Team Members:

Vishnu Rama Varma Thampuran

Pranav Kuchibhotla

**PROJECT DESIGN DOCUMENT**

Abstract:

Create a fully functional piece of software that can store specific sets of data in a file, manipulate the file by performing tasks like appending, deleting, searching and displaying said data and ensure it is properly tested and debugged before deployed for real-world applications.

Tools Used:

Java, Command Line, CSV files and Microsoft Excel

Agenda:

Refer end of document.

System Architecture:

Main program would be developed using the programming language Java. The code would contain multiple methods and classes using recursion, iterations, etc. that aid in achieving the user’s requirements by establishing a connection with its database in .csv files which will be used to store and update user generated data and input. This file can later be used to display data on Microsoft Excel as a whole or to display necessary and prompted data using the command line. Database stored in csv files would contain information stored in table formats with columns like name, brand, price, expiration, and status. Key goals include developing a main menu that provide three options, namely adding, searching and listing data alongside input received from the user, testing and debugging the code and finally optimise it for efficiency and accuracy.

**Term Project – Project Management Schedule**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Name** | Cs\_201\_Project | | | | |
| **Team Mates** | Vishnu Rama Varma Thampuran | | | Pranav Kuchibhotla | |
| **Start Date** | 10/28/2022 | **End Date** | 11/28/2022 | **Overall Progress** | 100% |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Both team members would work together for an hour everyday*** | | | | |
| **S.No** | **Task Name** | **Tools** | **Start/End Date** | **Duration** |
| **1)** | Learning/training on how to use CSV. | CSV | 28/10 to 4/11 | 7 days |
| **2)** | Learning GUI/Command line for displaying main menu | GUI/Command line | 5/11 to 10/11 | 6 days |
| **3)** | Adding a new Product. | Java | 11/11 to 13/11 | 3 days |
| **4)** | Searching the product by name. | Java | 14/11 to 16/11 | 3 days |
| **5)** | Listing the products by categories. | Java | 17/11 to 20/11 | 4 days |
| **6)** | Reviewing and testing the code. | Java | 21/11 to 22/11 | 2 days |
| **7)** | Complete the requirements which are required to submit the project. |  | 23/11 to 25/11 | 3 days |

**DESIGN DIAGRAM**

Diagram

Description automatically generated

**UML Diagram**Diagram

Description automatically generated

**OPERATIONAL DOCUMENT (USER’S MANUAL)**

* Input-only string values for the category, name, brand, price, expiration period, and status.
* Make sure that the expiration period is between 1-365 days.
* "Add another product(1) or Return to main menu(2):". When this pops up make sure to only enter the option number.
* Only type Yes(or)No when asked to input one of them.
* In the main menu when asked input only one of 1, 2, or 3.
* Make sure you double-check the input values which you entered.

**PROJECT REPORT**

1) This is our initial list (master list).

Table

Description automatically generated

2) This is our Main menu. Here you can select the option to choose a function.

Text

Description automatically generated

3) When we select the first option it will take us to the adding product’s function.

Text, letter

Description automatically generated

4) We can add a product by entering the values-

Text, letter

Description automatically generated

5) You can either add another product or return to the main menu. If you want to add a product again it will ask you to confirm again.

Graphical user interface, text, application

Description automatically generated

6) Repeat the process for another category. If you select No you will go back to the main menu.Text

Description automatically generated

7) It will be entered in the excel sheet once we reopen it.

Table

Description automatically generated

8) Now, selecting option 2 will take us to the *searching for product* method.

Graphical user interface, text, application

Description automatically generated

9) Enter the name of the product you want to search for.

A picture containing text

Description automatically generated

10) Entering a product name that is not in the loop will show” No product found”.Entering No will take you back to the main menu.

Text

Description automatically generated

11) Now selecting the third function in the main menu will show all available categories in the current list and let the user choose one category.

Graphical user interface, text, application

Description automatically generated

12) Now selecting one of the categories will give us -

Graphical user interface, text, application, Word

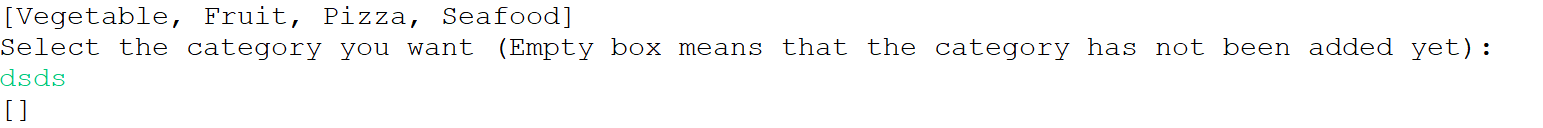
Description automatically generated

13) As we enter the correct product name. Now, you have the option to go back to the main menu or list another category.

Text

Description automatically generated

14) Giving a random category name will give you an empty array.



15) Giving and product name which is unavailable will give the user a message telling that the product was not found.

Graphical user interface, text, application

Description automatically generated

16) Selecting ‘No’ will take you back to the main menu, the place where it started.Graphical user interface, text, application

Description automatically generated

17) If you enter any other value instead of 1,2 or 3 this will show up.

Text, letter

Description automatically generated

**DEBUGGING NOTES**

* Integrate exit to main menu function into all classes that execute functions of appending data, searching, and categorized listing.
* Search function must display all details of products listed
* Files must be opened in append mode instead of write mode to avoid erasing all of the pre-existing data in the file.

**FUTURE IMPROVEMENTS**

* Manage time in a better way to optimally arrive at a solution for the problem statement at the earliest.
* Explore displaying output and integrating existing code with GUI.
* Run more test cases with different sizes of data to assess efficiency of our algorithms.
* Create hints and features supporting multiple data types as input.