

Gigabyte

Naimah-Joy Chapman
05/02/2022

1 Introduction

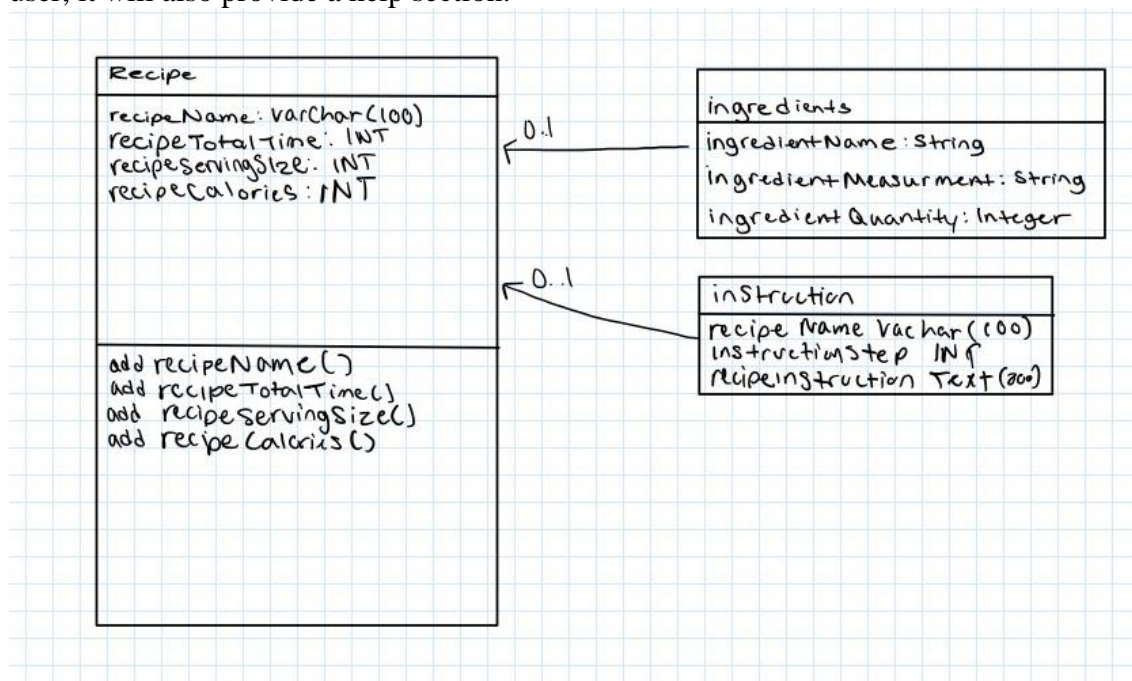
This program is for personal use for the organization of recipes. The system must be used with a Java IDE such as Eclipse as well as use GUI features. The data that will be stored within this project is recipe name, total time in minutes, serving size, calories, ingredients, and instructions. This program will allow you to create, update, review, and delete each recipe. There will be a help section to help clarify any confusion on what the use of each button is. A demonstration video is also attached so the user can better understand how the program works. Deliverables 1, 2, 3, and 4 were updated to reflect any changes to implementation and feedback given.

2 Deliverable 1: Project Description:

Many families have recipes that they pass down generation to generation, unfortunately a lot of these recipes get lost. Even when the recipes get written down it can easily be lost, discarded, or become hard to read. A digital recipe book would make it easier to create, save, delete, share, and revise your personal recipes.

2.1 User Classes:

This will be a program that is easy to use so that the user will not have to have an extensive knowledge of databases. Once the program has started it will appear to the user as a normal application for easy use. It will have a simple design as to not confuse the user, it will also provide a help section.



2.2 Stored Information:

The main data that needs to be stored is the name of the recipe. This is an important part because without the name of the recipe it would make it hard to find the recipe. The serving size and preparation time is also important data that will help the user determine if the recipe is something they can achieve. The directions and ingredient name are also an important data that needs to be stored, without them the user won't know how to create the recipe. ingredient measurement and ingredient quantity are data that doesn't always need to be stored for every recipe.

2.3 Database Operations Needed:

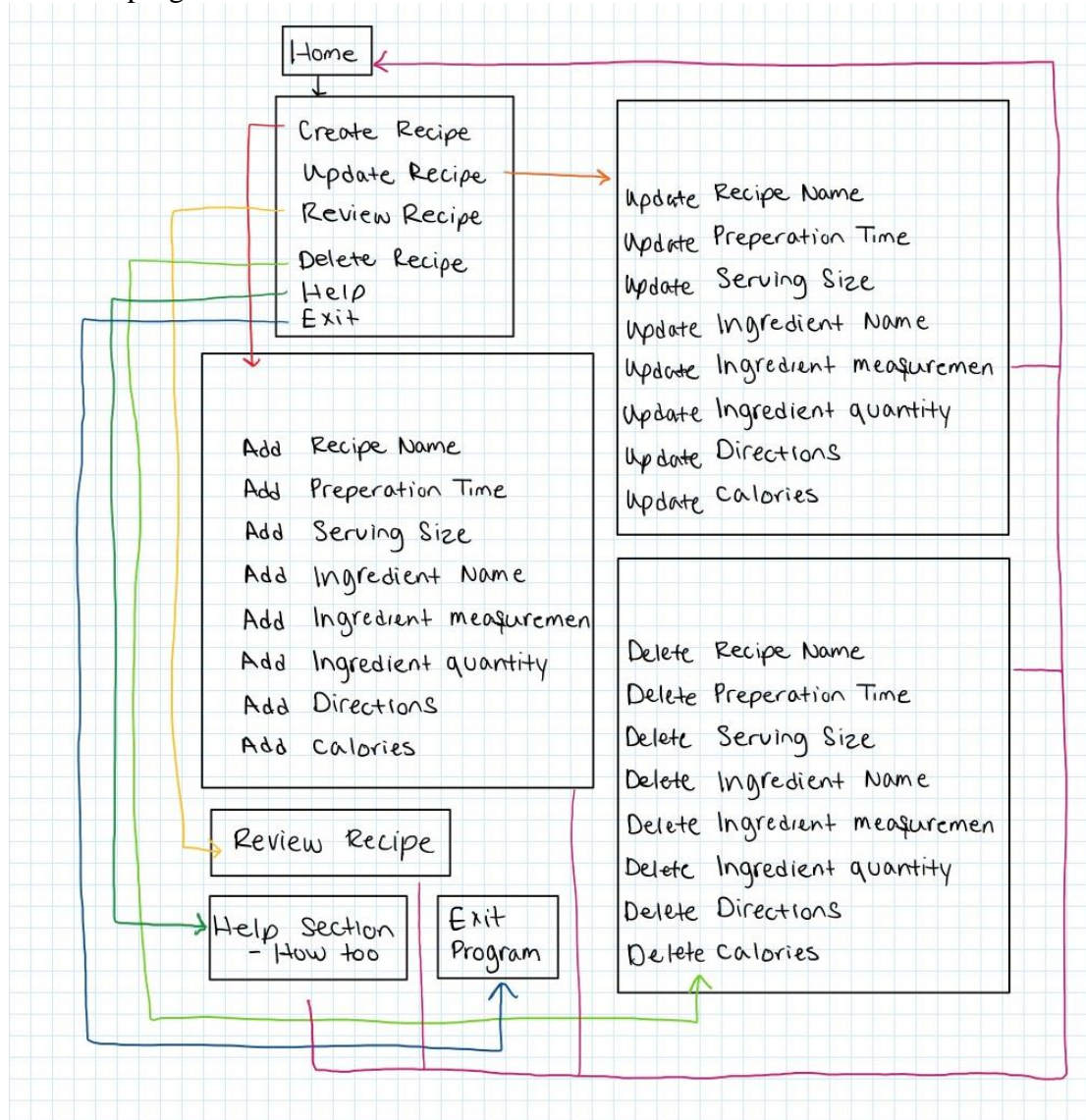
The user shall be able to create recipes

The user shall be able to update recipes
The user shall be able to review recipes
The user shall be able to delete recipes
The user shall be able to create ingredient name
The user shall be able to update ingredient name
The user shall be able to review ingredient name
The user shall be able to delete ingredient name
The user shall be able to create ingredient measurement
The user shall be able to update ingredient measurement
The user shall be able to review ingredient measurement
The user shall be able to delete ingredient measurement
The user shall be able to create ingredient quantity
The user shall be able to update ingredient quantity
The user shall be able to review ingredient quantity
The user shall be able to delete ingredient quantity
The user shall be able to create instructions
The user shall be able to update instructions
The user shall be able to review instructions
The user shall be able to delete instructions
The user shall be able to create recipe name
The user shall be able to update recipe name
The user shall be able to review recipe name
The user shall be able to delete recipe name
The user shall be able to create cooking time
The user shall be able to update cooking time
The user shall be able to review cooking time
The user shall be able to delete cooking time
The user shall be able to create serving size
The user shall be able to update serving size
The user shall be able to review serving size
The user shall be able to delete serving size
The user shall be able to open the help page
The user shall be able to exit the program

2.4 Mockup of Application

The user will start on the home screen which will have the buttons create recipe, update recipe, review recipe, delete recipe, help, and exit. When the user presses the create recipe button a window will pop up that will allow them to create a new recipe that will have recipe name, total time in minutes, serving size, calories, ingredients, and instructions. Once submitted it will bring the user back to the home screen. From there the user can review a recipe, this window will be a file directory. The user will pick which recipe they want to view, and it will pop up in a new window. Once the user is done it will bring them back to the home screen. There the user can choose to update any recipe by picking the update recipe button. Once pressed it will bring up a file directory and the user will pick which one, they want to update. There they can update the recipe

name, total time in minutes, serving size, calories, ingredients, and instructions. Once submitted it will bring them back to the home screen where they can then choose the delete button. This will allow the user to delete any recipe they want. The user can also check the help page for assistance on how to use the application or the exit button to close the program.

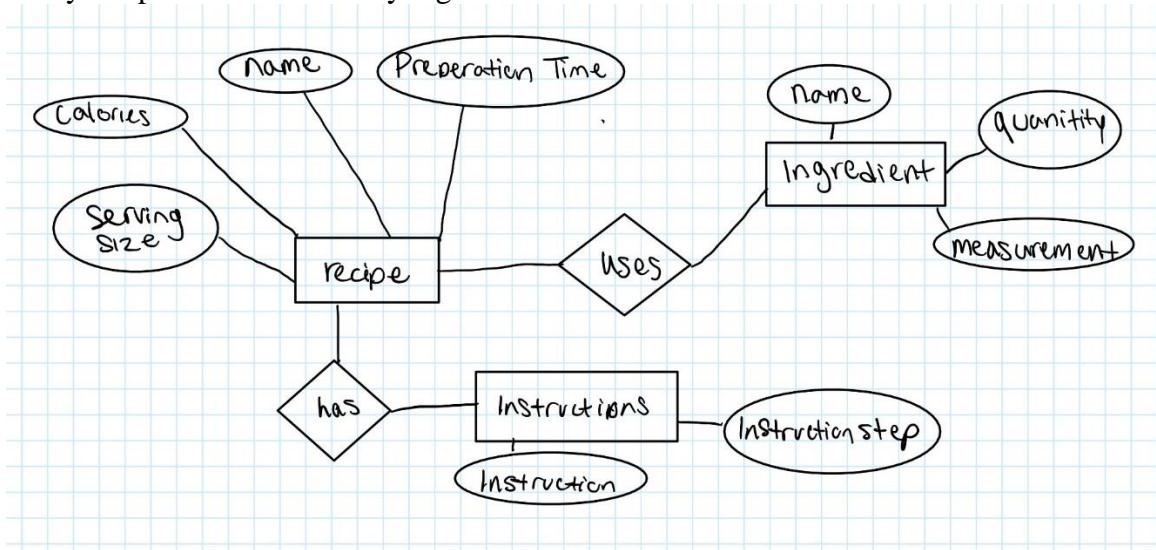


3 Deliverable 2: Design of Relations

In this section, the design of the relational database is described. First an E/R diagram models our problem. We translate the model into a relational schema. Analysis demonstrates that the schema is in 3rd normal form.

3.1 E/R Modeling

One recipe can only have one name, preparation time, calories or serving size listed. Every recipe can have as many ingredients or directions as needed.



3.2 Relational Schema

```
recipe(recipeName: VARCHAR(100), instruction: TEXT(800), recipeTotalTime: INT,
recipeServingSize: INT, recipeCalories: INT)
ingredient(ingredientName: VARCHAR(30), ingredientQuantity: INT,
ingredientMeasurement: VARCHAR(15))
instruction(instructionStep: INT, recipeInstruction: TEXT(800))
```

The design is simple enough that it is already in 3NF, there are no violations and there is no need to decompose.

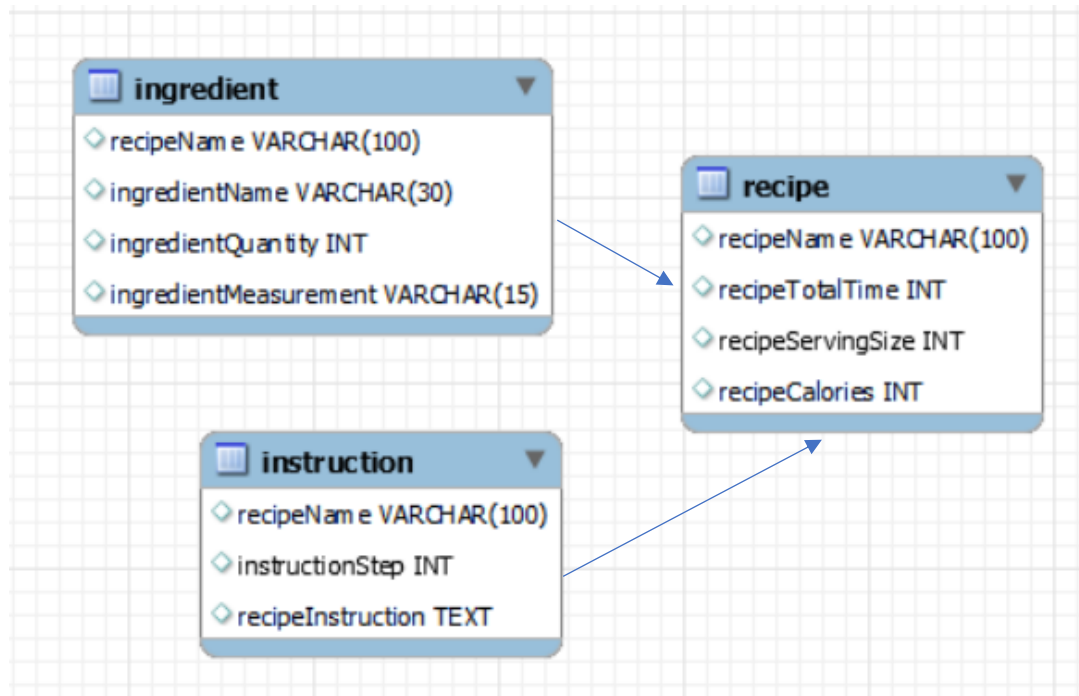
4 Deliverable 3: Implementation of Database

This section discusses the translation of the design into an actual database in SQL. It focuses upon the database exclusively and does not include detailed design/implementation of the application that utilizes the database.

4.1 Definition of SQL Database Schema

This section presents the SQL schema for the database designed in the previous section.

4.1.1 Database schema



4.1.2 Sample Data

```
-- recipeInstruction, instructionStep
34 • INSERT INTO instruction VALUES
35     ('Chocolate Pretzel', 1, 'Preheat oven to 175 degrees F ( 80 degrees C)'),
36     ('Chocolate Pretzel', 2, 'Arrange pretzels on a baking sheet. Place a candy kiss on the center of each pretzel'),
37     ('Chocolate Pretzel', 3, 'Warm pretzels in the preheated oven until candy kiss is shiny and slightly softened, 2 minutes'),
38     ('Chocolate Pretzel', 4, 'Place a candy-coated chocolate piece atop the candy kiss on each pretzel; press down. Chill in the refrigerator')
39
40 -- recipeName, ingredientName, ingredientQuantity, ingredientMeasurement
41 INSERT INTO ingredient VALUES
42     ('Chocolate Pretzel', 'Square Pretzels', 15, 'ounce'),
43     ('Chocolate Pretzel', 'Hersheys Kisses', 8, 'ounce'),
44     ('Chocolate Pretzel', 'M&M', 1.69, 'ounce'),
45     ('Chocolate Pretzel', 'Corn', 2, 'cob');
46
47
24
25 -----
26 -- Insert Data
27 -- recipeName, recipeTotalTime, recipeServingSize, recipeCalories
28 • INSERT INTO recipe VALUES
29     ('Chocolate Pretzel', 22, 40, 37),
30     ('Green Eggs and Ham', 15, 1, 210),
31     ('Beans and Cheese', 7, 2, 230),
32     ('One-pan Salmon with Roast Asparagus', 70, 2, 483);
33
34 -- recipeInstruction, instructionStep
35 • INSERT INTO instruction VALUES
36     ('Chocolate Pretzel', 1, 'Preheat oven to 175 degrees F ( 80 degrees C)'),
37     ('Chocolate Pretzel', 2, 'Arrange pretzels on a baking sheet. Place a candy kiss on the center of each pretzel'),
38     ('Chocolate Pretzel', 3, 'Warm pretzels in the preheated oven until candy kiss is shiny and slightly softened, 2 minutes'),
39     ('Chocolate Pretzel', 4, 'Place a candy-coated chocolate piece atop the candy kiss on each pretzel; press down. Chill in the refrigerator')
```

4.1.3 Schema Test Results

```
SELECT recipeName, recipeTotalTime, recipeServingSize, recipeCalories
FROM recipe
WHERE recipeCalories <= 300;
```

```
SELECT recipeName, instructionStep, recipeInstruction
FROM instruction
WHERE instructionStep <= 2;
```

```
SELECT recipeName, ingredientName, ingredientQuantity, ingredientMeasurement
FROM ingredient
WHERE ingredientMeasurement = 'cob';
```

#	Time	Action	Message	Duration / Fetch
97	17:56:10	INSERT INTO instruction VALUES (Chocolate Pretzel', 1, 'Preheat oven to 175 degrees F (80 degrees C)'), (C...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
98	17:56:10	INSERT INTO ingredient VALUES (Chocolate Pretzel', 'Square Pretzels', 15, 'ounce'), (Chocolate Pretzel', 'He...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
99	17:56:11	SELECT recipeName, recipeTotalTime, recipeServingSize, recipeCalories FROM recipe WHERE recipeC...	3 row(s) returned	0.000 sec / 0.000 sec
100	17:56:11	SELECT recipeName, instructionStep, recipeInstruction FROM instruction WHERE instructionStep <= 2 ...	2 row(s) returned	0.000 sec / 0.000 sec
101	17:56:11	SELECT recipeName, ingredientName, ingredientQuantity, ingredientMeasurement FROM ingredient W...	1 row(s) returned	0.000 sec / 0.000 sec

Result Grid				
Filter Rows:				
Export: Wrap Cell Content:				
	recipeName	recipeTotalTime	recipeServingSize	recipeCalories
▶	Chocolate Pretzel	22	40	37
	Green Eggs and Ham	15	1	210
	Beans and Cheese	7	2	230

Result Grid		
Filter Rows:		
Export: Wrap Cell Content:		
	recipeName	instructionStep
▶	Chocolate Pretzel	1
	Chocolate Pretzel	2

Result Grid				
Filter Rows:				
Export: Wrap Cell Content:				
	recipeName	ingredientName	ingredientQuantity	ingredientMeasurement
▶	Chocolate Pretzel	Corn	2	cob




4.2 Database Operations

```

62 •      UPDATE recipe
63          SET recipeTotalTime = 'QUICK'
64          WHERE recipeTotalTime <= 10;
65
66 •      SELECT *
67          FROM recipe.recipeTotalTime;
68
69 •      DELETE recipe
70          FROM recipe;

```


4.3 Operations Testing / Demonstration

Result Grid			Filter Rows: <input type="text"/>	Export: 	Wrap Cell Contents: <input type="checkbox"/>
	recipeName	recipeTotalTime	recipeServingSize	recipeCalories	
▶	Chocolate Pretzel	22	40	37	
	Green Eggs and Ham	15	1	210	
	Beans and Cheese	7	2	230	

5 Deliverable 4: Application Implementation

This section briefly describes the application that was developed. Appendix A contains a code listing of all applications. In this section, each source file is briefly described in the same order in which it is presented in the appendix.

5.1 CreateRecipe

Description	Create Recipe
User Input	Press create recipe button, input recipe name, total cook time, total calories and serving size.
User Output	A file will be created with recipe name, total cook time, total calories, serving size, ingredients, and instructions.
Database Operations	Create recipe

5.2 UpdateRecipe

Description	Update Recipe
User Input	Press update recipe button, choose file, input total cook time, total calories, serving size, ingredients, and instructions.
User Output	Chosen file will be updated with total cook time, total calories, serving size, ingredients, and instructions.
Database Operations	Update recipe

5.3 ReviewRecipe

Description	Review Recipe
User Input	Press review recipe button, choose file
User Output	Chosen file will display recipe name, total cook time, total calories, serving size, ingredients, and instructions.
Database Operations	Review recipe

5.4 DeleteRecipe

Description	Delete Recipe
User Input	Press delete recipe button, choose file, confirm deletion
User Output	Chosen file will be deleted
Database Operations	Delete recipe

5.5 Help

Description	Help
User Input	Press help button
User Output	Help and instructions for each button will pop up
Database Operations	Help

5.6 Exit

Description	Exit
User Input	Press exit button
User Output	The program will exit out

Database Operations	Exit
------------------------	------

6 Conclusion

We have the home screen GUI built. You can create, delete, review, and update a recipe. The help page works and displays information about the program, the exit button exits out of the program. The program creates text files on your computer that you can send to others. All the buttons work as intended, data can be saved, view, updated and deleted. If a user wants to rename a recipe, they will have to create a new recipe and delete the recipe with the unwanted name. The program still can't add multiple directions and ingredients, they must be added as one text field. The text file that is created is only to be viewed to make sure that the program is working correctly. If the user is not debugging then the user should only use the program to create, update, review or delete the recipe. We will continue to work on this program in the future as a working personal project to better understand databases and MySQL.

References

- A. J. Suarez and K. S. Gorde, "Text editor in Java project with source code," *Itsourcecode.com*, 16-Dec-2021. [Online]. Available: <https://itsourcecode.com/free-projects/java-projects/text-editor-in-java-project-with-source-code/>. [Accessed: 03-May-2022].
- J. D. Ullman and J. Widom, *A first course in database systems*. Upper Saddle River, NJ: Prentice-Hall, 1997.
- "Java gridlayout - javatpoint," *www.javatpoint.com*. [Online]. Available: <https://www.javatpoint.com/GridLayout>. [Accessed: 03-May-2022].
- Java swing how to - display the contents of a text file in a jtextarea*. [Online]. Available: http://www.java2s.com/Tutorials/Java/Swing_How_to/JFileChooser/Display_the_Contents_of_a_text_file_in_a_JTextArea.htm. [Accessed: 03-May-2022].
- Naomi HarrisNaomi Harris 3511 gold badge11 silver badge44 bronze badges, JulienJulien 2, JitendraJitendra 1, and Vicky ThakorVicky Thakor 3, "How to open a text file?," *Stack Overflow*, 01-May-1961. [Online]. Available: <https://stackoverflow.com/questions/18551251/how-to-open-a-text-file>. [Accessed: 03-May-2022].
- R. S. Stansbury, "CS 317 Files/Database Systems," *Embry-Riddle Aeronautical University - Login*. [Online]. Available: <https://erau.instructure.com/courses/140247>. [Accessed: 03-May-2022].
- SunnYSunnY 4911 gold badge44 silver badges1111 bronze badges, nIcE cOwnIcE cOw 24.2k77 gold badges4747 silver badges133133 bronze badges, and Dan D.Dan D. 32.1k55 gold badges6161 silver badges7979 bronze badges, "Java- how to add more Textfields by clicking a button?," *Stack Overflow*, 01-May-1960. [Online]. Available: <https://stackoverflow.com/questions/12251618/java-how-to-add-more-textfields-by-clicking-a-button>. [Accessed: 03-May-2022].
- Swing examples - open file dialog in swing*. [Online]. Available: https://www.tutorialspoint.com/swingexamples/show_open_file_dialog.htm. [Accessed: 03-May-2022].
- Use One actionlistener to handle event from multiple buttons - java swing*. [Online]. Available: <http://www.java2s.com/example/java/swing/use-one-actionlistener-to-handle-event-from-multiple-buttons.html>. [Accessed: 03-May-2022].
- "Writing a text file in Java using JFrame/GUI," *YouTube*, 05-Jan-2017. [Online]. Available: <https://www.youtube.com/watch?v=y8LiDp5jgTc>. [Accessed: 03-May-2022].

Appendix A: Source File Listing

<https://vimeo.com/705656990>

[nchapman6577/Gigabyte-CS-317-Final-Project: Recipe Management System
\(github.com\)](https://github.com/nchapman6577/Gigabyte-CS-317-Final-Project)