

Course Work

ITS1010 – Programming Fundamentals

BSc (Hons.) in Computing



Take-home assignment and

VIVA Total Marks: 100

Objectives

- Solve fundamental programming problems using a variety of skills and strategies.
- Analyze the logical solutions to any real-world problems you may encounter.
- Understanding the operation of CLI applications.

Coursework requirements and instructions

- You should develop this system utilizing your understanding of CLI (Command Line Interface) software.
- You should've sound knowledge of Multi-Dimensional Arrays.
- You are required to successfully attempt all the parts and face the viva evaluation at the end of this coursework.
- Apart from explaining the code you have written, you should be able to explain the theory concepts tested in this coursework.
- Refer to the coursework guidelines from beginning to end to fully comprehend the unique criteria for constructing this system.
- Demo videos are provided at relevant points to help you better grasp the curriculum requirements.

Submission

- You should submit the deliverables of the coursework on or before the due date designated. If not, you will not be eligible for the final viva evaluation.
-
- Google Classroom is where you should submit your deliverables after converting your work into a .zip file with the filename format **[GDSEBatch_ITS1010_Name]**.

Case study

The Pigeon Service is a prominent distributed company that is near the Colombo area. It imports a variety of goods from abroad and sells them to wholesale shops in Sri Lanka.

The company has several departments to manage the business. The stock management department is one of the most responsible parts of the business. It imports items from suppliers and maintains a stock of those items.

The stock management department requires a system to manage this item stock accurately. As you are a well-trained GDSE student, they thought to give you a chance of making a system for them.

Use case diagram



Figure 1 - use case

Let's jump into this. 

In the application, you need to implement the following use cases.

When you run the application, you should come up with something similar to the following Command Line Interface (CLI), where the user can interact with the system. User can enter an option number to execute the function that he/she wants.

The following figures will be the application pages you will be developing.

1. Login to the system [\[Demo\]](#)

This system should have a login page to the giving password protection. First, you need to enter the user name and then only if it is a valid username, you are asked for the password. You can store your pre-defined credential somewhere and compare them with it.

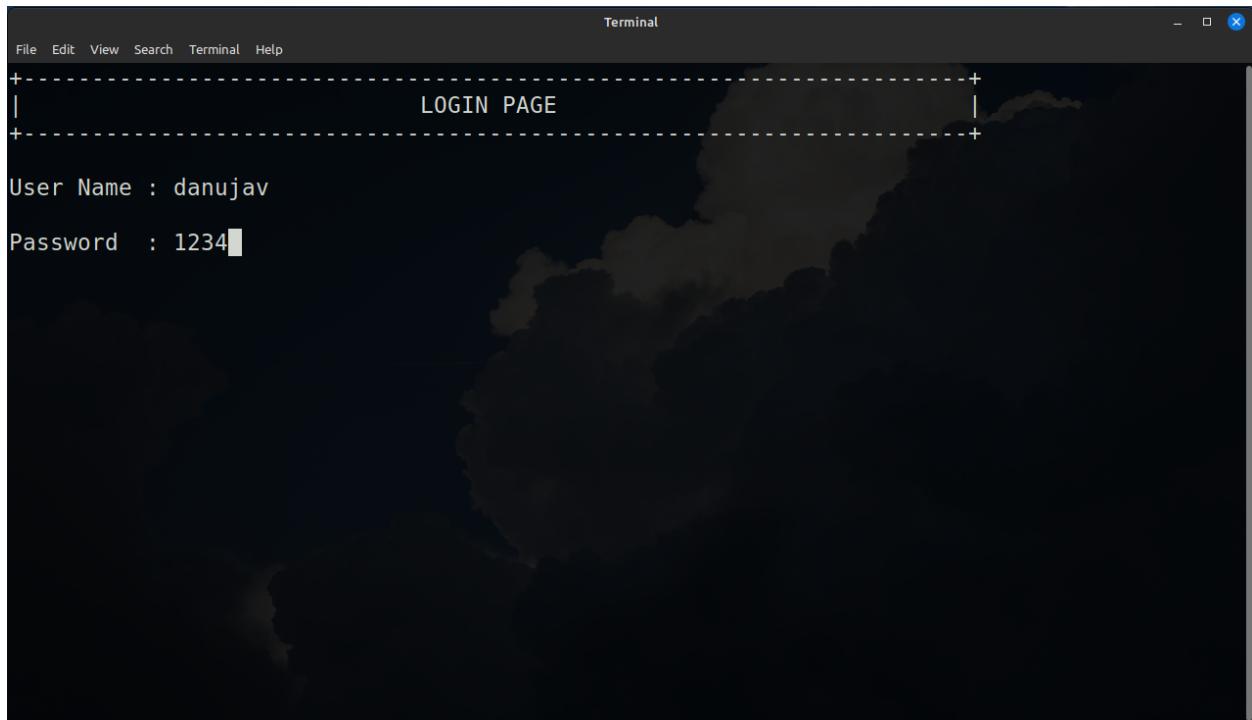
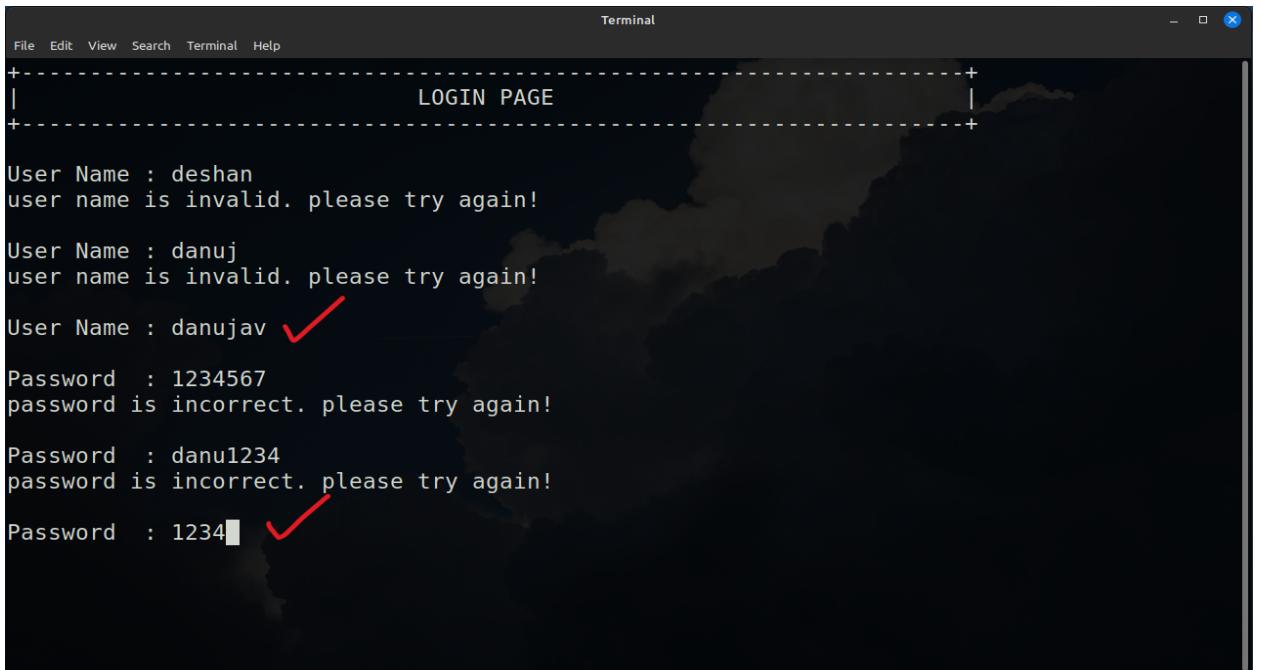


Figure 2 - Login with credentials

If somehow you enter an invalid username or incorrect password, the system should be notified about it and asked to enter the correct credentials again. This should keep prompting until the user enters valid credentials.



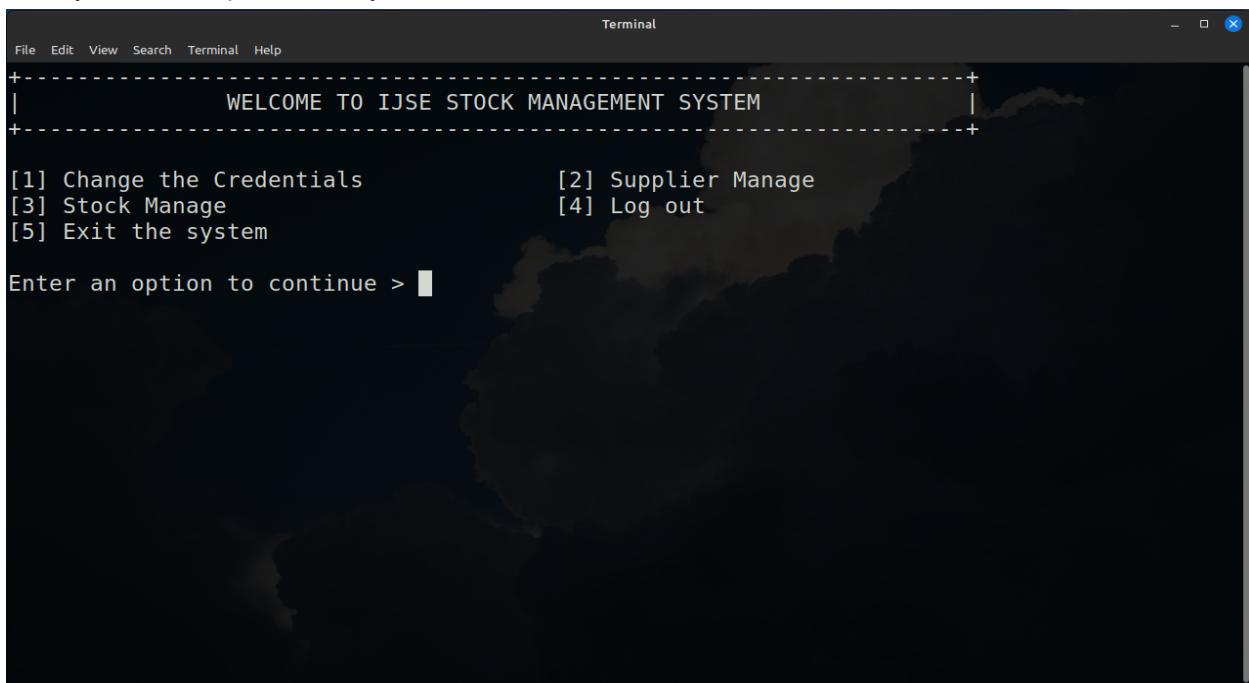
```
Terminal
File Edit View Search Terminal Help
+-----+
|          LOGIN PAGE
+-----+
User Name : deshan
user name is invalid. please try again!
User Name : danuj
user name is invalid. please try again!
User Name : danujav ✓
Password : 1234567
password is incorrect. please try again!
Password : danu1234
password is incorrect. please try again!
Password : 1234█ ✓
```

A terminal window titled "Terminal" showing a "LOGIN PAGE". It displays several failed login attempts. The first two attempts ("User Name : deshan" and "User Name : danuj") show an invalid user name warning. The third attempt ("User Name : danujav") is marked with a red checkmark. The next three attempts ("Password : 1234567", "Password : danu1234", and "Password : 1234█") show an incorrect password warning. The last attempt is also marked with a red checkmark. The terminal has a dark background with a faint mountain silhouette.

Figure 3 - Login with a credential (Warning - invalid and incorrect credentials)

2. Home Page

Here you have 5 options mainly.



```
Terminal
File Edit View Search Terminal Help
+-----+
|          WELCOME TO IJSE STOCK MANAGEMENT SYSTEM
+-----+
[1] Change the Credentials      [2] Supplier Manage
[3] Stock Manage                [4] Log out
[5] Exit the system
Enter an option to continue > █
```

A terminal window titled "Terminal" showing the "WELCOME TO IJSE STOCK MANAGEMENT SYSTEM". It presents five main menu options: [1] Change the Credentials, [2] Supplier Manage, [3] Stock Manage, [4] Log out, and [5] Exit the system. Below the options, it prompts the user to "Enter an option to continue >". The terminal has a dark background with a faint mountain silhouette.

Figure 4 - Home Page of the Stock Management System

3. Change the Credentials [Demo]

So after the login to the system, if you want to change your login password, go for an [1] option. It will be asking for your valid user name to verify it's you. Then it asks for your current password for the security reason. Finally, you can change your password with your new password. If you successfully change your password, the system will notify you about it.

The screenshot shows a terminal window titled "Terminal". The window has a dark background with white text. At the top, there is a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". Below the menu, there is a dashed-line box containing the text "CREDENTIAL MANAGE". The terminal output is as follows:

```
Please enter the user name to verify it's you: danu X
invalid user name. try again!

Please enter the user name to verify it's you: danujav
Hey danujav
Enter your current password: 123 X
incorrect password. try again!

Enter your current password: 1234
Enter your new password: danu1234

Password changed successfully! Do you want to go home page (Y/N):
```

The line "Hey danujav" is highlighted with a red box, and the line "Password changed successfully!" is also highlighted with a red box.

Figure 5 - Change the password

4. Log out[Demo]

By the time the user wants to sign out from the system, he/she can use the [4] option to log out.

The screenshot shows a terminal window titled "Terminal". The window has a dark background with white text. At the top, there is a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". Below the menu, there is a dashed-line box containing the text "WELCOME TO IJSE STOCK MANAGEMENT SYSTEM". The terminal output is as follows:

```
[1] Change the Credentials      [2] Supplier Manage
[3] Stock Manage               [4] Log out
[5] Exit the system

Enter an option to continue >
```

The option "[4] Log out" is highlighted with a red box.

Figure 6 - Log out from the System

5. Exit the System [Demo]

If the user wants to exit the system directly, he/she can use the [5] option for that.

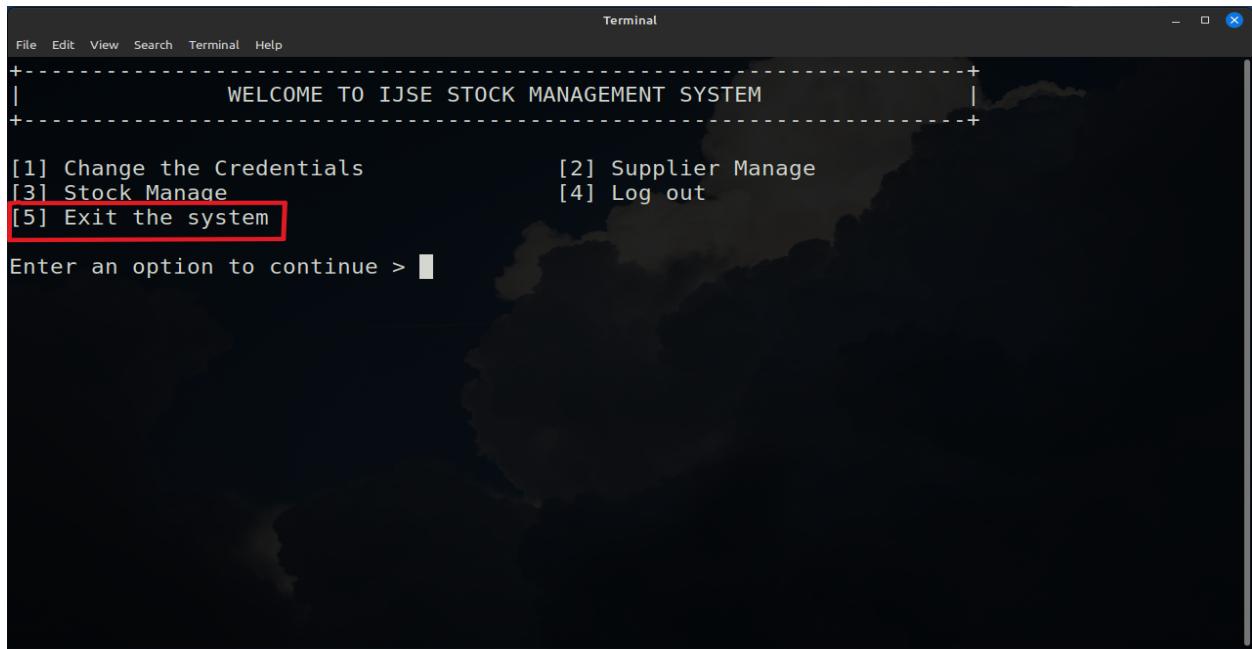


Figure 7 - Exit the System Directly

6. Supplier Manage

As mentioned earlier while maintaining item stock, the stock management department should want to keep the suppliers' detail on the system. You can go with the [2] option for it.

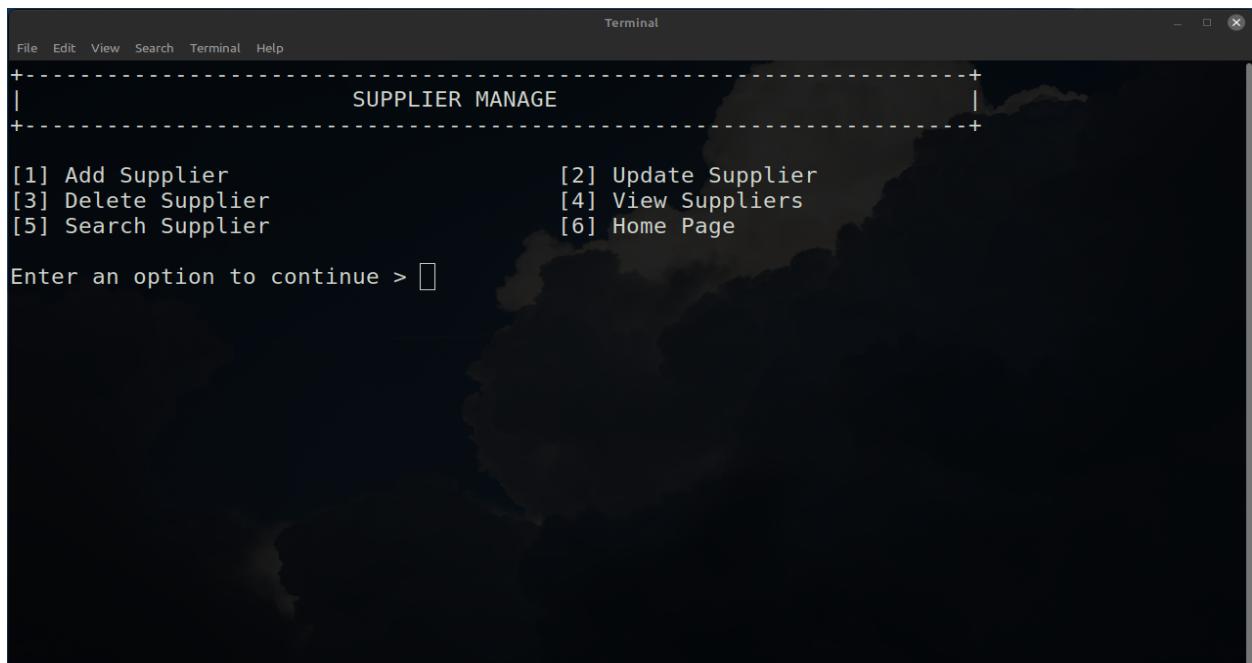
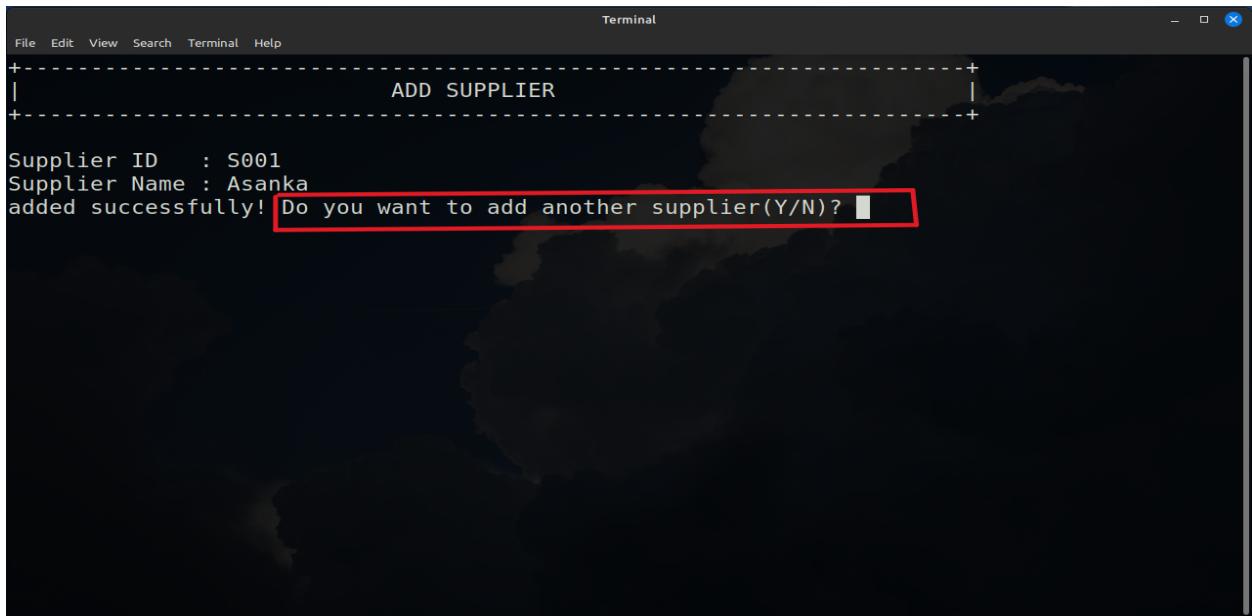


Figure 8 - Supplier Manage Page

7. Add Supplier[[Demo](#)]

Previously users were navigating Home Page to the Supplier Management page. Here, the user can add suppliers to the system. The first user needs to enter the supplier id and supplier name. Once the user has added a new supplier successfully, a message should prompt to ask where the user wants to add a new supplier again or go back to the supplier management page.



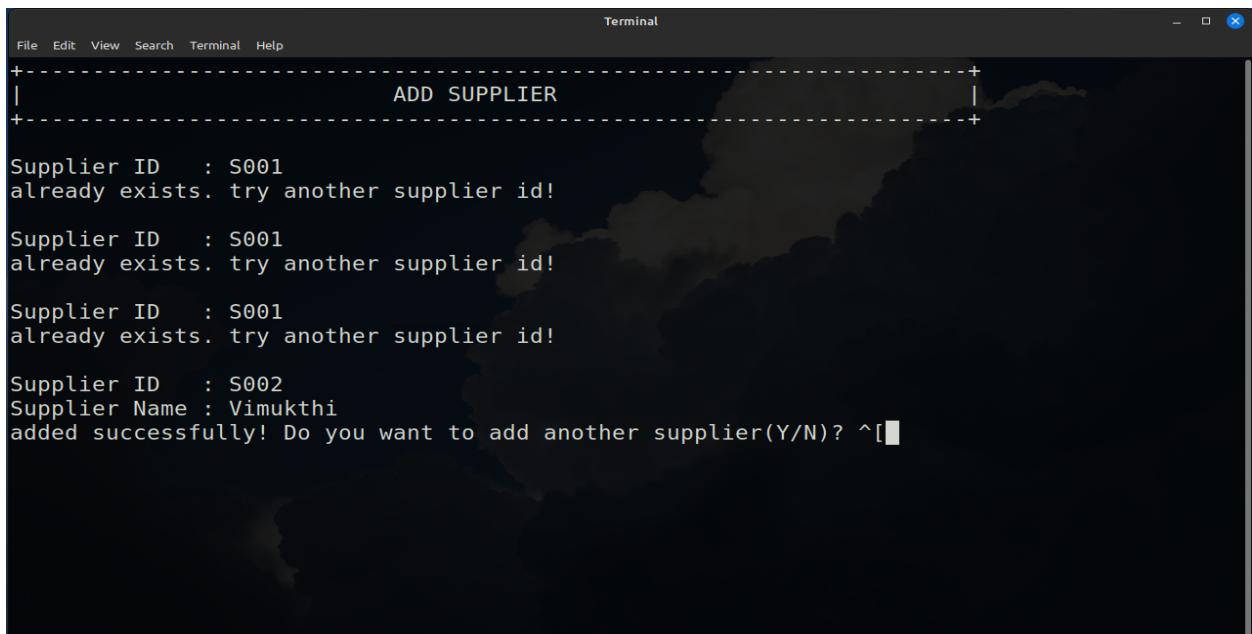
A screenshot of a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The main area displays the following text:

```
Supplier ID : S001
Supplier Name : Asanka
added successfully! Do you want to add another supplier(Y/N)? █
```

The last line of text is highlighted with a red rectangular box.

Figure 9 - Add New Supplier (Successful)

It is not allowed to add the same Supplier ID again. If the user has entered a supplier ID that already exists in the system, the user should be notified about it and asked to enter the correct Supplier ID again. This should keep prompting until the user enters a valid Supplier ID.



A screenshot of a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The main area displays the following text:

```
Supplier ID : S001
already exists. try another supplier id!

Supplier ID : S001
already exists. try another supplier id!

Supplier ID : S001
already exists. try another supplier id!

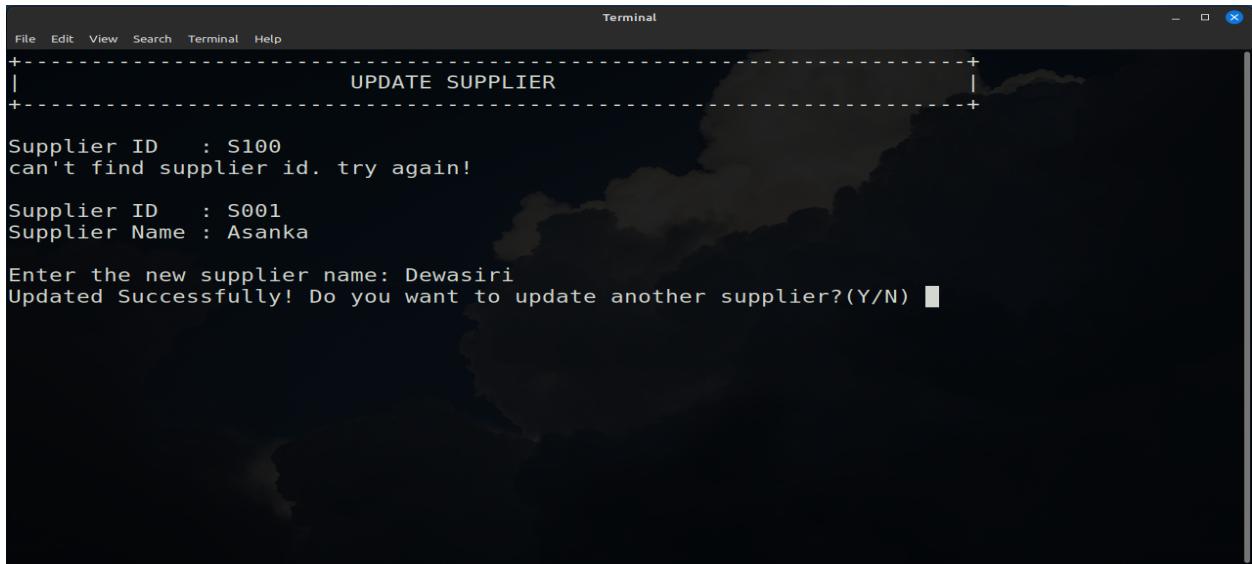
Supplier ID : S002
Supplier Name : Vimukthi
added successfully! Do you want to add another supplier(Y/N)? ^█
```

The first three lines of text are repeated, each followed by a warning message. The last line of text is highlighted with a red rectangular box.

Figure 10 - Add New Supplier (Warning - already exists)

8. Update Supplier[[Demo](#)]

With this option [2] (Supplier Management), the user can update existing supplier details (Supplier Name). First, the user should enter a valid Supplier ID, otherwise, it should prompt like previously. Once the user enters the valid Supplier ID, it will display the current Supplier Name and ask for the new Supplier Name. Once the data has been updated successfully, it should prompt whether to update another supplier or go back to the stock management menu.

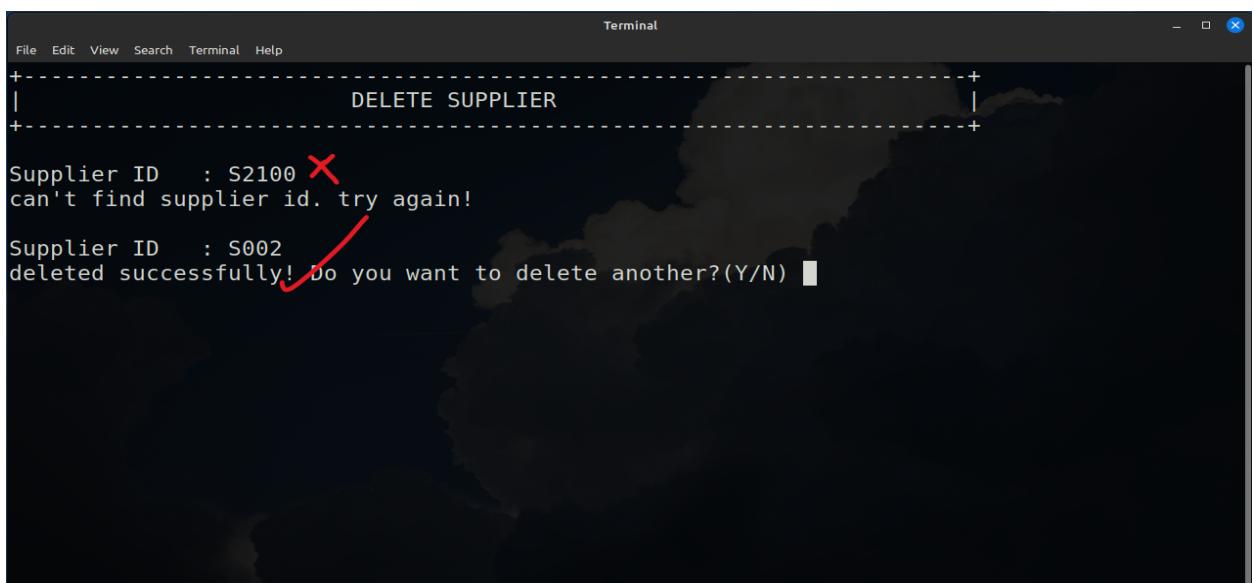


```
Terminal
File Edit View Search Terminal Help
+-----+
|          UPDATE SUPPLIER
+-----+
Supplier ID : S100
can't find supplier id. try again!
Supplier ID : S001
Supplier Name : Asanka
Enter the new supplier name: Dewasiri
Updated Successfully! Do you want to update another supplier?(Y/N) █
```

Figure 11 - Update Supplier Details

9. Delete Supplier[[Demo](#)]

With this, if the user thinks the one supplier doesn't want to be in the system, then the user can simply remove that supplier from the system. First users need to enter a valid and respective Supplier ID. Upon successful deletion, the user should be prompted whether to delete another supplier or go back to the supplier management menu.

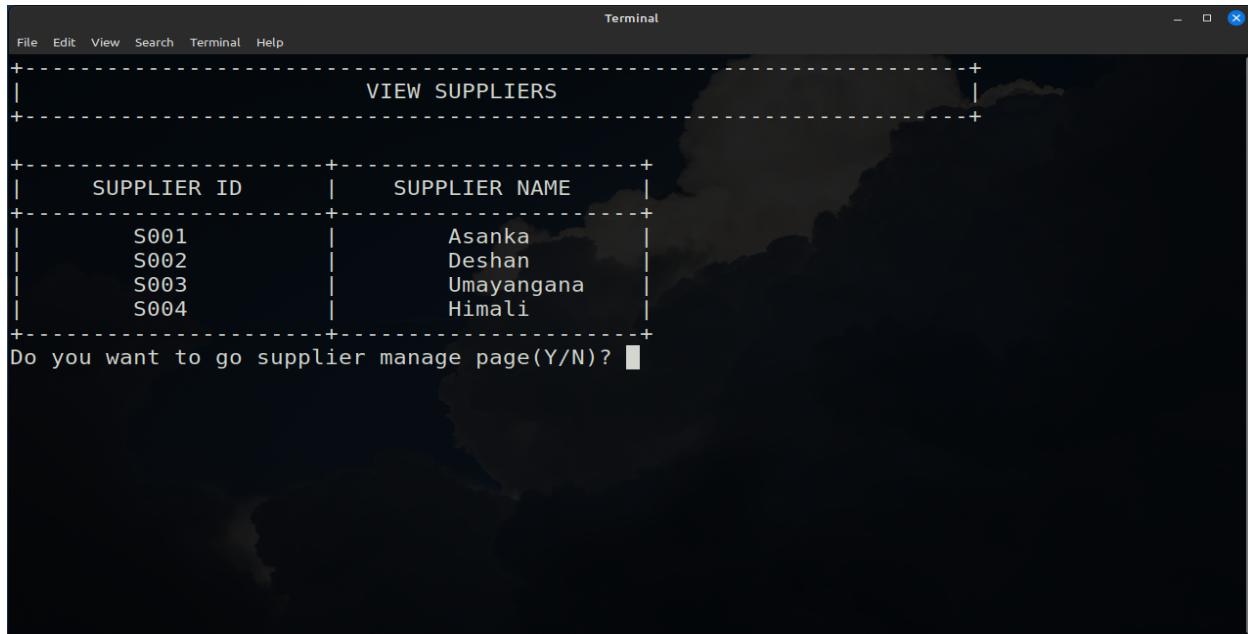


```
Terminal
File Edit View Search Terminal Help
+-----+
|          DELETE SUPPLIER
+-----+
Supplier ID : S2100 X
can't find supplier id. try again!
Supplier ID : S002
deleted successfully! Do you want to delete another?(Y/N) █
```

Figure 12 -Delete Supplier

10. View Suppliers[[Demo](#)]

With this option, the user can view Supplier details respectively. Users should view those supplier details in a tabular format like below. After that system will be asking to go supplier manages the menu or not. If the user presses the option “N” then the system will stop its execution.



The screenshot shows a terminal window titled "Terminal". At the top, there is a menu bar with options: File, Edit, View, Search, Terminal, Help. Below the menu, the title "VIEW SUPPLIERS" is displayed. A table is shown with columns "SUPPLIER ID" and "SUPPLIER NAME". The data rows are:

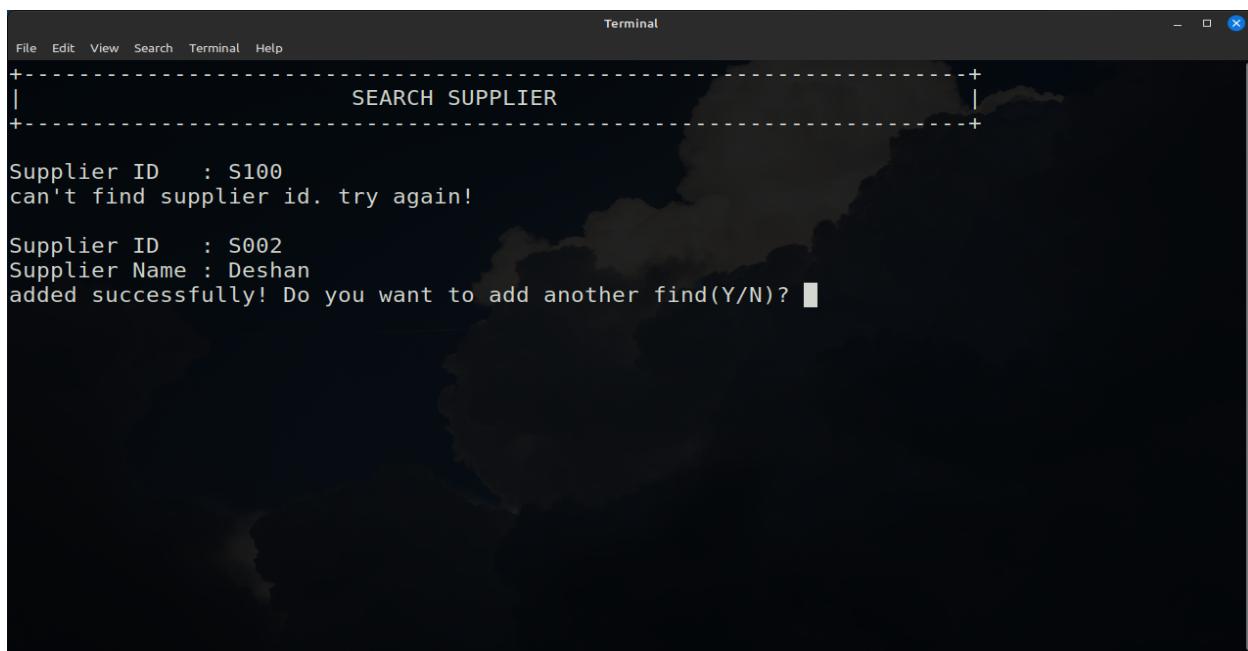
SUPPLIER ID	SUPPLIER NAME
S001	Asanka
S002	Deshan
S003	Umayangana
S004	Himali

At the bottom of the terminal, the question "Do you want to go supplier manage page(Y/N)? " is displayed, followed by a small input field.

Figure 13 - View Suppliers' details through tabular format

11. Search Supplier[[Demo](#)]

In this option, the user can see the details of each supplier individually. Users have to enter a valid user id. Otherwise, it will continuously ask you to enter a valid Supplier ID.



The screenshot shows a terminal window titled "Terminal". At the top, there is a menu bar with options: File, Edit, View, Search, Terminal, Help. Below the menu, the title "SEARCH SUPPLIER" is displayed. The terminal output shows:

```
Supplier ID : S100
can't find supplier id. try again!

Supplier ID : S002
Supplier Name : Deshan
added successfully! Do you want to add another find(Y/N)? 
```

Figure 14 - Search Supplier Details Individually

12. Home Page

If the user's job is done inside the supplier management menu, then he should come back to the Home Page back. If the user presses the respective option, then the user will navigate to the Home Page.

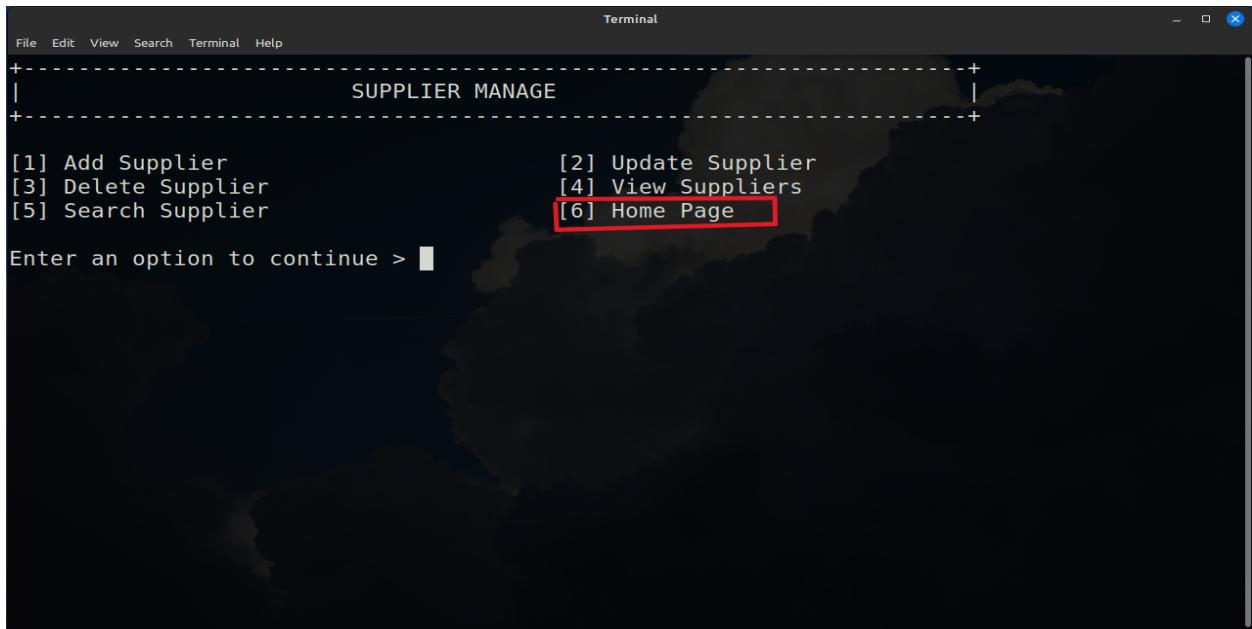


Figure 15 - Navigate to the Home Page back

13. Stock Manage

In this system, the main business process is stock management. So if the user wants to manage the stock who needs to navigate this menu.

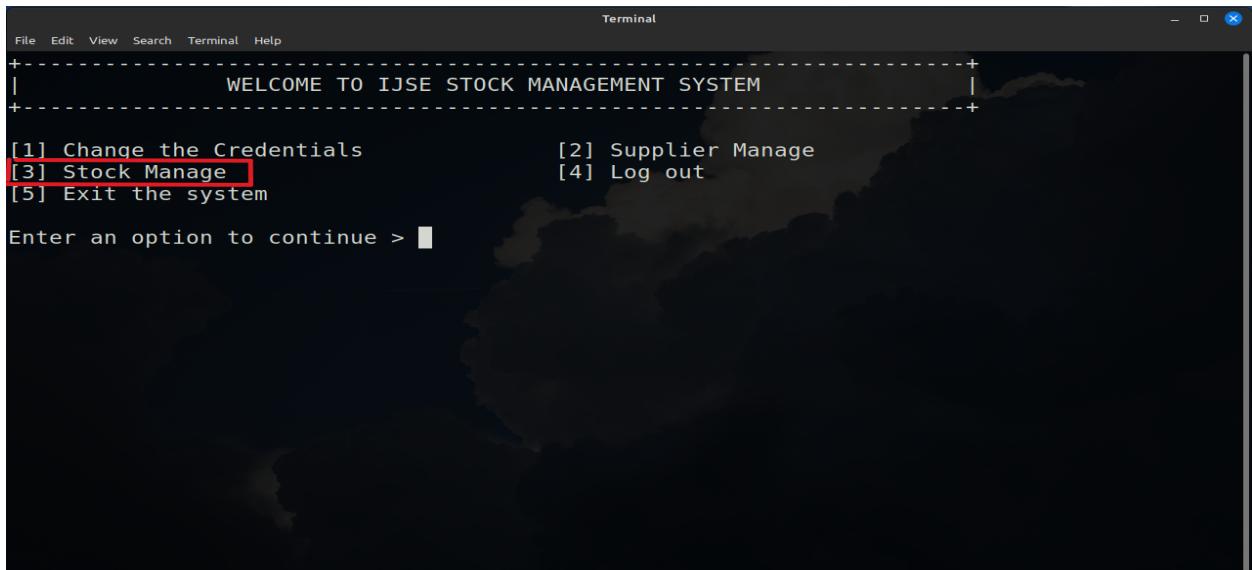


Figure 16 - Navigate to Stock Management

When this company imports Items from aboard, every item has a category, for example, Food, Gift, Jewellery, etc... So, before the manage the items, the user needs to store those categories in the system.

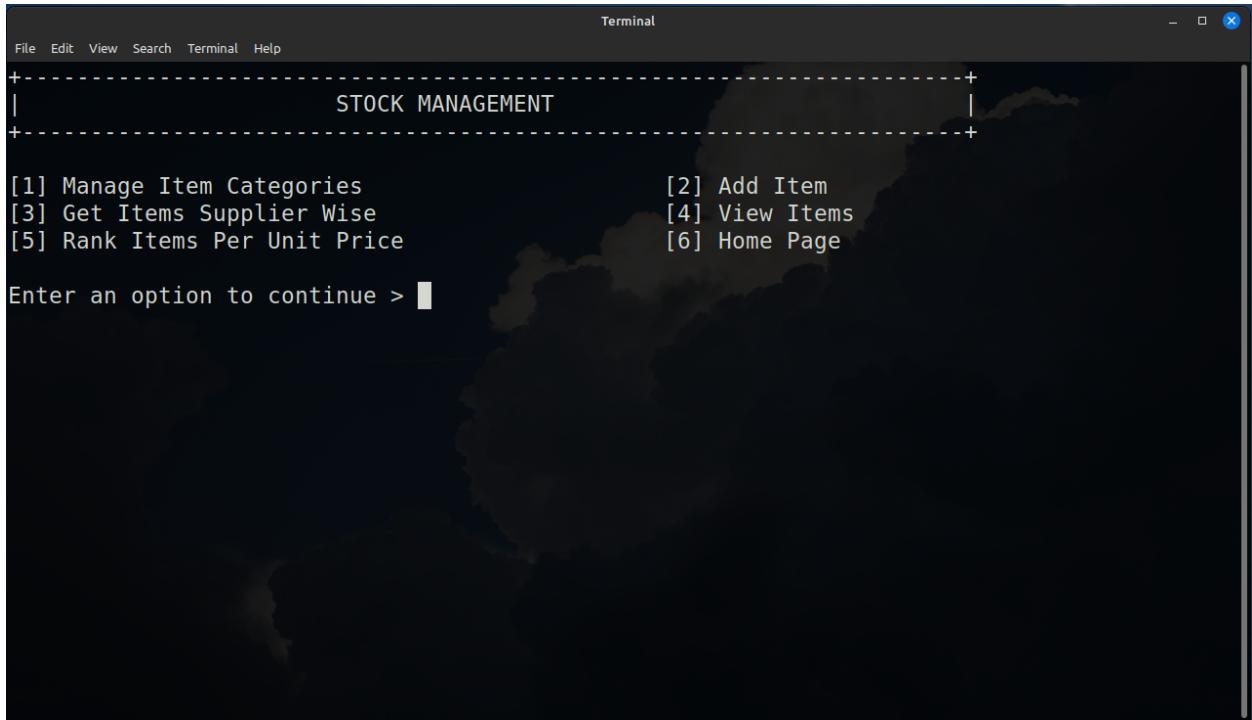


Figure 17 - Stock Management

14. Manage Item Categories

As mentioned earlier, the user needs to Manage Item Categories before the manage items. So the user needs to navigate the [1] option.

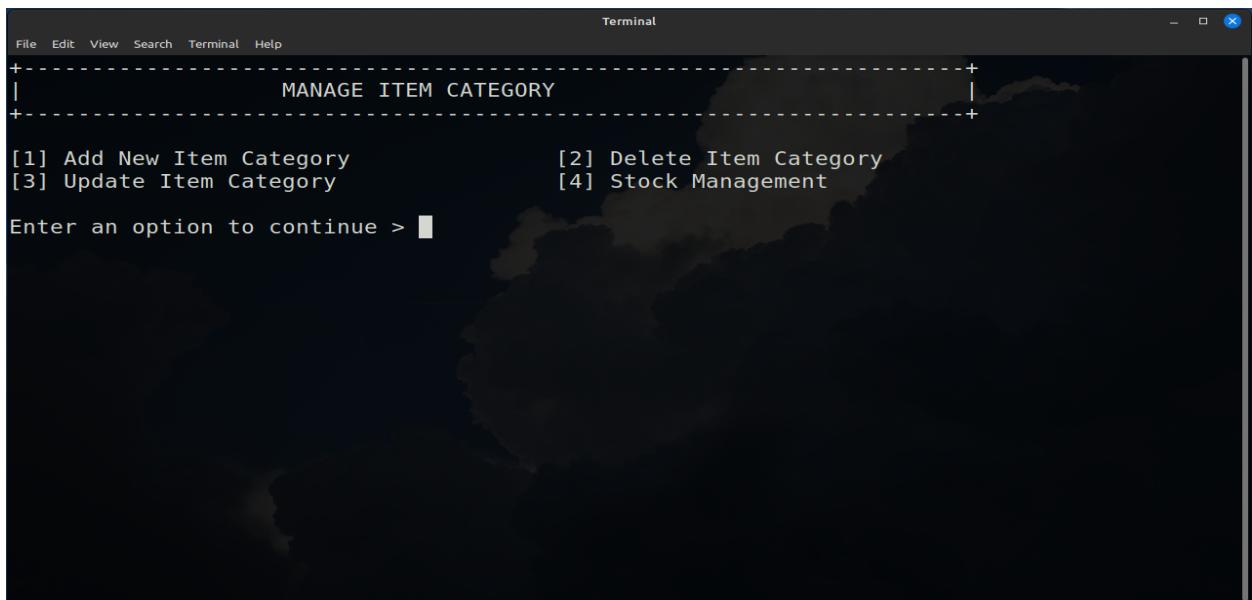


Figure 18 - Manage Item Categories

15. Add New Item Category[\[Demo\]](#)

User can add item categories in the above menu's [1] option.

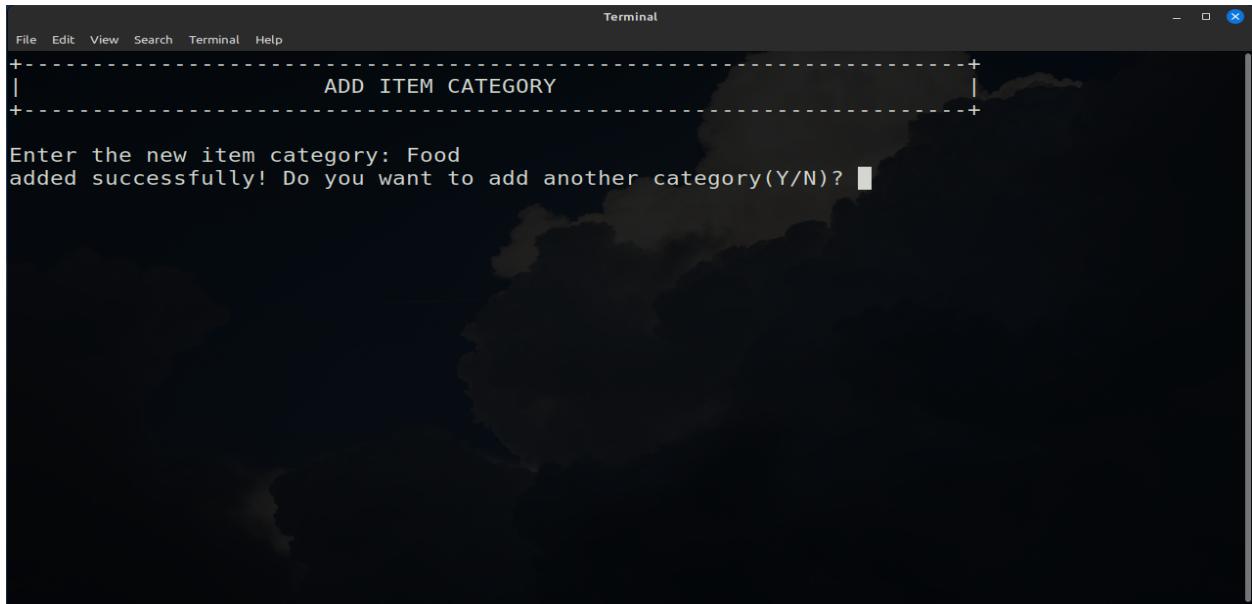


Figure 19 - Add Item Category

User can also manage item categories like delete categories and update categories.

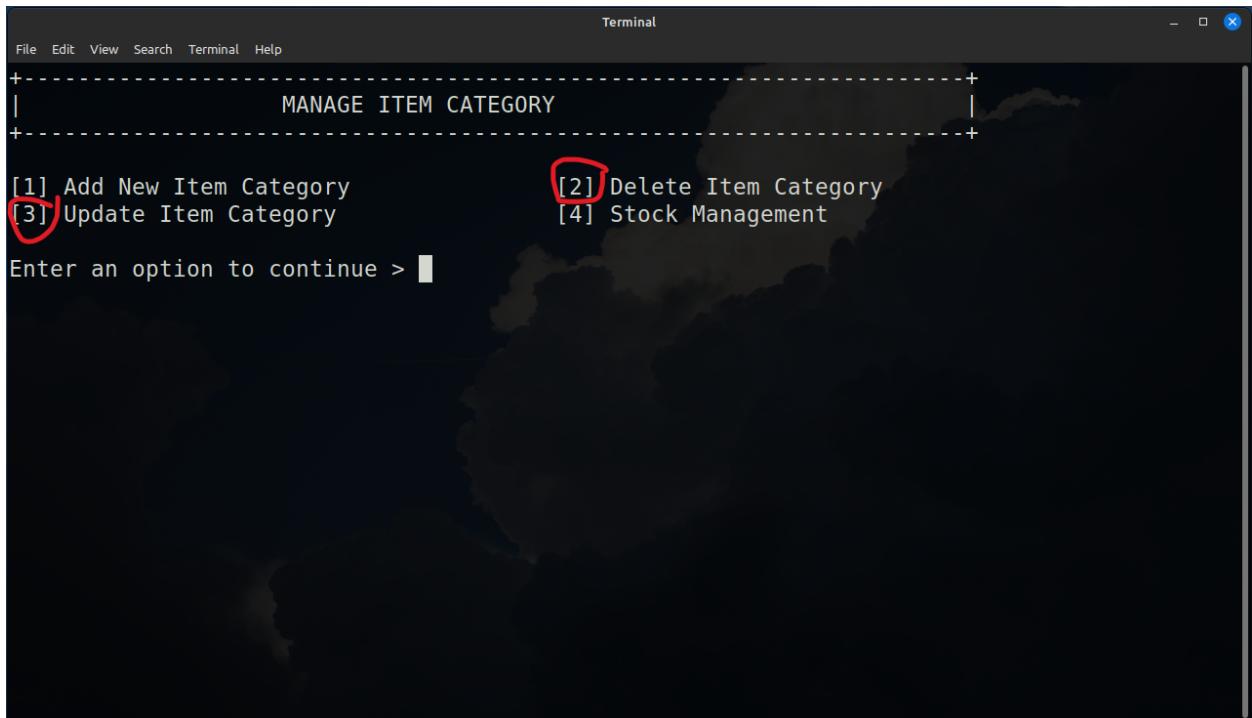


Figure 20 - Item Category basic operations (Update, Delete)

16. Add Item[[Demo](#)]

In here is the main process of the business happening. When items are arriving from other countries, the user should store them in the system. When storing items, the user should know about each item's category, supplier details, and other details about the items.

And the other special thing is before adding an item to the system, the user should add the item's category names, and the supplier details to the system. Otherwise, the system will be notified you about adding them in the first place.

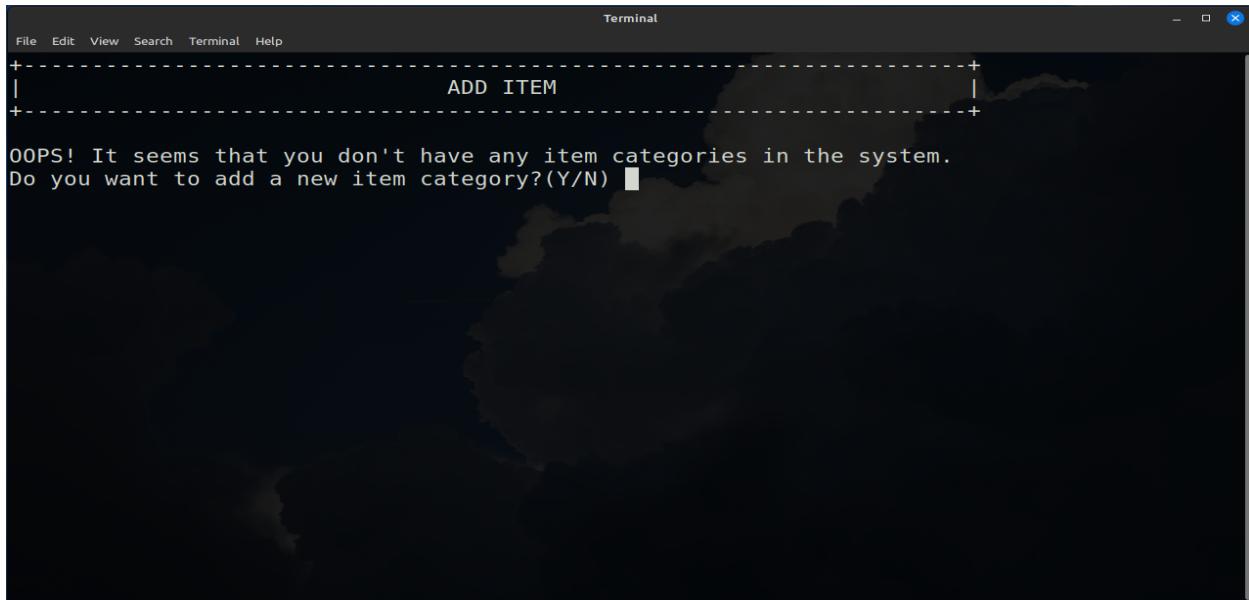


Figure 21 - Add Item form (Without adding Item Categories to the System)

In here the system has asked you to add a new item category or not. If you wish to continue this process, you should add item categories by pressing **Y**.

After that process when you try to add a new item again, in that time if you are not added any Supplier, the program will be asking you to add supplier details at first.

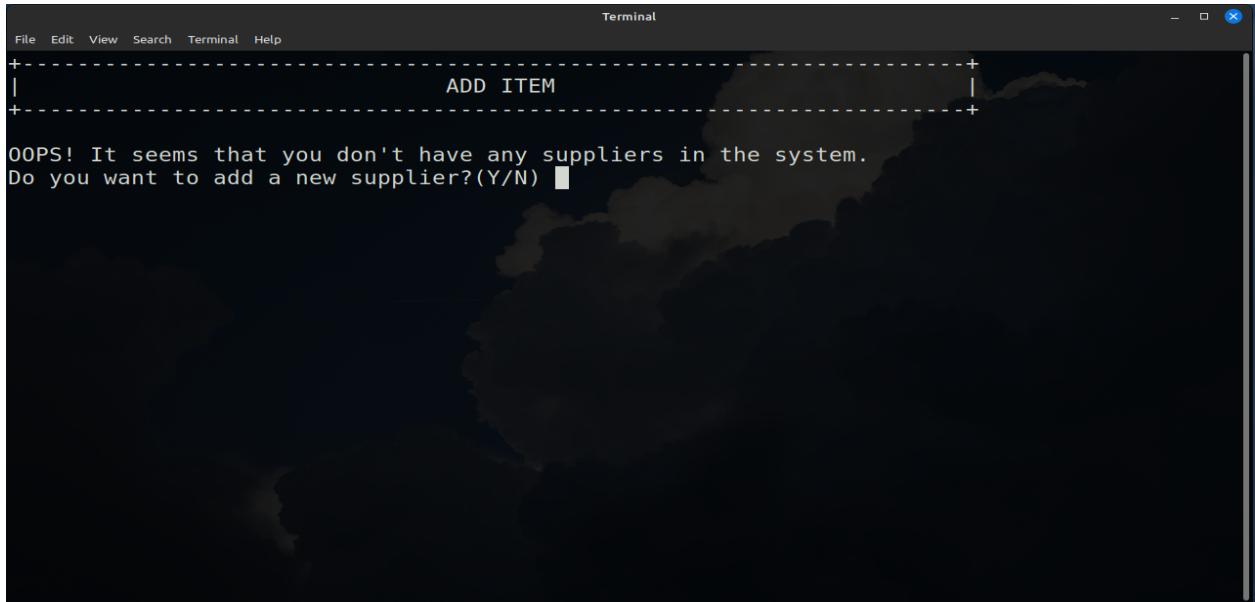


Figure 22 - Add Item form (Without adding Suppliers to the System)

If you are already added both suppliers and item categories, then the system will continue.



Figure 23 - Add Item

You need to give an Item code in here. After that system should automatically load all the supplier details with the number in tabular format. It should ask you for a number for a respective supplier who belongs to that Item.

After that, the system asks to enter the Item category name as previously.

The screenshot shows a terminal window titled "Terminal". The menu bar includes "File", "Edit", "View", "Search", "Terminal", and "Help". Below the menu, there is a table with four rows:

1	S001	Asanka
2	S002	Himali
3	S003	Vimukthi
4	S004	Prasanna

Below the table, the text "Enter the supplier number > 1" is displayed. Then, the text "Item Categories:" is shown, followed by another table:

#	CATEGORY NAME
1	Food
2	Medicine
3	Gift

Finally, the text "Enter the category number > 1" is displayed at the bottom of the terminal window.

Figure 24 - Add Item

After that you can simply add the item details. But the system should know which items belong to which category and the supplier.

The screenshot shows a terminal window titled "Terminal". The menu bar includes "File", "Edit", "View", "Search", "Terminal", and "Help". The process starts with "Enter the supplier number > 1", followed by "Item Categories:" and the same table as in Figure 24. Then, "Enter the category number > 2" is displayed. Subsequent lines show item details being entered:

Description : Panadol
Unit price : 45
Qty on hand : 2
added successfully! Do you want to add another Item(Y/N)?

Figure 25 - Add Item (Successful)

Each item has an item code, supplier's id, item category, description, unit price, and quantity on hand.

17. Get Items Supplier Wise[[Demo](#)]

In the next option the user can see items by supplier wise. You need to enter the valid supplier id. The system will show the supplier-wise result.

The screenshot shows a terminal window titled "Terminal". The command "SEARCH SUPPLIER" is run, followed by entering "Supplier Id: S003". The output displays the supplier name "Vimukthi" and a table of items:

ITEM CODE	DESCRIPTION	UNIT PRICE	QTY ON HAND	CATEGORY
I002	TOFEE	50.0	120	FOOD
I004	MONKEY	4550.0	34	GIFT

The message "search successfully! Do you want to another search?(Y/N)" is displayed at the bottom.

Figure 26 - Search Item Supplier Wise

18. View Items[[Demo](#)]

In this option the user can view items grouped by their item category. All data should be printed in tabular format.

The screenshot shows a terminal window titled "Terminal". The command "VIEW ITEMS" is run, displaying items grouped by category:

Food:

SID	CODE	DESC	PRICE	QTY
S003	I002	TOFEE	50.0	120

Medicine:

SID	CODE	DESC	PRICE	QTY
S002	I001	PANADOL	45.0	23

Gift:

SID	CODE	DESC	PRICE	QTY
-----	------	------	-------	-----

Figure 27 - View Items

19. Rank Items Per Unit Price[\[Demo\]](#)

In this option all the Items will be sorted on their unit price and they will be showing in ascending order.

RANKED UNIT PRICE							
SID	CODE	DESC	PRICE	QTY	CAT		
S002	I001	PANADOL	45.0	23	MEDICINE		
S003	I002	TOFEE	50.0	120	FOOD		
S001	I003	TEDDY	1540.0	30	GIFT		
S003	I004	MONKEY	4550.0	34	GIFT		

Figure 28 - Ranked Item List per unit price

Guidelines

- You may use any number of arrays and any dimensional array.
- You can use Google as your supporter.
- Expect for arrays you can't use any other Java data structure classes in Java SE framework like ArrayList, LinkedList, etc...
- You can't use any other 3rd party libraries or frameworks.
- You can't create classes except for the class that holds the main method.
- User Scanner and the Random for handle keyboard inputs.
- All validations that have been mentioned in this document should be implemented.
- It is not required to clear the command line screen while navigating between the options. But doing so highly recommend it.
- The code to clear the command line from inside a Java application is as follows. You can use this code when you need to clear the command line.

```
private final static void clearConsole() {
    final String os = System.getProperty("os.name");
    try {
        if (os.equals("Linux")) {
            System.out.print("\033\143");
        } else if (os.equals("Windows")) {
            new ProcessBuilder("cmd", "/c", "cls").inheritIO().start().waitFor();
        } else {
            System.out.print("\033[H\033[2J");
            System.out.flush();
        }
    } catch (final Exception e) {
        //handle the exception
        System.err.println(e.getMessage());
    }
}
```

- You should write all the sorting algorithms by yourself. You should be able to explain the code that you have written.
- You can create as many methods as you wish in the only class that you have.
- Demo videos may help you to clarify your doubts to some extent.

Evaluation Criteria

Component	Marks
Application	50%
Individual Viva	50%

*Pass Mark: 50%

References

- 2D Arrays:
<https://www.freecodecamp.org/news/2d-array-in-java-two-dimensional-and-nested-arrays/>
- String to Integer converting:
<https://codegym.cc/groups/posts/string-to-int-java>
- Integer to String converting:
https://www.tutorialspoint.com/java/java_string_valueof.htm
- System.out.printf("");
<https://www.theserverside.com/blog/Coffee-Talk-Java-News-Stories-and-Opinions/How-to-use-Java printf-to-format-output>
- PRF theory interview question:
<https://docs.google.com/document/d/1PBAL8Q2Tu0gDBuKJYRdq9nP0w55CrllkO9OhPG-mRSq/edit?usp=sharing>

If you have still any questions, you know how to contact me. Right? 😊