

**Module Name - Fundamentals in Programming**

**Code - CSE4002**

**Assignment 01**

**Student Name : L D ISHANI HANSIKA JAYASEKARA**

**Student ID : CL-HDCSE-CMU-115-107**

**Assignment Cover Sheet**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | | **Module Number and Title** | |
| HD in Computing and Software Engineering /Network Technology and Cyber Security | | CSE 4002  Fundamentals in Programming | |
| **Student Name & No.** | | **Assessor** | |
| **Student Name : L D ISHANI HANSIKA JAYASEKARA**  **Student ID : CL-HDCSE-CMU-115-107** | |  | |
| **Hand over date** | | | **Submission Date** |
|  | | |  |
| **Assessment type** | **Duration/Length of**  **Assessment Type** | | **Weighting of Assessment** |
| **Coursework** | Software Submission and demonstration | | 100% |

|  |  |
| --- | --- |
| **Learner declaration** | |
| I, **L D ISHANI HANSIKA JAYASEKARA**.<name of the student and registration number>, certify that the work submitted for this assignment is my own and research sources are fully acknowledged. | |
| |  |  |  |  | | --- | --- | --- | --- | | **Marks Awarded** | | | | | First assessor | |  | | | IV marks | |  | | | Agreed grade | |  | | | Signature of the assessor |  | Date |  | |

**Feedback Form**

**International College of Business & Technology**

**Module :** CSE 4002

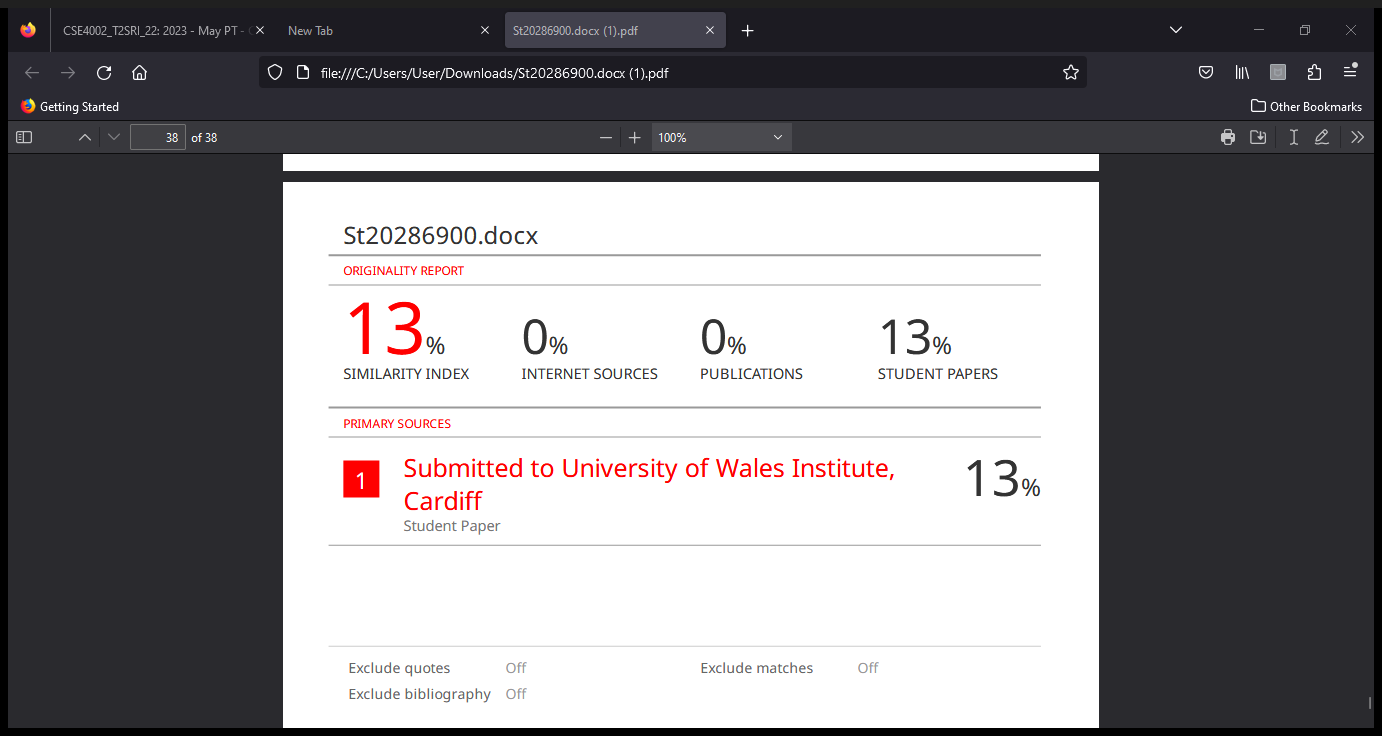
**Student : L D ISHANI HANSIKA JAYASEKARA**

**Assessor : Mrs. Nisansala Athapaththu**

**Assignment** **:** Bun Talk business automation system

**Assessor Feedback:**

**Marks Awarded:**



Contents

[**Introduction** 3](#_Toc149769109)

[**Structed Oriented Programming** 4](#_Toc149769110)

[**Object Oriented Programming** 4](#_Toc149769111)

[**Main Concepts of C++** 4](#_Toc149769112)

[**Bun Talk Bakery** 4](#_Toc149769113)

[**Introduction** 4](#_Toc149769114)

[**Purpose** 5](#_Toc149769115)

[**Scope** 5](#_Toc149769116)

[**1.Authentication** 5](#_Toc149769117)

[**2. Display Breakfast Items** 6](#_Toc149769118)

[**3. Selection of Items** 6](#_Toc149769119)

[**4. Calculation and Printing of Bill** 6](#_Toc149769120)

[**5. Help** 6](#_Toc149769121)

[**6. Exit** 6](#_Toc149769122)

[**Modularization** 6](#_Toc149769123)

[**Data Storage** 7](#_Toc149769124)

[**User-Friendly Interface** 7](#_Toc149769125)

[**Error Handling** 7](#_Toc149769126)

[**System Requirements Specification** 7](#_Toc149769127)

[**C++ CODE** 8](#_Toc149769128)

[**Pseudo Code For Each Function** 13](#_Toc149769129)

[**User Enter** 13](#_Toc149769130)

[**Enter Account** 13](#_Toc149769131)

[**Change Settings** 14](#_Toc149769132)

[**Add Bakery Item** 14](#_Toc149769133)

[**Employe Information** 15](#_Toc149769134)

[**Enter Cashier Information** 15](#_Toc149769135)

[**Cashier Settings** 16](#_Toc149769136)

[**Invalid Information** 16](#_Toc149769137)

[**bakery item** 17](#_Toc149769138)

[**Create Bill** 17](#_Toc149769139)

[**Main** 18](#_Toc149769140)

[**Flow chart for each function** 21](#_Toc149769141)

[**User Enter** 21](#_Toc149769142)

[**Enter Information** 22](#_Toc149769143)

[**Change Settings** 23](#_Toc149769144)

[**Enter Cashier Information** 24](#_Toc149769145)

[**Cashier Settings** 25](#_Toc149769146)

[**Invalid Information** 26](#_Toc149769147)

[**Test Plan** 27](#_Toc149769148)

[**Test Cases** 29](#_Toc149769149)

# **Introduction**

Since programming is used to construct websites, software applications, and a multitude of other tools that we use on a daily basis, programming is a crucial skill in today's culture. Programming is the process of designing, developing, testing, and maintaining computer programs to carry out specified tasks or solve certain problems.

## **Structed Oriented Programming**

Programming with well-defined structures is emphasized in the structured programming paradigm to increase programming's effectiveness, comprehensibility, and maintainability. Three basic structures are used: picking, repetition, and sequencing.

## **Object Oriented Programming**

Classes have instances that are objects. Code may be arranged into classes thanks to OOP. Classes are modular, reusable parts that can be used to create objects with specific properties and functions.

# **Main Concepts of C++**

Programming has three fundamental structures

* Repetition Structure
* Selection Structure
* Sequence Structure

# **Bun Talk Bakery**

## **Introduction**

The quickly expanding local bakery Bun Talk Bakery now uses manual cash book procedures to handle orders and billing. The manual method is laborious, prone to mistakes, and may result in dissatisfied customers because of errors. We suggest putting in place an automated bakery billing system to get around these obstacles and improve customer service. In order to increase operational effectiveness and customer happiness, this document describes the specific needs, goals, and scope of the project.

## **Purpose**

* This paper also acts as a reference manual for developers, testers, and project managers, among other project stakeholders.
* By automating and streamlining the bakery's invoicing procedure, the Bun Talk Bakery Automation System hopes to lower errors and give consumers better, more effective service.

## **Scope**

The following elements are part of the Bun Talk Bakery Automation System's scope

### **1.Authentication**

To provide access control, put in place a login mechanism that verifies both the username and the password.

### **2. Display Breakfast Items**

To show the things offered in the bakery, create a menu system. This information can be stored using vectors or arrays.

### **3. Selection of Items**

Permit the consumer to choose more than one item, keeping a record of their selections with data structures.

### **4. Calculation and Printing of Bill**

Determine the total cost of the chosen goods and produce an invoice.

To represent the objects, their names, pricing, and quantities, you can use a structure or class.

Continue to save the chosen items in a data structure (such as a vector).

### **5. Help**

Provide instructions for using the application through the implementation of a help component.

### **6. Exit**

Let users get out of the application

### **Modularization**

To cut down on complexity, group your code into functions or classes.For every task listed above, create a separate function.For declarations and implementations, use different header and source files.

### **Data Storage**

Information like menu items, their prices, and authentication credentials can be stored in text files.Utilize these files for reading and writing as required.

### **User-Friendly Interface**

Assist the user in navigating the program by providing clear and succinct messages.Provide a menu-driven interface that is easy for users to utilize to move through various options.

### **Error Handling**

To deal with problems like inaccurate input, illegal access, or insufficient money for the order, provide error handling.

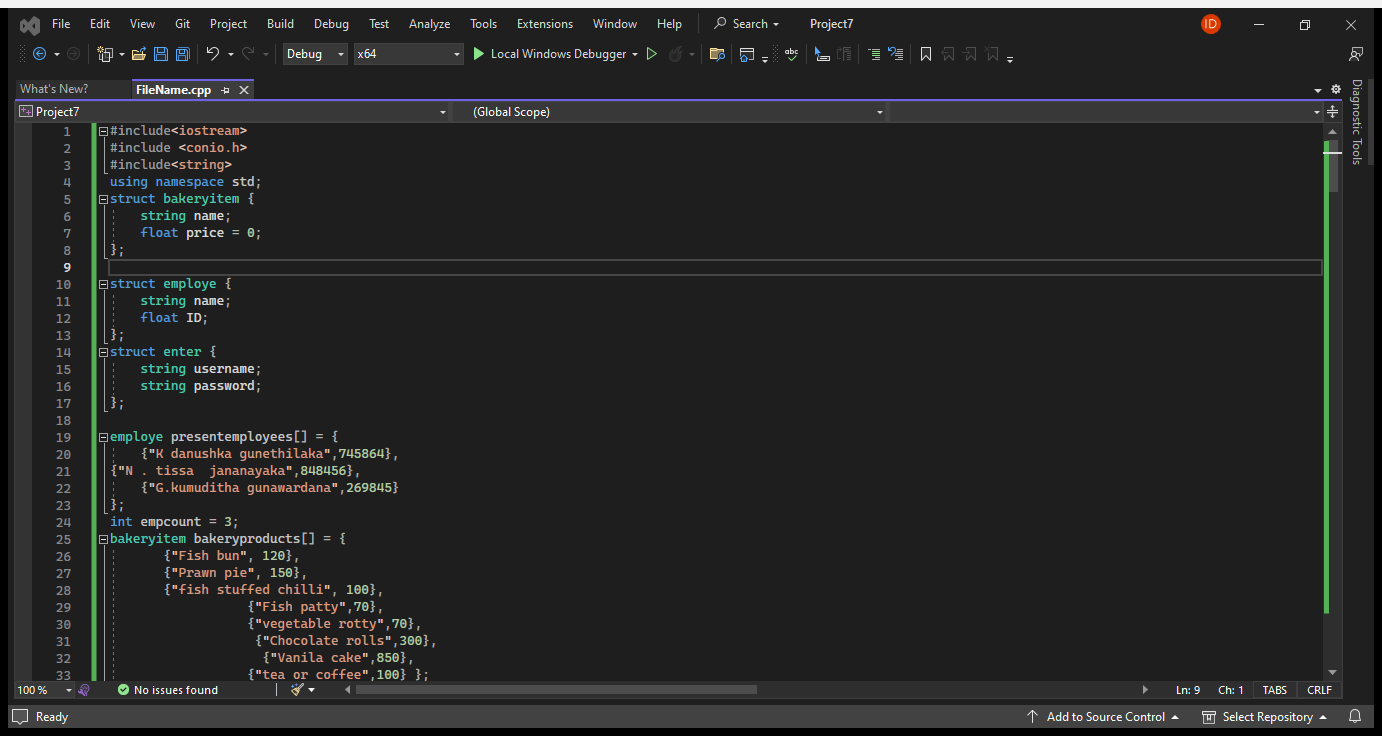
### **System Requirements Specification**

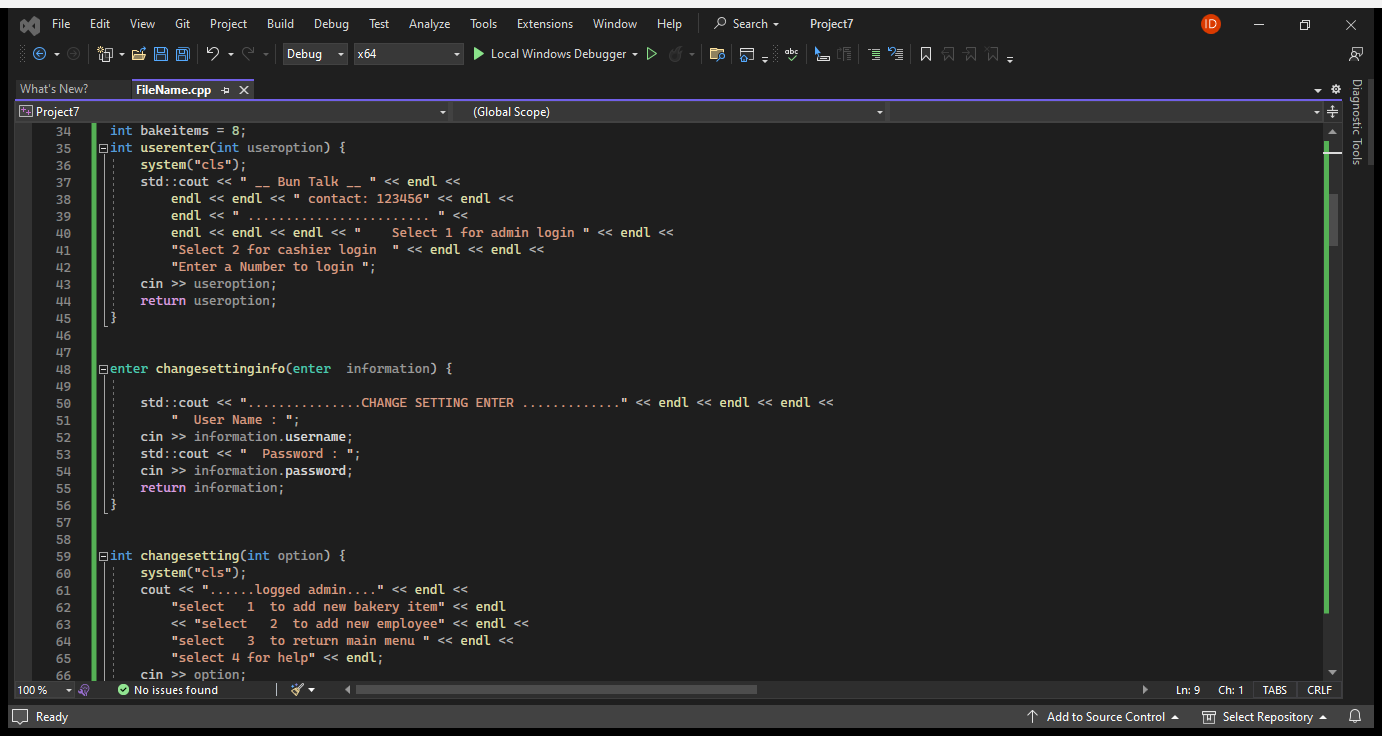
Including the minimal system needs, create a comprehensive document outlining the system's hardware and software requirements.

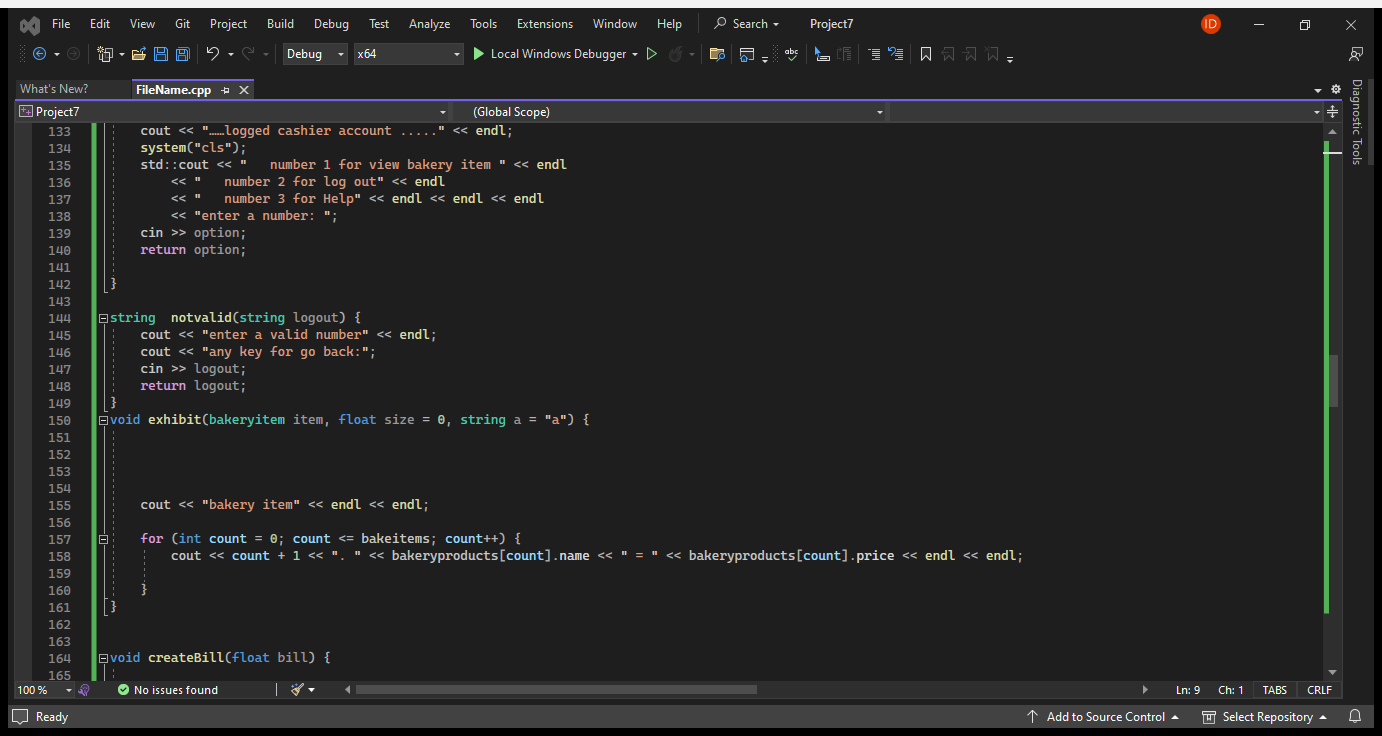
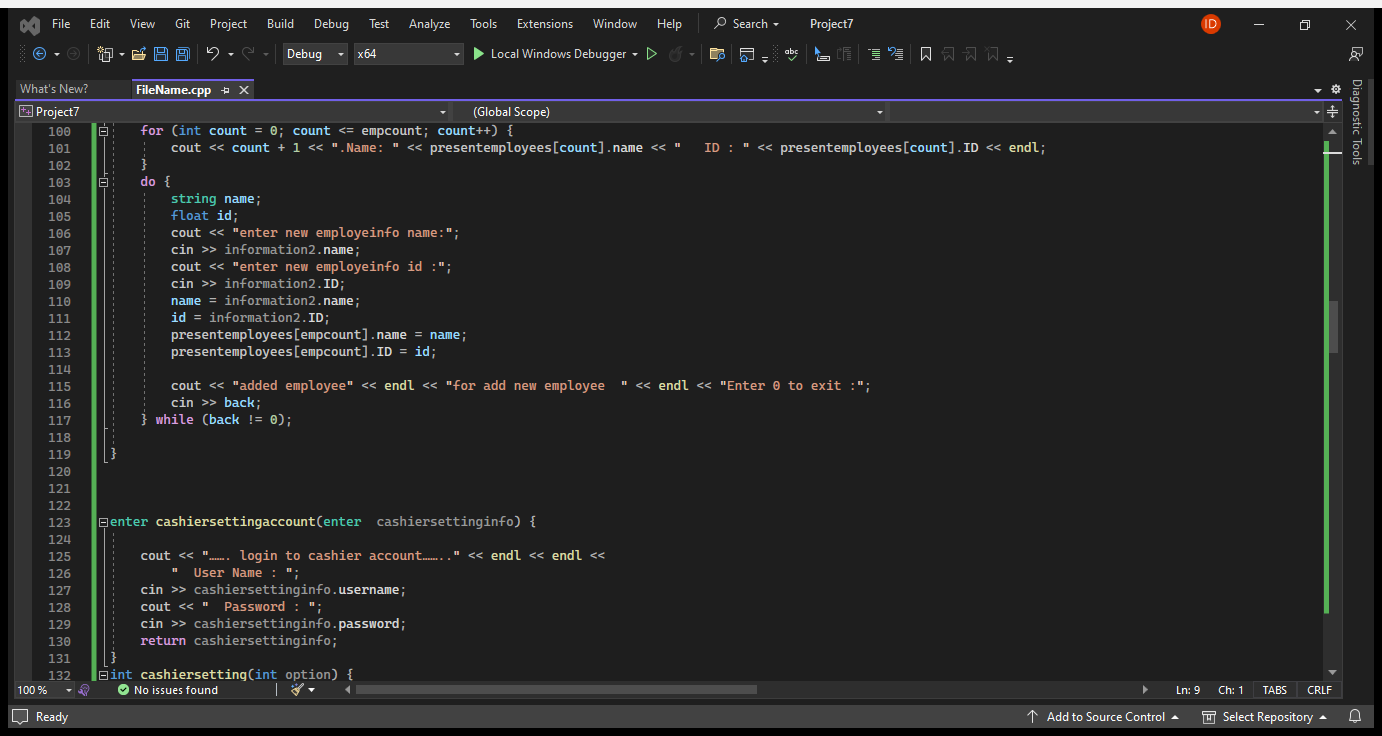
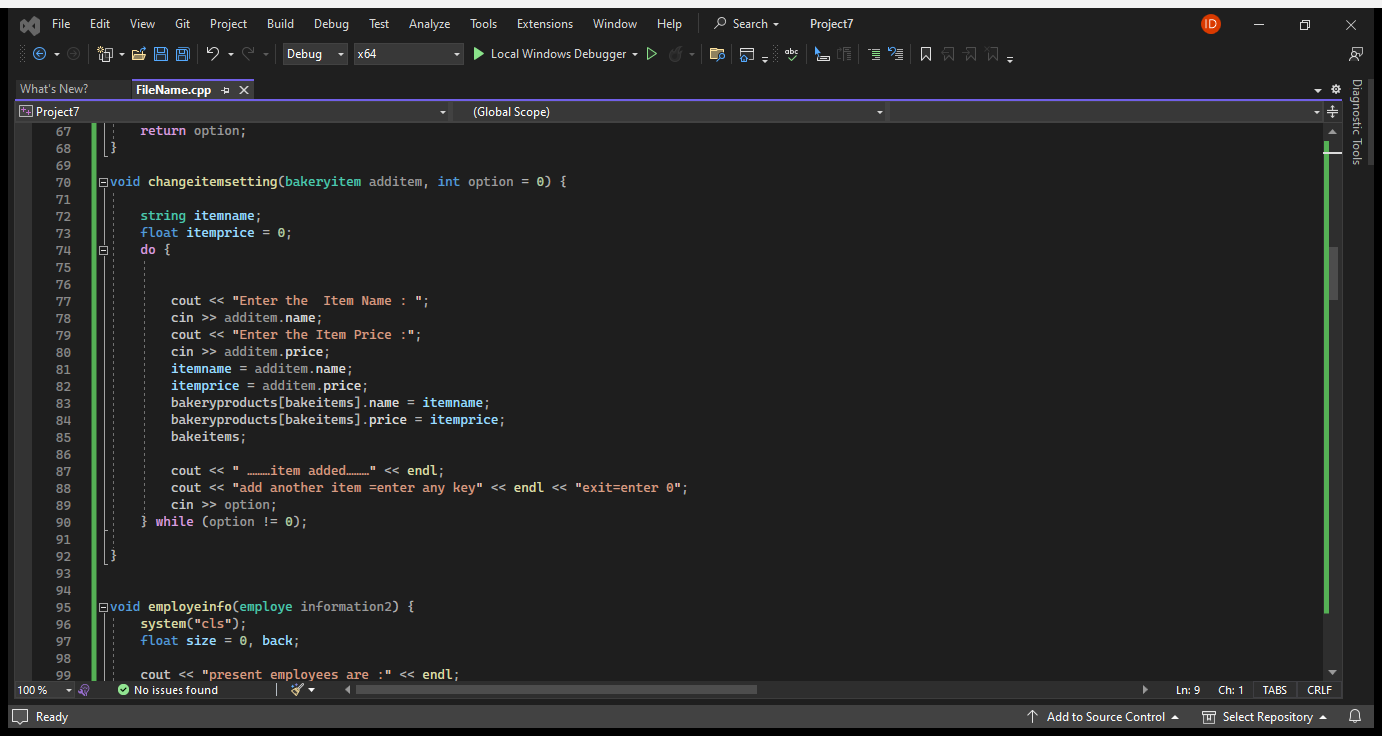
Diagrams with logic:

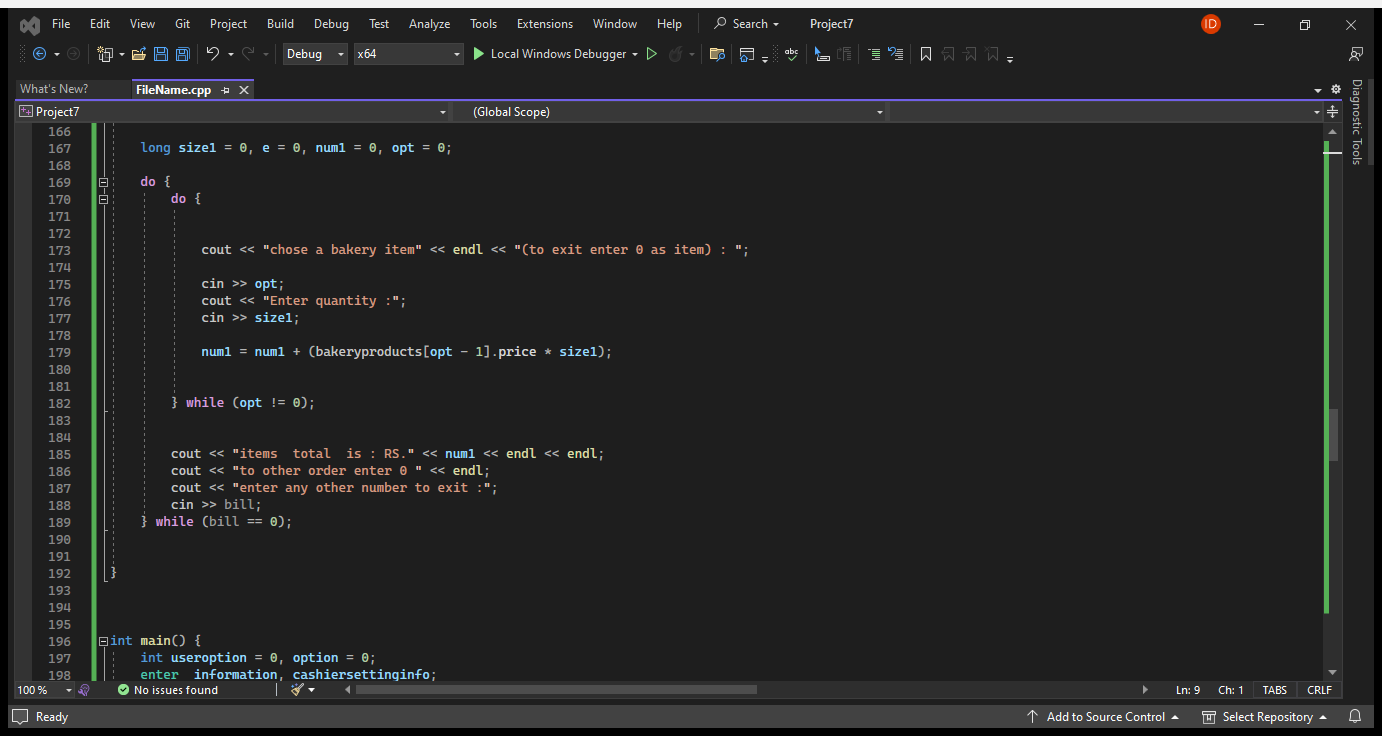
To demonstrate the logical progression of essential features such as menu display, bill computation, and login, create flowcharts or UML diagrams.

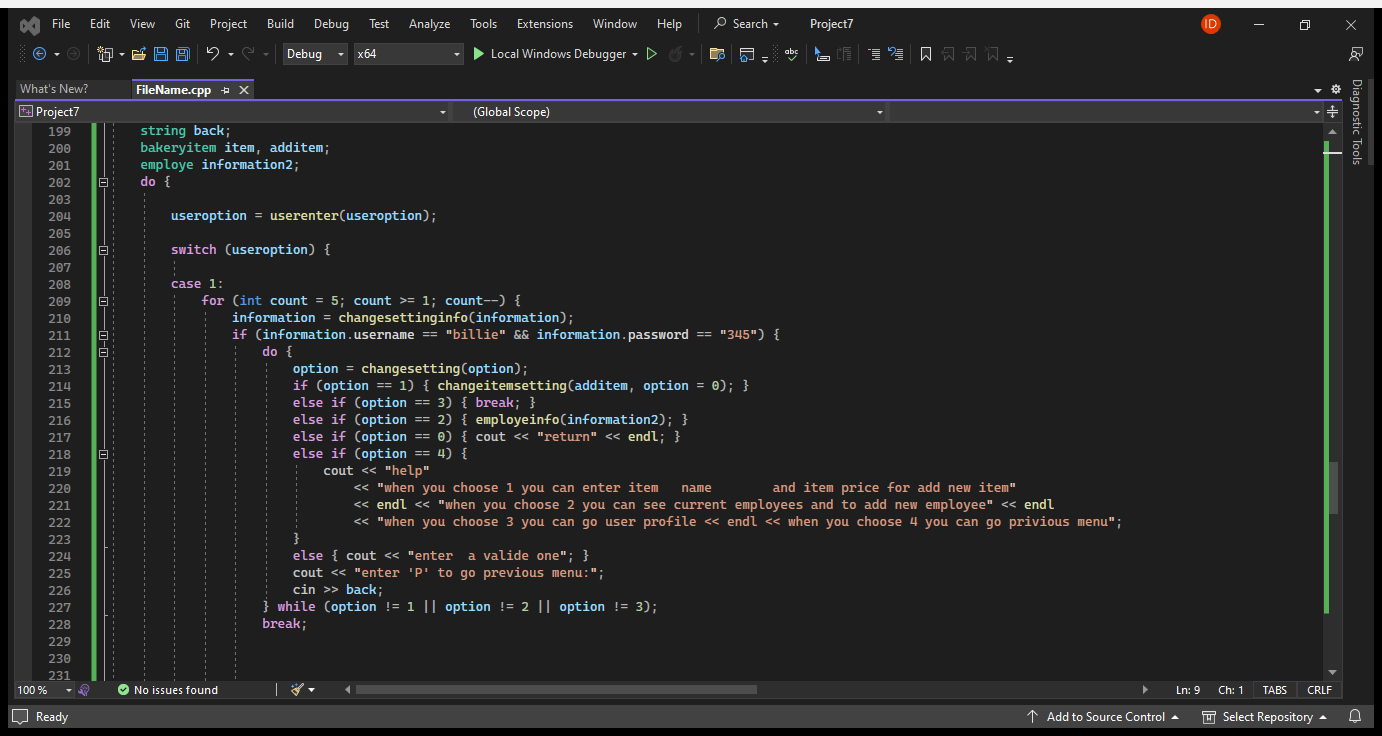
# **C++ CODE**

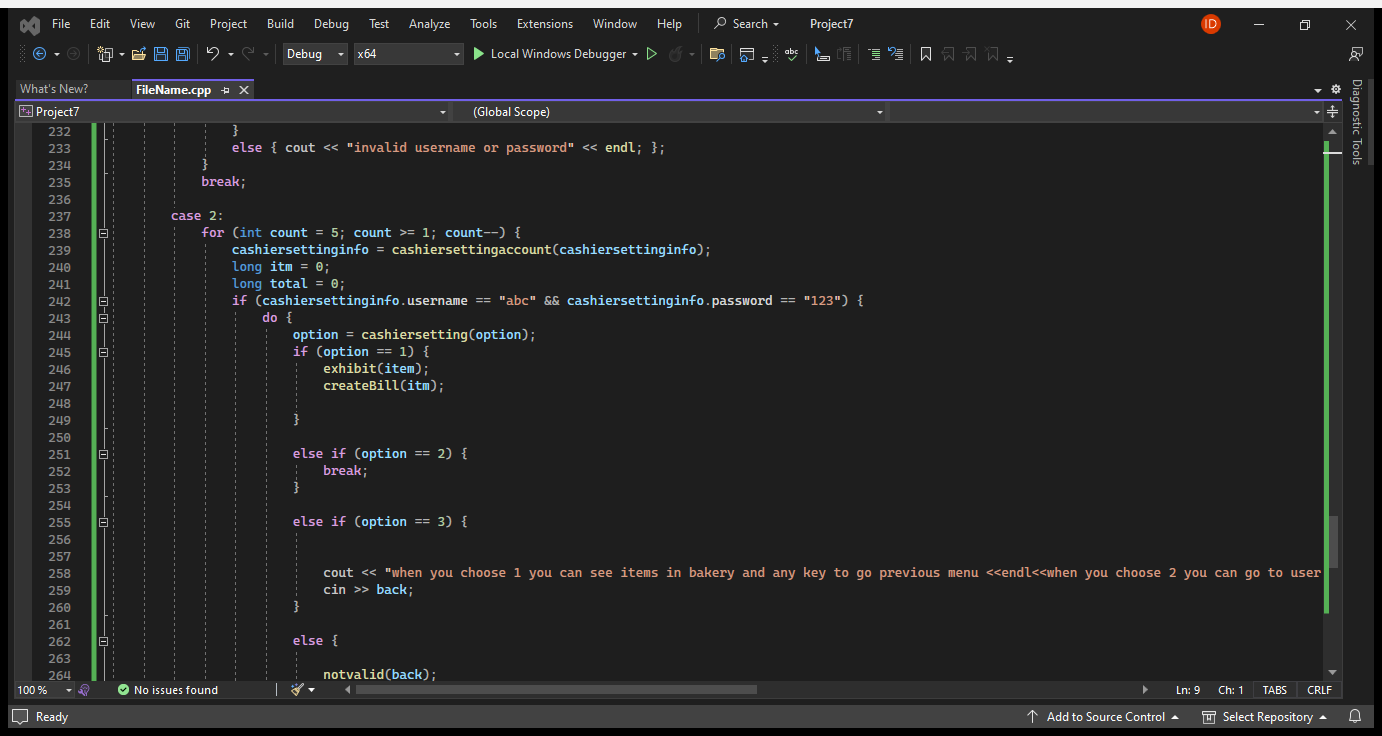


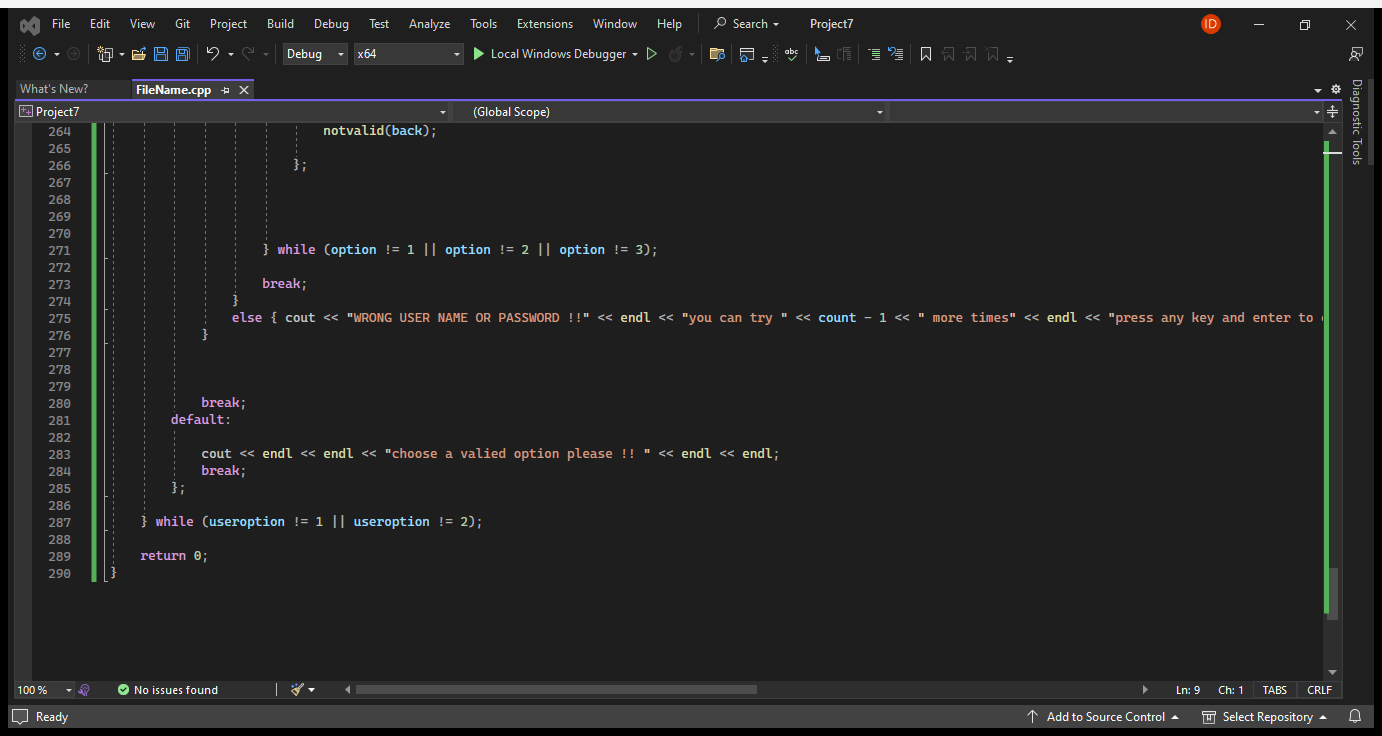












# **Pseudo Code For Each Function**

## **User Enter**

Function userenter(useroption)

Display "Select 1 for admin login"

Display "Select 2 for cashier login"

Display "Enter a Number to login"

Input useroption

Return useroption

End Function

## **Enter Account**

Function changesettinginfo(information)

Display "User Name : "

Input information.username

Display "Password : "

Input information.password

Return information

End Function

## **Change Settings**

Function changesetting(option)

Display "select 1 to add new bakery item"

Display "select 2 to add new employee"

Display "select 3 to return main menu"

Display "select 4 for help"

Input option

Return option

End Function

## **Add Bakery Item**

Function changeitemsetting(additem, option)

Do

Display "Enter the Item Name : "

Input additem.name

Display "Enter the Item Price :"

Input additem.price

Add additem to bakeryproducts array

Display "…item added…"

Display "add another item = enter any key"

Display "exit = enter 0"

Input option

While option is not equal to 0

End Function

## **Employe Information**

Function employeinfo(information2)

Display "present employees are :"

For each employe in presentemployees

Display employe.name and employe.ID

End For

Do

Input information2.name

Input information2.ID

Add information2 to presentemployees array

Display "added employee"

Display "for add new employee"

Display "Enter 0 to exit"

Input back

While back is not equal to 0

End Function

## **Enter Cashier Information**

Function cashiersettingaccount(cashiersettinginfo)

Display "User Name : "

Input cashiersettinginfo.username

Display "Password : "

Input cashiersettinginfo.password

Return cashiersettinginfo

End Function

## **Cashier Settings**

Function cashiersetting(option)

Display "number 1 for view bakery item"

Display "number 2 for log out"

Display "number 3 for Help"

Input option

Return option

End

## **Invalid Information**

Function notvalid(logout)

Display "enter a valid number"

Display "any key for go back"

Input logout

Return logout

End Function

## **bakery item**

Function exhibit(item, size, a)

Display "bakery item"

For each bakeryitem in bakeryproducts

Display bakeryitem.name and bakeryitem.price

End For

End Function

## **Create Bill**

Function createBill(bill)

Do

Do

Input opt

Input size1

Calculate num1 as bakeryproducts[opt - 1].price \* size1

Add num1 to total

While opt is not equal to 0

Display "items total is: RS." and total

Display "to other order enter 0"

Display "enter any other number to exit"

Input bill

While bill is equal to 0

End Function

## **Main**

Function main()

Declare useroption, option as integers

Declare information, cashiersettinginfo as enter

Declare back as string

Declare item, additem as bakeryitem

Declare information2 as employe

Do

Input useroption

Switch useroption

Case 1:

For count from 5 down to 1

Call changesettinginfo(information)

If information.username is "billie" and information.password is "345"

Do

Input option

If option is 1

Call changeitemsetting(additem)

Else If option is 2

Call employeinfo(information2)

Else If option is 3

Exit loop

Else If option is 4

Display help message

Else

Display "enter a valid one"

End If

Input back

While option is not 1 or option is not 2 or option is not 3

Exit loop

Else

Display "invalid username or password"

End If

End For

Exit loop

Case 2:

For count from 5 down to 1

Call cashiersettingaccount(cashiersettinginfo)

If cashiersettinginfo.username is "abc" and cashiersettinginfo.password is "123"

Do

Input option

If option is 1

Call exhibit(item)

Call createBill(itm)

Else If option is 2

Exit loop

Else If option is 3

Display help message

Input back

Else

Call notvalid(back)

End If

While option is not 1 or option is not 2 or option is not 3

Exit loop

Else

Display "WRONG USER NAME OR PASSWORD !!"

End If

End For

Exit loop

Default:

Display "choose a valid option please !!"

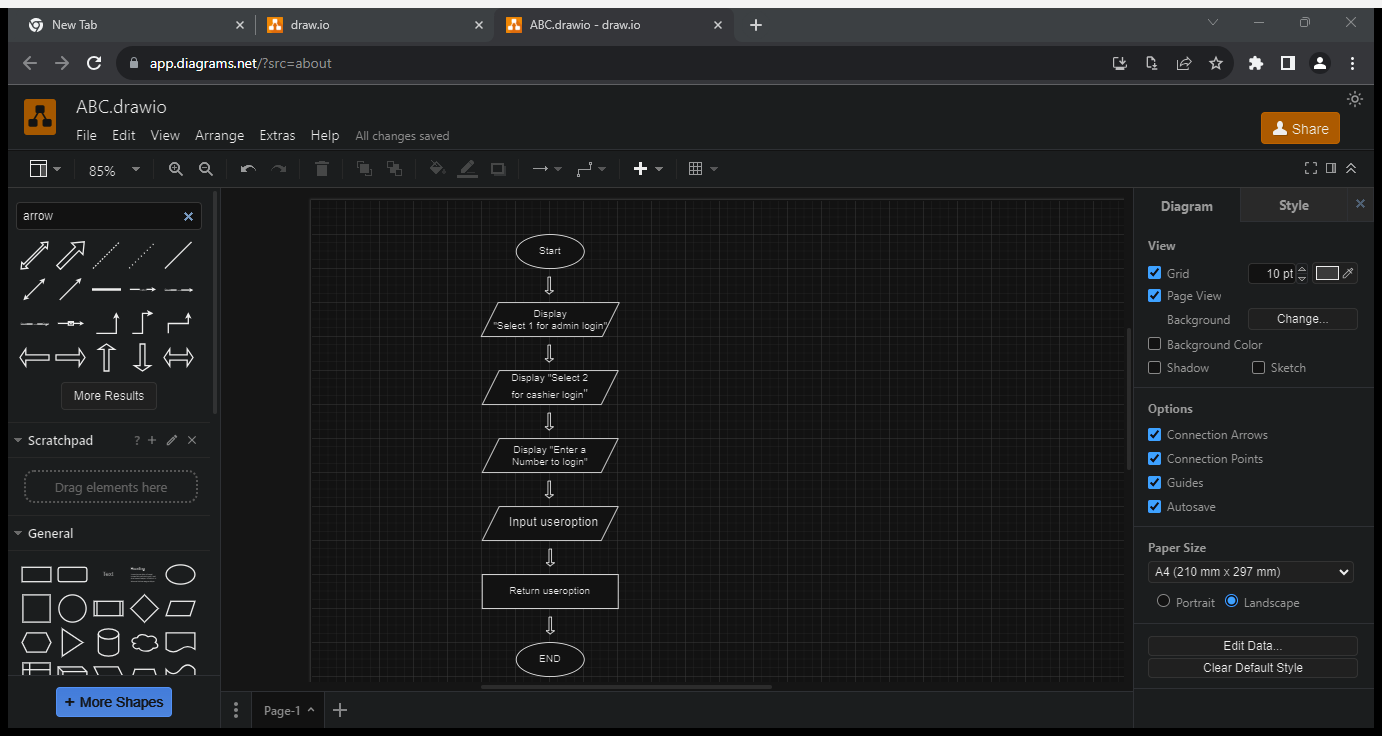
End Switch

While useroption is not 1 or useroption is not 2

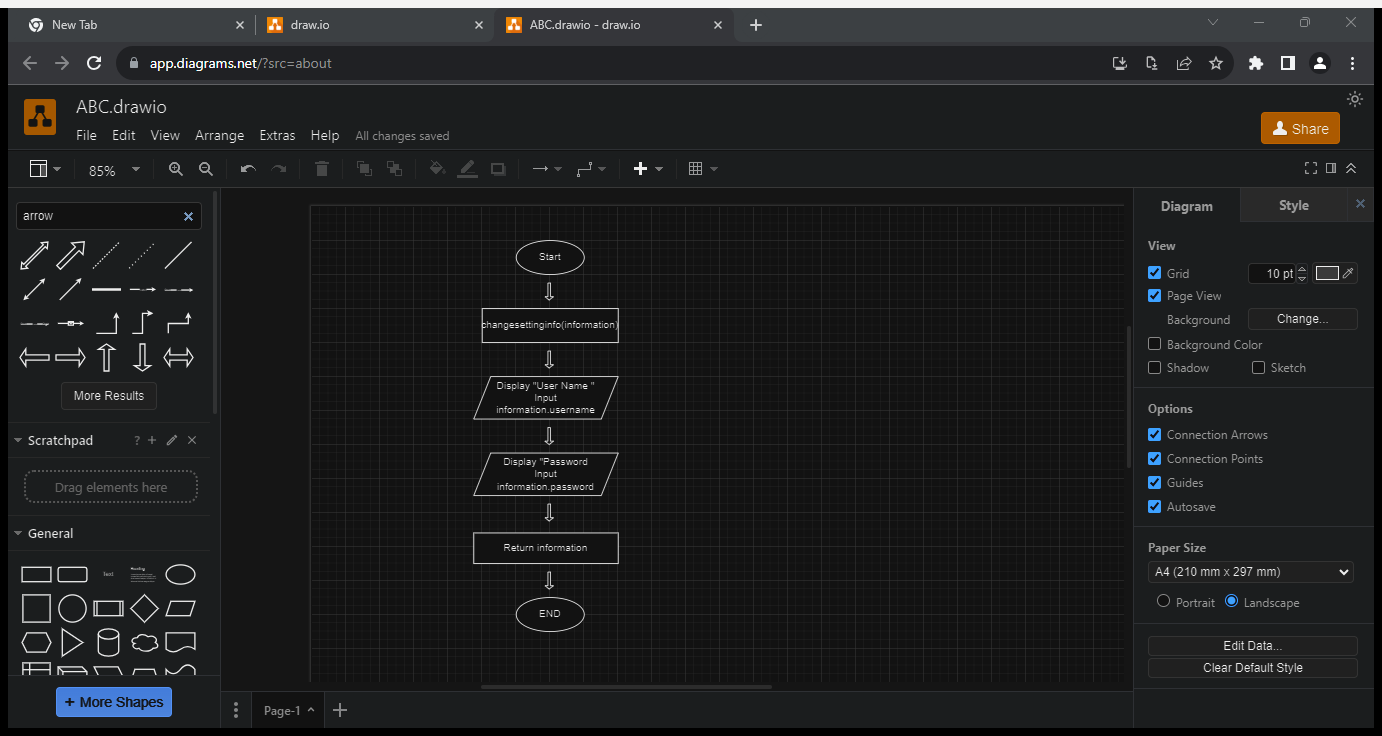
End Function

# **Flow chart for each function**

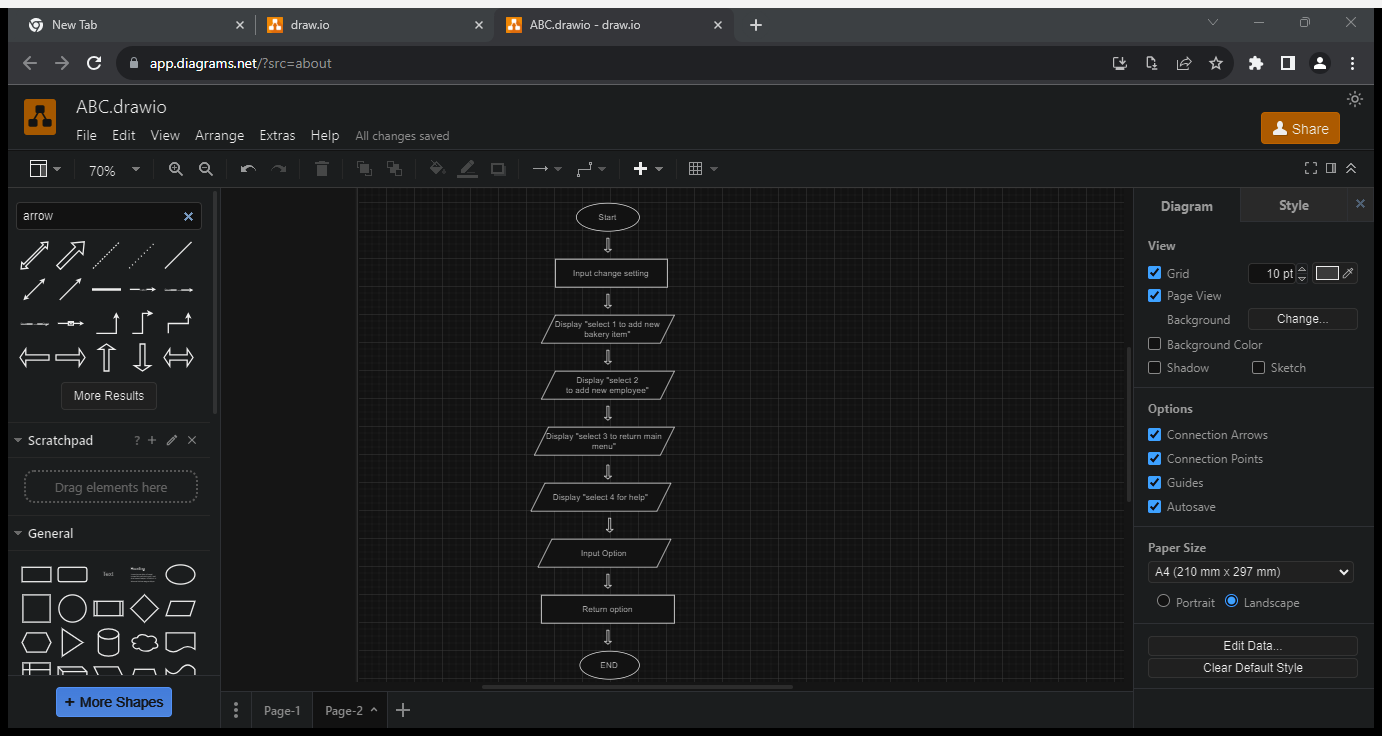
# **User Enter**



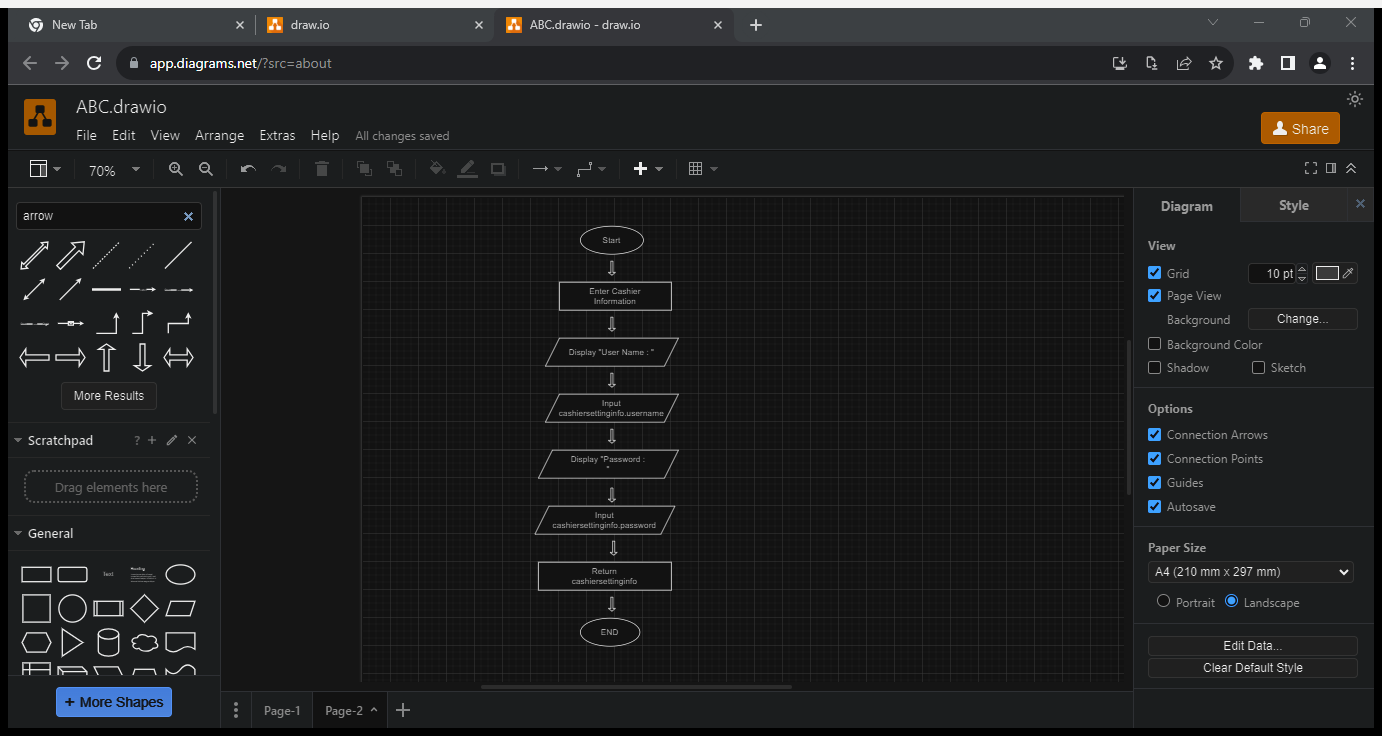
## **Enter Information**



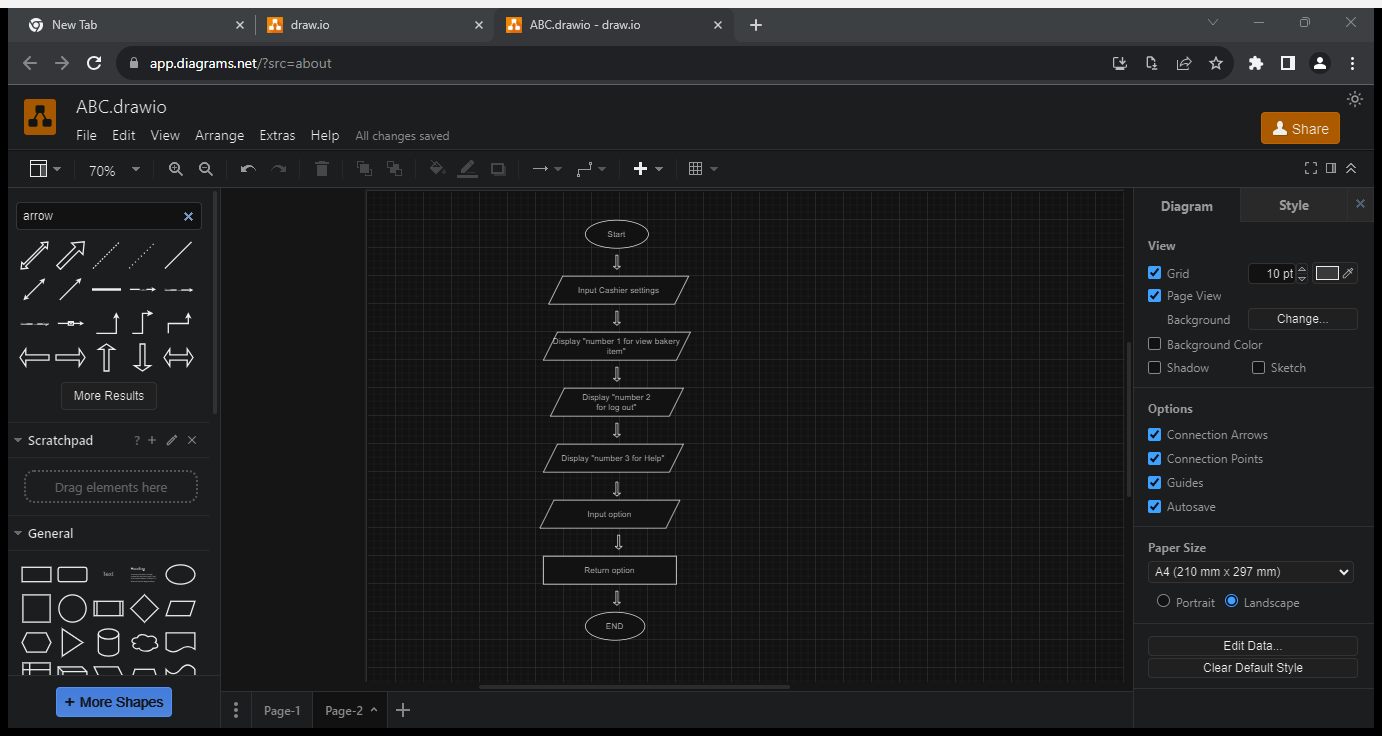
## **Change Settings**



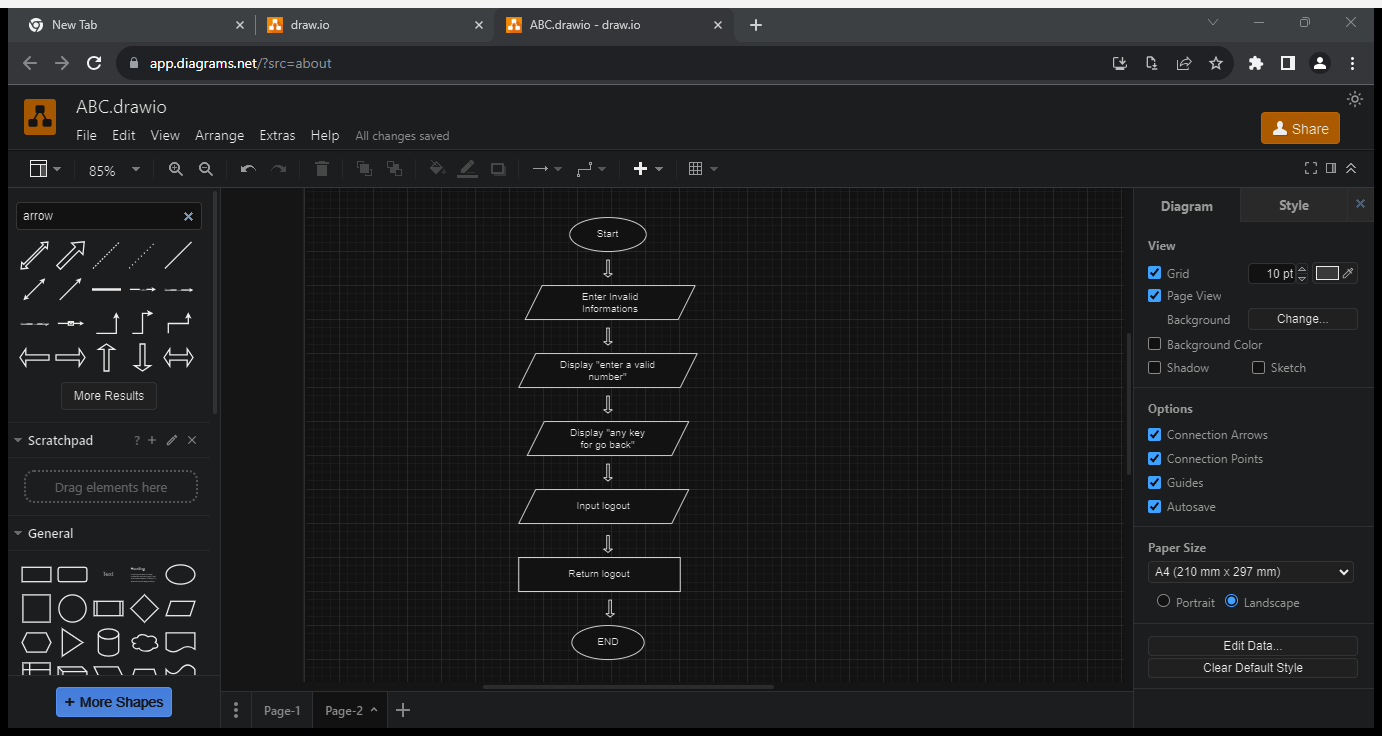
## **Enter Cashier Information**



## **Cashier Settings**



## **Invalid Information**



# **Test Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| Case ID | Case Name | Senario | Expected Result |
| **01** | First page opening | Select desired  account to loggin | To see Bun talk  details and,cashier  accounts to choose |
| **02** | Admin Login - 1 | Correct Username  and Password | To see log in page |
| Incorrect Username  and Password | Error Message |
| **03** | Cashier Login - 2 | Correct Username  and Password | To see log in page |
| Incorrect Username  and Password | Error Message |
| **04** | Admin  Change settings | Select Add new  bakery items | To add new items |
| **05** | Add employee | Add multiple  Employees | To add new  employee |
| **06** | Admin  Change settings | Back to the main  menu | To return main menu |
| **07** | Admin  Change settings | Help optiom | Help instructions  for admin |
| **08** | Cashier  Change settings | Cashier change  settings to view  bakery items | To display bakery  items |
| **09** | Cashier  Change settings | Logout | Logout and again  login |
| **10** | Cashier  Change settings | Help | To get help  Information |
| **11** | Exit | Enter invalid option | To Exit |

# **Test Cases**

|  |  |
| --- | --- |
| Test Case ID | 01 |
| Test Name | First page opening |
| Input Data | Nothing |
| Expected Result | Open as expected |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Program opened successfully |

|  |  |
| --- | --- |
| Test Case ID | 2 |
| Test Name | Select a account |
| Input Data | Enter “1” for login to admin account |
| Expected Result | Open Admin login |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Opened as expected |

|  |  |
| --- | --- |
| Test Case ID | 2.1 |
| Test Name | Admin Login - 1 |
| Input Data | Enter Correct Username and Password  Username : “Bille”  Password : “345” |
| Expected Result | Open login menu and ask for user name and pass word |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Opened as expected |

|  |  |
| --- | --- |
| Test Case ID | 2.2 |
| Test Name | Wrong login info error |
| Input Data | Enter Incorrect Username and Password  Username = “ishani”  Password = “1230” |
| Expected Result | Wrong login info error Message |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Expected error given successfully |

|  |  |
| --- | --- |
| Test Case ID | 3 |
| Test Name | Choose a account |
| Input Data | Enter “2” for login to cachier account |
| Expected Result | Open cashier login |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Opened as expected |

|  |  |
| --- | --- |
| Test Case ID | 3.1 |
| Test Name | Cachier Account - 2 |
| Input Data | Enter Correct Username and Password  Username : “abc”  Password : “123” |
| Expected Result | Open login menu and ask for user name and pass word |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Opened as expected |

|  |  |
| --- | --- |
| Test Case ID | 3.2 |
| Test Name | Wrong login info error |
| Input Data | Enter Incorrect Username and Password  Username = “ishani”  Password = “567” |
| Expected Result | Wrong login info error Message |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Expected error given successfully |

|  |  |
| --- | --- |
| Test Case ID | 4 |
| Test Name | Admin - Change settings (Add new bakery item view item list with categories) |
| Input Data | Enter “1” Add new bakery items |
| Expected Result | First View available bakery item list |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Opened as expected |
| Test Case ID | 5 |
| Test Name | Admin - Change settings (Add new add new employee) |
| Input Data | Enter “2” Add new add new employee |
| Expected Result | Display current employee list |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Displayed as expected |

|  |  |
| --- | --- |
| Test Case ID | 6 |
| Test Name | Admin - Change settings (Back to the main menu) |
| Input Data | Enter “3” to go to the main menu |
| Expected Result | Going back to main menu |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Displayed as expected |

|  |  |
| --- | --- |
| Test Case ID | 7 |
| Test Name | Admin - Change settings (Admin help option) |
| Input Data | Enter “4” to go to the help option |
| Expected Result | Help instructions for admin |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Explaind help option is given |

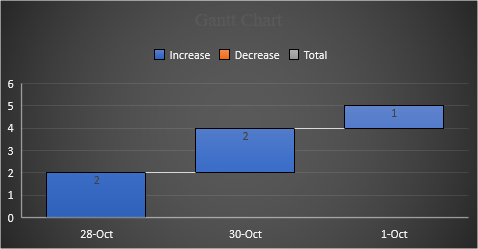
|  |  |
| --- | --- |
| Test Case ID | 8 |
| Test Name | Cashier - Change settings ( view bakery items) |
| Input Data | Enter “1” for view bakery items |
| Expected Result | TO Display bakery items |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Displayed as expected |

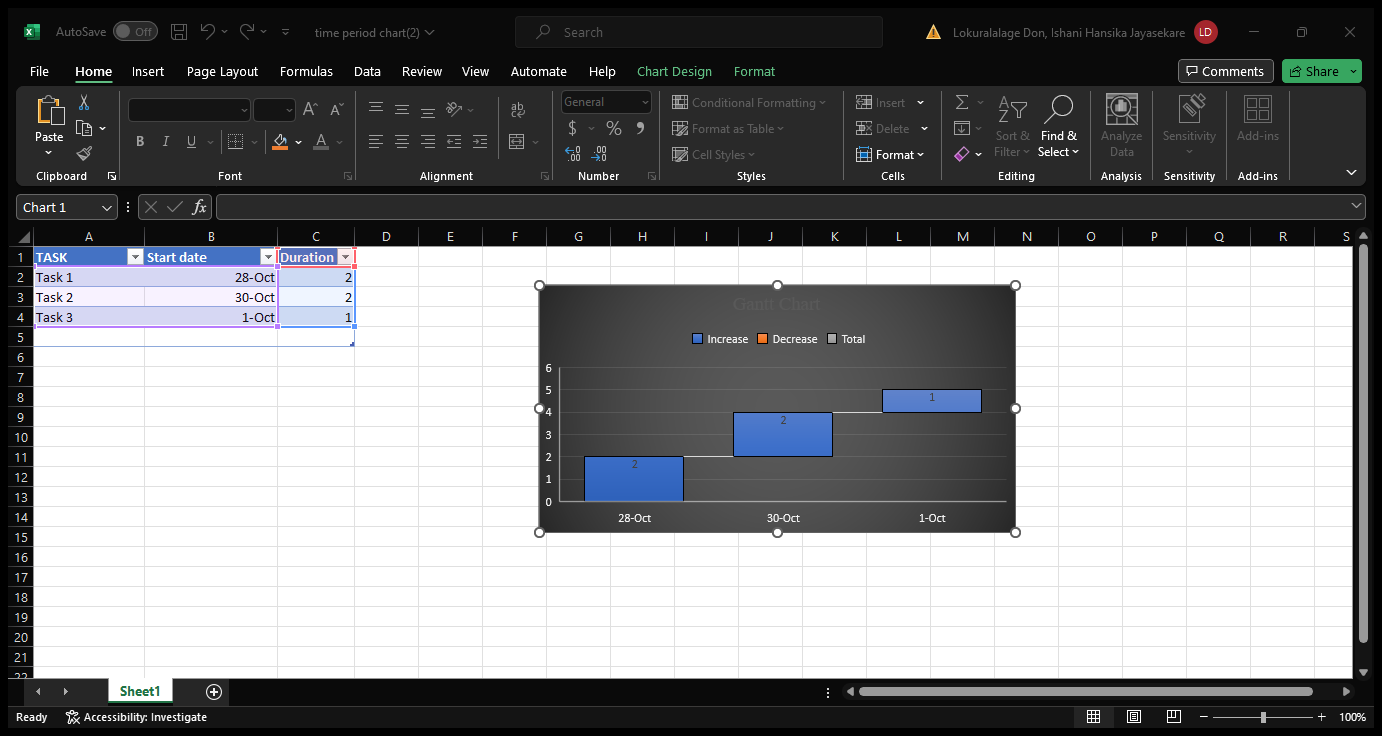
|  |  |
| --- | --- |
| Test Case ID | 9 |
| Test Name | Login from the cashier account |
| Input Data | Enter “2” Logout |
| Expected Result | Logout and again login |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | logout as expected |

|  |  |
| --- | --- |
| Test Case ID | 10 |
| Test Name | Cashier - Change settings (Cashier help option) |
| Input Data | Enter “3” to go to the help option |
| Expected Result | Help instructions for admin |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Explaind help option is given |

|  |  |
| --- | --- |
| Test Case ID | 12 |
| Test Name | Exit |
| Input Data | Enter “0” for Exit |
| Expected Result | Exit |
| Screenshot | C:\Users\User\source\repos\Project8\x64\Debug\Project8.exe |
| Conclusion | Displayed as expected |

# **Gantt Chart**





# References

Anon., 2023. *asq.org.* [Online]   
Available at: https://asq.org/quality-resources/flowchart  
[Accessed octomber 2023].

Anon., 2023. *lucidchart.* [Online]   
Available at: https://www.lucidchart.com/pages/what-is-a-flowchart-tutorial

Anon., 2023. *programiz.* [Online]   
Available at: https://www.programiz.com/cpp-programming

Anon., 2023. *tutorialspoint..* [Online]   
Available at: https://www.tutorialspoint.com/cplusplus/index.htm

t, 2023. *techtarge.* [Online]   
Available at: https://www.techtarget.com/whatis/definition/pseudocode