4   |  Exit                 |
               +-----+-----------------------+


CHOOSE AN OPTION:
3
Recipe Id : 1
Recipe Name :  Fruit Cake
Missing Number of ingredients : 2
Name of missing ingredients : [Kiwi, Baking Powder]
Missing Number of equipments : 0
Missing Name of equipments : []
Total time to prepare: 0 minutes



Enter the Recipe ID to be cooked: ('0' TO EXIT)
```



```
CHOOSE AN OPTION:
3
Recipe Id : 1
Recipe Name :  Fruit Cake
Missing Number of ingredients : 2
Name of missing ingredients : [Kiwi, Baking Powder]
Missing Number of equipments : 0
Missing Name of equipments : []
Total time to prepare: 0 minutes



Enter the Recipe ID to be cooked: ('0' TO EXIT)
1
Your recipe selected is Fruit Cake!
You need to prepare the following ingredients:
1. 2.0 of White Flour. Qty(1.0)
2. 6.0 of Kiwi. Qty(10.0)
3. 1.0 of Chicken Egg. Qty(8.0)
4. 5.0 of Baking Powder. Qty(1.0)
Enter number for the ingredients you have prepared:
4
1. 2.0 of White Flour. Qty(1.0)
2. 6.0 of Kiwi. Qty(10.0)
3. 1.0 of Chicken Egg. Qty(8.0)
Enter number for the ingredients you have prepared:
3
1. 2.0 of White Flour. Qty(1.0)
2. 6.0 of Kiwi. Qty(10.0)
Enter number for the ingredients you have prepared:
2
1. 2.0 of White Flour. Qty(1.0)
Enter number for the ingredients you have prepared:
1
RECIPE INSTRUCTIONS:
************************************************************************
                           RECIPE
************************************************************************
               +-----+-----------------------+
               | ID  |         Option        |
               +-----+-----------------------+
               | 1   |  Add Recipe           |
               | 2   |  Delete Recipe        |
               | 3   |  View Recipe          |
               | 4   |  Exit                 |
               +-----+-----------------------+


CHOOSE AN OPTION:
```

Current Program:



HONEY CORNFLAKES CUP

(SNACK)

Ingredients :

1) 200ml of Ced Pure Honey

2) 100g of Scs Butter Salted

3) 500g of Kellogg's Special K Original Cornf

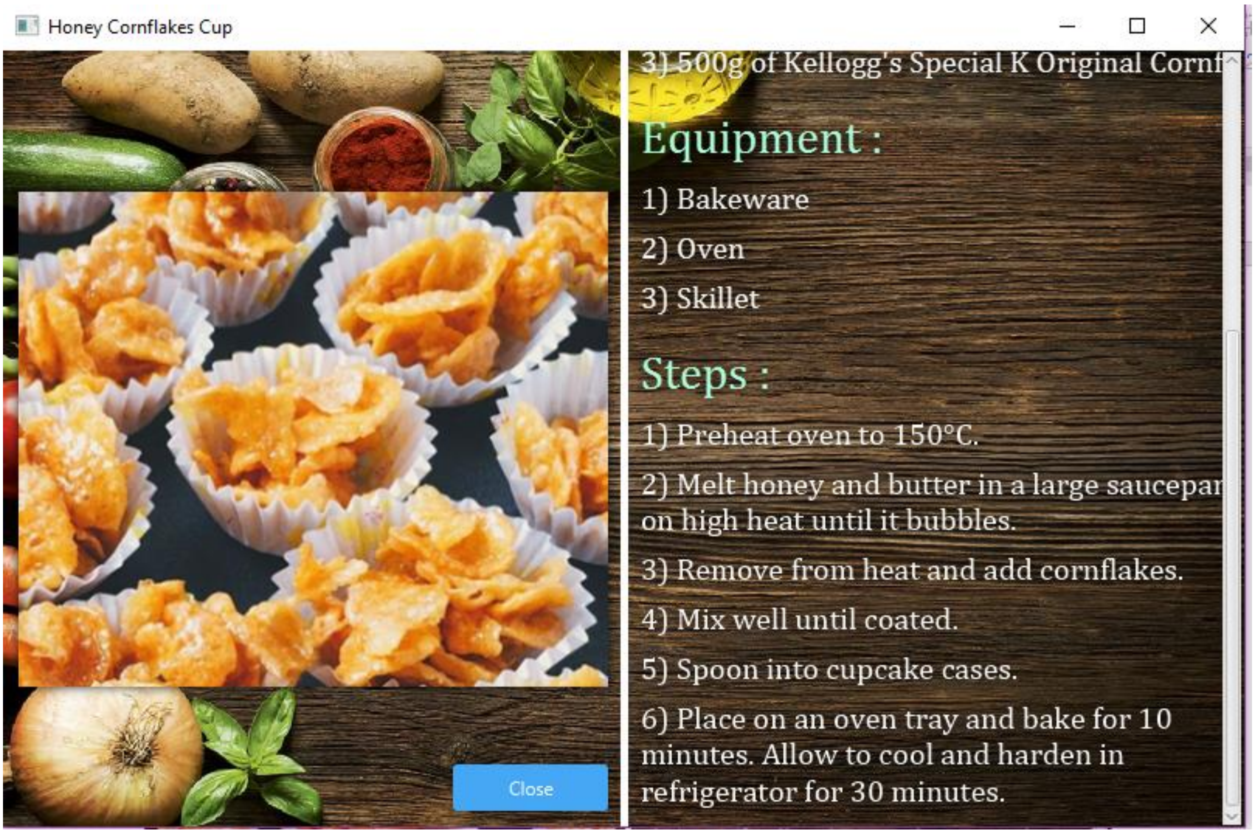
Equipment :

1) Bakeware

2) Oven

3) Skillet

Steps :



3) 500g of Kellogg's Special K Original Cornf

Equipment :

1) Bakeware

2) Oven

3) Skillet

Steps :

1) Preheat oven to 150°C.

2) Melt honey and butter in a large saucepan on high heat until it bubbles.

3) Remove from heat and add cornflakes.

4) Mix well until coated.

5) Spoon into cupcake cases.

6) Place on an oven tray and bake for 10 minutes. Allow to cool and harden in refrigerator for 30 minutes.

Problems encountered

1. We have decided to develop this program based on the program that we did in OOP1. The problem is that the program that we did back in OOP1 was very complicated since it consists of many classes and methods that are linked to one another. However, we managed to solve this problem by analyzing the program and removing redundant or unnecessary classes and methods which we managed to cut down from around 30 classes to just 4 main classes.

2. Since the overall design of the program must be consistent, it was hard to distribute work since different people have different taste in design. Thus, what we had done was everyone was asked to come up with his or her own design first and then we discussed and thoroughly analyzed the good and bad in each design to come up with the perfect design in a consensual way.

3. There were some technical issues here there during the development of this program. For example, we've been struggling with trying to get the JavaFX works on our machine but to no avail. It turns out that some of us had JDK 12 installed which does not come prebuilt with JavaFX. This problem was solved by simply installing an older yet more stable version of JDK which in our opinion is 1.8 . Also, due to everyone was working on different machines, the layout was not consistent because of difference in the display resolution. After some research online, we managed to solve it by enabling auto-scaling in our IDE (IntelliJ).