



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

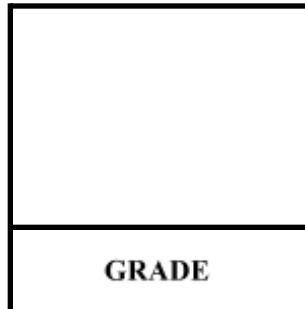
Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

INTERMEDIATE PROGRAMMING

(ITC123L)

FOOD ORDERING SYSTEM



Submitted by:



Almirol, Nadine B.

BSIT

CEIT-37-202A



Castañeda, John Lloyd C.

BSIT

CEIT-37-202A



Culasing, Jayrene G.

BSIT



Maluping, Crystal Angel

BSIT

CEIT-37-202A



Mataverde, Athena Bea R.

BSIT

CEIT-37-202A



Moriones, Rose-Ann S.

BSIT

CEIT-37-202A



Santillan, Kuzui R.

BSIT

CEIT-37-202A



Vinluan, Deserre C.

BSIT

CEIT-37-202A

Submitted to:

MAY FIGUEROA

Professor

May 23, 2023



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

TABLE OF CONTENTS

I.	Introduction	3
II.	Description of the project	3
III.	Objectives	4
IV.	Significance of the study	4
V.	Scope and Delimitations	5
VI.	Screen output	6
VII.	Source code	12
VIII.	Conclusion	48



I. INTRODUCTION

In today's rapidly evolving world, the significance of technology cannot be understated. It has become fundamental to our daily lives, offering improved accessibility and transformative experiences. The team recognized the power of technology and design to revolutionize the way food businesses interact with customers. Thus, the team embarked on a journey to create a cutting-edge system featuring a user-friendly graphical user interface (GUI) design and real-time time and date updates.

Our innovative system aims to empower food businesses by providing them with a dynamic interface that streamlines operations and enhances customer satisfaction. With just a few taps, restaurants can efficiently process orders and serve their customers promptly, saving valuable time and effort. By leveraging our system, establishments can boost productivity and create a seamless experience for both customers and staff.

II. DESCRIPTION OF THE PROJECT

The self-service food ordering system offers a menu interface that allows customers to choose and place their orders in any quantity and size they wish. The approach is similar to the standard fast food ordering process, which does not involve the usage of accounts. The system's menu is made up of cream cheese, milk tea series, and fruit tea series, all of which come in a wide variety. The system automatically calculates the order's total price, afterwards, it prints the reference number and the order's total for the customer to pay at the cash register.



III. OBJECTIVES

The main objective of the Food Ordering System is to provide customers the freedom to choose from the menu, have the cost calculated automatically, and receive a printed order of reference number and the order's total.

The following are the objectives of Food Ordering System:

1. To develop a program using C++ that is highly efficient in ordering drinks and cream cheese.
2. To improve the system's presentation using GUI design. GUI adds visual appeal and a user-friendly interface with icons, buttons, menus, and colors, resulting in an interactive and attractive user experience.
3. To implement the C++ system program using ideas such as designing the framework, structure, user interaction, computation of the payment and purchases' total cost.

IV. SIGNIFICANCE OF THE STUDY

The significance of our program is that it will provide users with a functional system that facilitates smooth ordering transactions, eliminating interruptions, and lengthy conversations that cater to their needs. With the menu's milk tea, fruit tea, and cream cheese available, customers can easily interact with the program when placing their desired order. Through this system, customers can conveniently order in any quantity and size they wish and choose from three categories of beverages.

This paper serves to offer guidance and insights for developing an effective food ordering system. It aims to broaden the understanding of individuals dealing with ordering-related issues, providing them with an opportunity to comprehend the current reality and explore possible solutions.



V. SCOPE AND DELIMITATIONS

The scope of this study is to design, and develop a system that can be used by the company and its customers to order food through a program. A GUI is used to design the system as well as to make it simpler to apply graphical icons. The study is focused on providing a good service to customers who purchase food online using a food ordering system. It also aims to develop a method for customers to order food that is simple for them to use and offers them good service.

The study is delimited only to the customers of “YummyTea Shop” in which they can access the program and it will show or display the welcome screen of the shop and it will display the menu that the user can choose from, place their order and print a ticket that contains the total amount of their order alongside the reference number. The customer cannot remove a specific order once it has been placed if that is the item they ordered.



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

IV. SCREEN OUTPUT

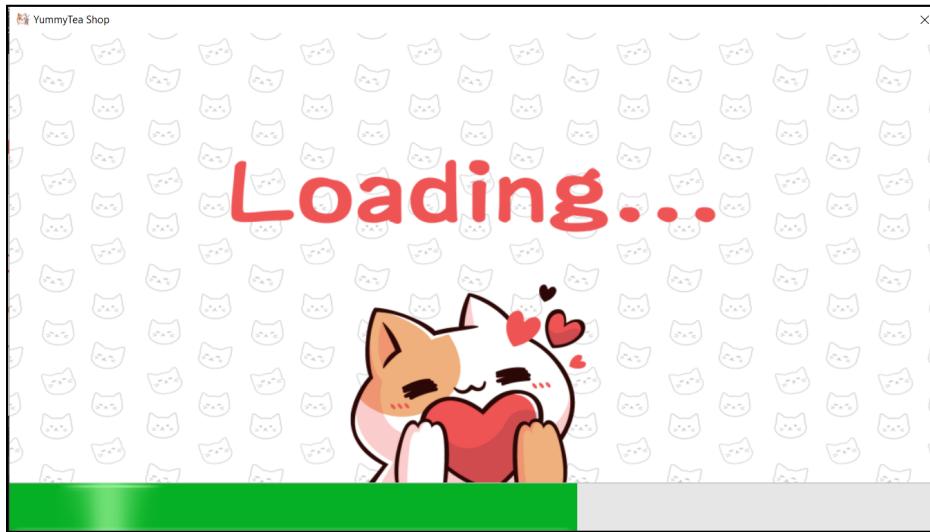


Figure 1. Loading Screen

Figure 1 displays the loading screen of the shop.



Figure 2. Welcome screen

Figure 2 displays the Welcome screen, featuring a start button that allows users to access the menu form.

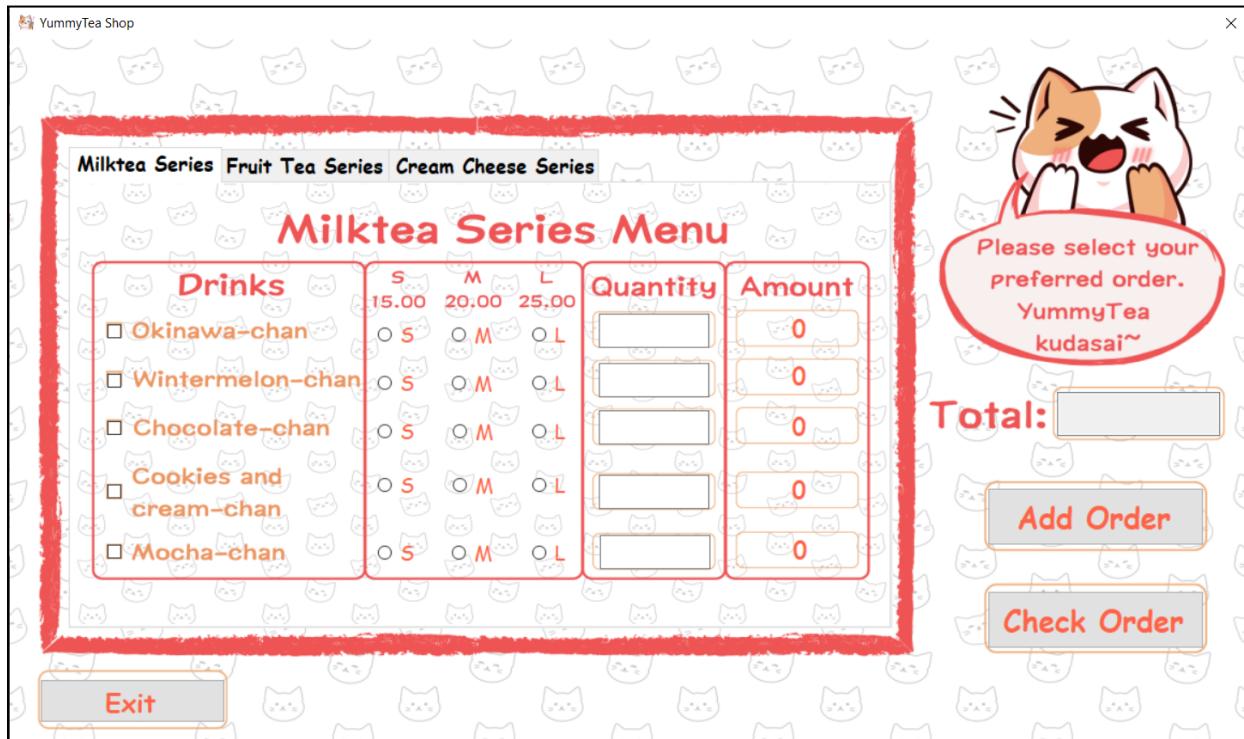


Figure 3. Menu for each category

Figure 3 displays the menus for Milk Tea, Fruit Tea, and Cream Cheese Series, accompanied by interactive components such as buttons, text boxes, radio buttons, and check boxes. These interactive components enable user engagement and include the following functionalities:

1. **ADD ORDER** – This button allows users to save or add their orders.
2. **CHECK ORDER** – Selecting this option displays a list of the user's orders.
3. **TOTAL** - This section presents the total amount for the user's order.
4. **AMOUNT** - This field indicates the cost of the user's orders.
5. **QUANTITY** - Users can enter the desired quantity of their orders in this input box.
6. **DRINKS** - This feature enables users to choose their preferred drink.
7. **SIZE** - Users can select the size (Small/Medium/Large) they desire for their drink.
8. **EXIT** – Choosing this option allows users to exit the system.



The screenshot shows the 'Milktea Series Menu' section of the YummyTea Shop application. At the top, there are three tabs: 'Milktea Series', 'Fruit Tea Series', and 'Cream Cheese Series'. The 'Milktea Series' tab is selected. Below the tabs, there is a title 'Milktea Series Menu' and a section for 'Drinks'. The drink list includes: Okinawa-chan (selected), Wintermelon-chan, Chocolate-chan, Cookies and cream-chan, and Mocha-chan. To the right of the drink list is a size selection grid with columns for S, M, and L, and rows for 15.00, 20.00, and 25.00. Next to the grid is a 'Quantity' column with five input fields, the first of which contains the value '1'. To the right of the quantity is an 'Amount' column with five input fields, the first of which contains the value '15'. On the far right, there is a cartoon dog character with a speech bubble saying 'Please select your preferred order. YummyTea kudasai~'. Below the dog are buttons for 'Total:', 'Add Order', and 'Check Order'. At the bottom left is an 'Exit' button.

Figure 4. Ordering process of Milktea Series Menu

Figure 4 illustrates the ordering process of the Milktea Series Menu and its functioning.

The screenshot shows the 'Fruit tea Series Menu' section of the YummyTea Shop application. At the top, there are three tabs: 'Milktea Series', 'Fruit Tea Series', and 'Cream Cheese Series'. The 'Fruit Tea Series' tab is selected. Below the tabs, there is a title 'Fruit Tea Series Menu' and a section for 'Drinks'. The drink list includes: Lychee-kun (selected), Strawberry-kun, Green Apple-kun, Yakult kun, and Mango-kun. To the right of the drink list is a size selection grid with columns for S, M, and L, and rows for 15.00, 20.00, and 25.00. Next to the grid is a 'Quantity' column with five input fields, the first of which contains the value '1'. To the right of the quantity is an 'Amount' column with five input fields, the first of which contains the value '20'. On the far right, there is a cartoon dog character with a speech bubble saying 'Please select your preferred order. YummyTea kudasai~'. Below the dog are buttons for 'Total:', 'Add Order', and 'Check Order'. At the bottom left is an 'Exit' button.

Figure 5. Ordering process of Fruit tea Series Menu

Figure 5 illustrates the ordering process of the Fruit tea Series Menu and its functioning.

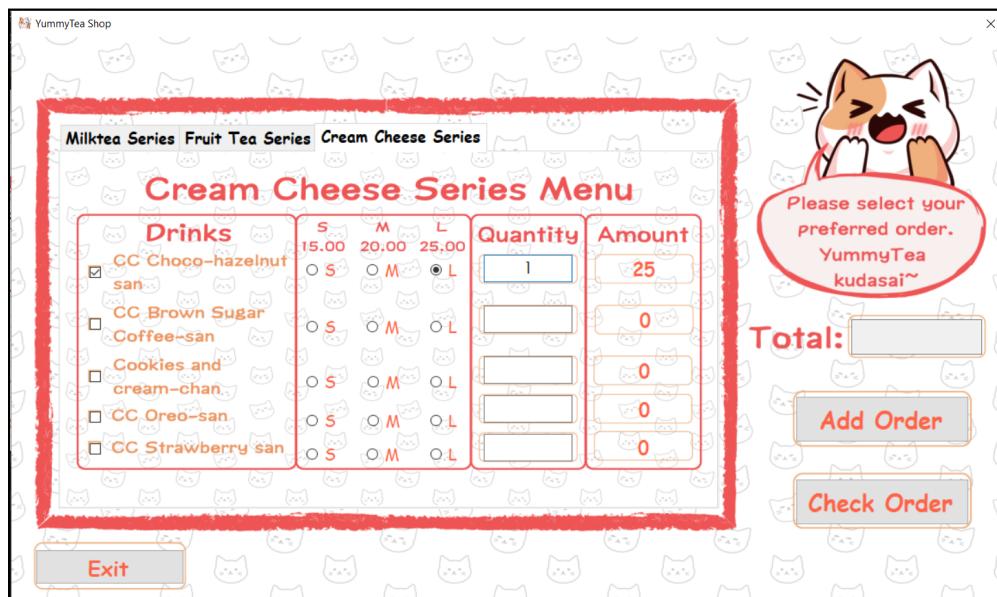


Figure 6. Ordering process of Cream Cheese Series Menu

Figure 6 illustrates the ordering process of the Cream Cheese Series Menu and its functioning.

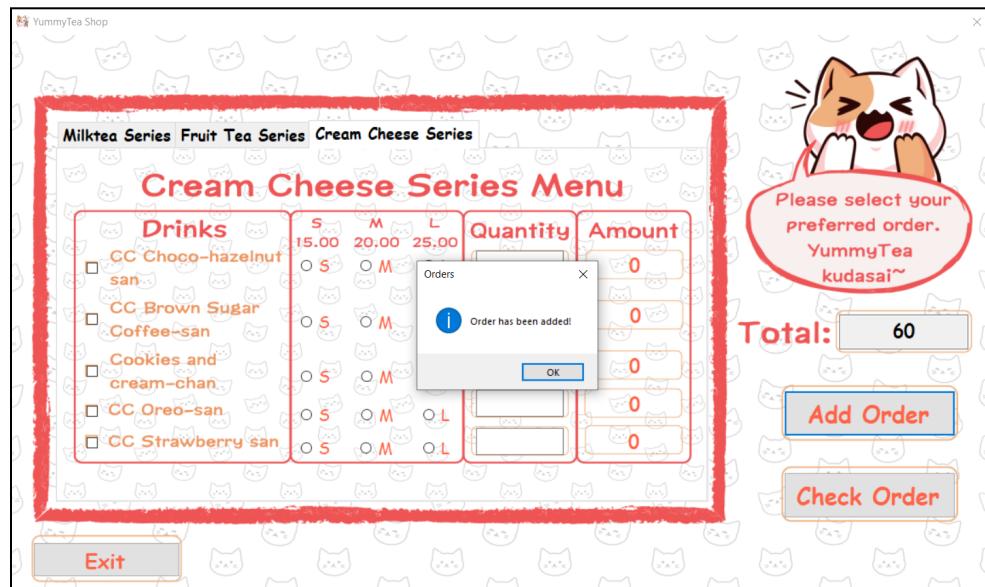


Figure 7. Add Button Functionality

Figure 7 illustrates the functionality of the Add Button, which displays a message box stating, "Order has been added!" and subsequently saves the user's order.



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

The screenshot shows a mobile application interface titled "YummyTea Shop". At the top, there is a header with the shop name and a close button (X). Below the header, the title "List of Orders" is displayed in red. A table lists the ordered items, their sizes, quantities, and amounts:

Drinks	Size	Quantity	Amount
Okinawa-chan	S	1	15
Lychee-kun	M	1	20
CC Choco-hazelnut san	L	1	25

At the bottom left, the text "Total:" is followed by a box containing the number "60". To the right of the total, there is a speech bubble with the text "ください Please verify your orders~" and a cartoon cat icon. Below the total, there are two buttons: "Print Ticket" and "Cancel Order".

Figure 8. Check Order Button Functionality

Figure 8 illustrates the functionality of the "Check Order" button, which directs the user to a list of their orders. This feature enables users to view their ordered items and the total amount they are required to pay. Additionally, there are interactive buttons available for user engagement, including:

- 1. PRINT TICKET** - This option displays the ticket that the user needs to present at the counter.
- 2. CANCEL ORDER** - This button allows users to cancel their transaction.



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

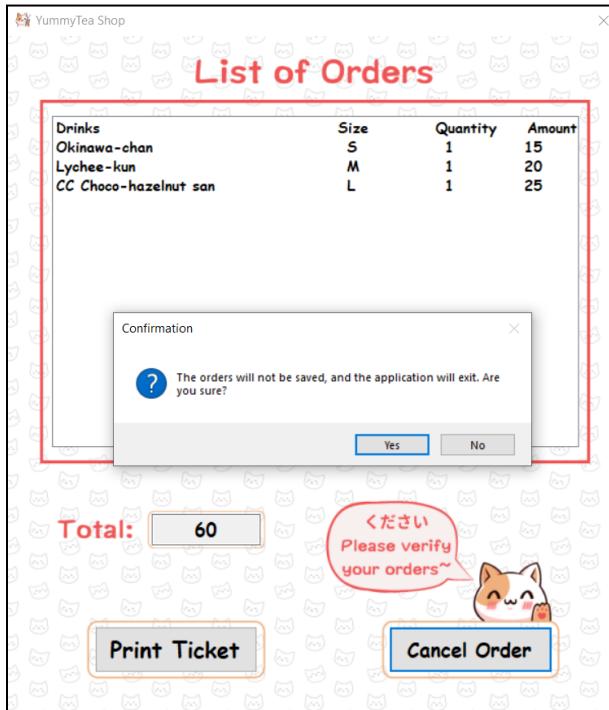


Figure 9: Cancel Order Button Functionality

Figure 9 illustrates the functionality of the Cancel Order button, allowing users to choose whether or not to cancel the transaction.



Figure 10. Print Ticket Button Functionality

Figure 10 demonstrates the functionality of the Print Ticket Button, wherein the system displays the user's ticket, allowing them to both print and save it.



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

VII. SOURCE CODE

Loading Screen Form

header code:

```
#include "WcScreenForm.h"
```

```
#include <Windows.h>
```

code:

```
#pragma endregion
```

```
// For loading screen
```

```
private: System::Void LoadScreenForm_Load(System::Object^ sender,  
System::EventArgs^ e) {  
  
    loadTimer->Interval = 200; // Interval in milliseconds  
  
    loadTimer->Start();  
  
}  
  
private: System::Void loadTimer_Tick(System::Object^ sender, System::EventArgs^ e) {  
  
    if (loadBar->Value < loadBar->Maximum) {  
  
        loadBar->PerformStep();  
  
    }  
  
    else {  
  
        loadTimer->Stop();  
  
        WcScreenForm^ wcForm = gcnew WcScreenForm();  
  
        wcForm->Show();  
  
        this->Hide();  
  
    }  
  
};  
  
}
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

Wc Screen Form

header code:

```
#include "MenuForm.h"  
#include "WcScreenForm.h"
```

code:

```
#pragma endregion
```

```
private: System::Void enterBT_Click(System::Object^ sender, System::EventArgs^ e) {  
    // Start Button  
    MenuForm^ menu = gcnew MenuForm();  
    menu->Show();  
    this->Hide();  
}  
};  
}
```

Menu Form

header code:

```
#include "ListofOrders.h"
```

code:

```
#pragma endregion
```

```
//Declaration  
double order, sum, total, totalong;  
private: ListofOrders^ orderlist;  
String^ stdOrders = "{0, -45}{1, -15}{2, -13}{3, -14}";
```

```
private: System::Void MenuForm_Load(System::Object^ sender, System::EventArgs^ e)  
{
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

//Header of the List box

```
orderlist = gcnew ListofOrders();
```

```
orderlist->orderListBox->Items->Add(String::Format(stdOrders, "Drinks",
"Size", "Quantity", "Amount"));
```

```
}
```

```
private: System::Void mtAdd_Click(System::Object^ sender, System::EventArgs^ e) {
```

// Computes all the amount

```
try {
```

```
totalAm->Text = total.ToString();
```

```
orderlist->totalAmt->Text = total.ToString();
```

```
int f1Amount = Int32::Parse(f1Am->Text);
```

```
int f2Amount = Int32::Parse(f2Am->Text);
```

```
int f3Amount = Int32::Parse(f3Am->Text);
```

```
int f4Amount = Int32::Parse(f4Am->Text);
```

```
int f5Amount = Int32::Parse(f5Am->Text);
```

```
int f6Amount = Int32::Parse(f6Am->Text);
```

```
int f7Amount = Int32::Parse(f7Am->Text);
```

```
int f8Amount = Int32::Parse(f8Am->Text);
```

```
int f9Amount = Int32::Parse(f9Am->Text);
```

```
int f10Amount = Int32::Parse(f10Am->Text);
```

```
int f11Amount = Int32::Parse(f11Am->Text);
```

```
int f12Amount = Int32::Parse(f12Am->Text);
```

```
int f13Amount = Int32::Parse(f13Am->Text);
```

```
int f14Amount = Int32::Parse(f14Am->Text);
```

```
int f15Amount = Int32::Parse(f15Am->Text);
```

```
int totalAmount = Int32::Parse(totalAm->Text);
```

```
total = f1Amount + f2Amount + f3Amount + f4Amount + f5Amount
```

```
+ f6Amount + f7Amount + f8Amount + f9Amount + f10Amount
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

+ f11Amount + f12Amount + f13Amount + f14Amount +

f15Amount;

```
totalong = total + totalAmount;  
totalAm->Text = totalong.ToString();  
orderlist->totalAmt->Text = totalong.ToString();  
}  
  
}
```

```
catch (FormatException^) {  
    totalAm->Text = "0";  
}
```

```
// Flavor Names  
String^ flav1 = "";  
String^ flav2 = "";  
String^ flav3 = "";  
String^ flav4 = "";  
String^ flav5 = "";  
String^ flav6 = "";  
String^ flav7 = "";  
String^ flav8 = "";  
String^ flav9 = "";  
String^ flav10 = "";  
String^ flav11 = "";  
String^ flav12 = "";  
String^ flav13 = "";  
String^ flav14 = "";  
String^ flav15 = "";  
String^ size1 = "";
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
String^ size2 = "";
String^ size3 = "";
String^ size4 = "";
String^ size5 = "";
String^ size6 = "";
String^ size7 = "";
String^ size8 = "";
String^ size9 = "";
String^ size10 = "";
String^ size11 = "";
String^ size12 = "";
String^ size13 = "";
String^ size14 = "";
String^ size15 = "";

if (mtFlavor1->Checked) { flav1 = "Okinawa-chan"; }
if (mtFlavor2->Checked) { flav2 = "Wintermelon-chan"; }
if (mtFlavor3->Checked) { flav3 = "Chocolate-chan"; }
if (mtFlavor4->Checked) { flav4 = "Cookies and cream-chan"; }
if (mtFlavor5->Checked) { flav5 = "Mocha-chan"; }

if (ft1->Checked) { flav6 = "Lychee-kun"; }
if (ft2->Checked) { flav7 = "Strawberry kun"; }
if (ft3->Checked) { flav8 = "Green Apple-kun"; }
if (ft4->Checked) { flav9 = "Yakult kun"; }
if (ft5->Checked) { flav10 = "Mango-kun"; }

if (cc1->Checked) { flav11 = "CC Choco-hazelnut san"; }
if (cc2->Checked) { flav12 = "CC Brown Sugar Coffee-san"; }
if (cc3->Checked) { flav13 = "CC Cookies and Cream chan"; }
if (cc4->Checked) { flav14 = "CC Oreo - san"; }
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
if(cc5->Checked) { flav15 = "CC Strawberry san"; }
```

```
// Flavor Size, Quantity, and Amount
```

```
//Milktea
```

```
if(f1Small->Checked) {
```

```
    size1 = "S";
```

```
}
```

```
else if(f1Medium->Checked) {
```

```
    size1 = "M";
```

```
}
```

```
else {
```

```
    size1 = "L";
```

```
}
```

```
String^ qty1 = f1quan->Text;
```

```
String^ amount1 = f1Am->Text;
```

```
if(f2Small->Checked) {
```

```
    size2 = "S";
```

```
}
```

```
else if(f2Medium->Checked) {
```

```
    size2 = "M";
```

```
}
```

```
else {
```

```
    size2 = "L";
```

```
}
```

```
String^ qty2 = f2quan->Text;
```

```
String^ amount2 = f2Am->Text;
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
if(f3Small->Checked) {  
    size3 = "S";  
}  
  
else if(f3Medium->Checked) {  
    size3 = "M";  
}  
  
else {  
    size3 = "L";  
}  
  
String^ qty3 = f3quan->Text;  
String^ amount3 = f3Am->Text;  
  
  
if(f4Small->Checked) {  
    size4 = "S";  
}  
  
else if(f4Medium->Checked) {  
    size4 = "M";  
}  
  
else {  
    size4 = "L";  
}  
  
String^ qty4 = f4quan->Text;  
String^ amount4 = f4Am->Text;  
  
  
if(f5Small->Checked) {  
    size5 = "S";  
}  
  
else if(f5Medium->Checked) {
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

size5 = "M";

}

else {

size5 = "L";

}

String^ qty5 = f5quan->Text;

String^ amount5 = f5Am->Text;

//FruitTea

if (f6Small->Checked) {

size6 = "S";

}

else if (f6Medium->Checked) {

size6 = "M";

}

else {

size6 = "L";

}

String^ qty6 = f6quan->Text;

String^ amount6 = f6Am->Text;

if (f7Small->Checked) {

size7 = "S";

}

else if (f7Medium->Checked) {

size7 = "M";

}

else {



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

size7 = "L";

}

String^ qty7 = f7quan->Text;

String^ amount7 = f7Am->Text;

if (f8Small->Checked) {

size8 = "S";

}

else if (f8Medium->Checked) {

size8 = "M";

}

else {

size8 = "L";

}

String^ qty8 = f8quan->Text;

String^ amount8 = f8Am->Text;

if (f9Small->Checked) {

size9 = "S";

}

else if (f9Medium->Checked) {

size9 = "M";

}

else {

size9 = "L";

}

String^ qty9 = f9quan->Text;

String^ amount9 = f9Am->Text;



```
if (f10Small->Checked) {  
    size10 = "S";  
}  
else if (f10Medium->Checked) {  
    size10 = "M";  
}  
else {  
    size10 = "L";  
}  
String^ qty10 = f10quan->Text;  
String^ amount10 = f10Am->Text;  
  
//CreamCheese  
if (f11Small->Checked) {  
    size11 = "S";  
}  
else if (f11Medium->Checked) {  
    size11 = "M";  
}  
else {  
    size11 = "L";  
}  
String^ qty11 = f11quan->Text;  
String^ amount11 = f11Am->Text;  
  
if (f12Small->Checked) {  
    size12 = "S";
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

}

```
else if (f12Medium->Checked) {
```

```
    size12 = "M";
```

```
}
```

```
else {
```

```
    size12 = "L";
```

```
}
```

```
String^ qty12 = f12quan->Text;
```

```
String^ amount12 = f12Am->Text;
```

```
if (f13Small->Checked) {
```

```
    size13 = "S";
```

```
}
```

```
else if (f13Medium->Checked) {
```

```
    size13 = "M";
```

```
}
```

```
else {
```

```
    size13 = "L";
```

```
}
```

```
String^ qty13 = f13quan->Text;
```

```
String^ amount13 = f13Am->Text;
```

```
if (f14Small->Checked) {
```

```
    size14 = "S";
```

```
}
```

```
else if (f14Medium->Checked) {
```

```
    size14 = "M";
```

```
}
```



```
else {  
    size14 = "L";  
}  
  
String^ qty14 = f14quan->Text;  
String^ amount14 = f14Am->Text;  
  
  
if (f15Small->Checked) {  
    size15 = "S";  
}  
else if (f15Medium->Checked) {  
    size15 = "M";  
}  
else {  
    size15 = "L";  
}  
  
String^ qty15 = f15quan->Text;  
String^ amount15 = f15Am->Text;  
  
  
// Add the order to the list box  
//Milktea  
  
if (mtFlavor1->Checked) {  
    mtFlavor1->Checked = false; f1Small->Checked = false;  
    f1Medium->Checked = false; f1Large->Checked = false; f1quan->Clear();  
  
    String^ orderEntry1 = flav1 + "\t\t\t" + size1 + "\t\t" + qty1 + "\t\t\t" +  
    amount1;  
  
    orderlist->orderListBox->Items->Add(orderEntry1);  
}  
  
if (mtFlavor2->Checked) {
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
mtFlavor2->Checked = false; f2Small->Checked = false;  
f2Medium->Checked = false; f2Large->Checked = false; f2quan->Clear();  
  
String^ orderEntry2 = flav2 + "\t\t\t" + size2 + "\t\t" + qty2 + "\t\t" +  
amount2;  
  
orderlist->orderListBox->Items->Add(orderEntry2);  
  
}  
  
if (mtFlavor3->Checked) {  
  
    mtFlavor3->Checked = false; f3Small->Checked = false;  
f3Medium->Checked = false; f3Large->Checked = false; f3quan->Clear();  
  
    String^ orderEntry3 = flav3 + "\t\t\t" + size3 + "\t\t" + qty3 + "\t\t" +  
amount3;  
  
    orderlist->orderListBox->Items->Add(orderEntry3);  
  
}  
  
if (mtFlavor4->Checked) {  
  
    mtFlavor4->Checked = false; f4Small->Checked = false;  
f4Medium->Checked = false; f4Large->Checked = false; f4quan->Clear();  
  
    String^ orderEntry4 = flav4 + "\t\t" + size4 + "\t\t" + qty4 + "\t\t" +  
amount4;  
  
    orderlist->orderListBox->Items->Add(orderEntry4);  
  
}  
  
if (mtFlavor5->Checked) {  
  
    mtFlavor5->Checked = false; f5Small->Checked = false;  
f5Medium->Checked = false; f5Large->Checked = false; f5quan->Clear();  
  
    String^ orderEntry5 = flav5 + "\t\t\t" + size5 + "\t\t" + qty5 + "\t\t" +  
amount5;  
  
    orderlist->orderListBox->Items->Add(orderEntry5);  
  
}  
  
//Fruittea  
  
if (ft1->Checked) {  
  
    ft1->Checked = false; f6Small->Checked = false; f6Medium->Checked =  
false; f6Large->Checked = false; f6quan->Clear();
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
String^ orderEntry6 = flav6 + "\t\t\t" + size6 + "\t\t" + qty6 + "\t\t\t" +
amount6;

orderlist->orderListBox->Items->Add(orderEntry6);

}

if (ft2->Checked) {

    ft2->Checked = false; f7Small->Checked = false; f7Medium->Checked =
false; f7Large->Checked = false; f7quan->Clear();

    String^ orderEntry7 = flav7 + "\t\t\t" + size7 + "\t\t" + qty7 + "\t\t\t" +
amount7;

orderlist->orderListBox->Items->Add(orderEntry7);

}

if (ft3->Checked) {

    ft3->Checked = false; f8Small->Checked = false; f8Medium->Checked =
false; f8Large->Checked = false; f8quan->Clear();

    String^ orderEntry8 = flav8 + "\t\t\t" + size8 + "\t\t" + qty8 + "\t\t\t" +
amount8;

orderlist->orderListBox->Items->Add(orderEntry8);

}

if (ft4->Checked) {

    ft4->Checked = false; f9Small->Checked = false; f9Medium->Checked =
false; f9Large->Checked = false; f9quan->Clear();

    String^ orderEntry9 = flav9 + "\t\t\t" + size9 + "\t\t" + qty9 + "\t\t\t" +
amount9;

orderlist->orderListBox->Items->Add(orderEntry9);

}

if (ft5->Checked) {

    ft5->Checked = false; f10Small->Checked = false; f10Medium->Checked =
false; f10Large->Checked = false; f10quan->Clear();

    String^ orderEntry10 = flav10 + "\t\t\t" + size10 + "\t\t" + qty10 + "\t\t\t" +
amount10;

orderlist->orderListBox->Items->Add(orderEntry10);
```



}

//Cream Cheese

if (cc1->Checked) {

 cc1->Checked = false; f11Small->Checked = false; f11Medium->Checked
 = false; f11Large->Checked = false; f11quan->Clear();

 String^ orderEntry11 = flav11 + "\t\t" + size11 + "\t\t" + qty11 + "\t\t" +
 amount11;

 orderlist->orderListBox->Items->Add(orderEntry11);

}

if (cc2->Checked) {

 cc2->Checked = false; f12Small->Checked = false; f12Medium->Checked
 = false; f12Large->Checked = false; f12quan->Clear();

 String^ orderEntry12 = flav12 + "\t\t" + size12 + "\t\t" + qty12 + "\t\t"
 + amount12;

 orderlist->orderListBox->Items->Add(orderEntry12);

}

if (cc3->Checked) {

 cc3->Checked = false; f13Small->Checked = false; f13Medium->Checked
 = false; f13Large->Checked = false; f13quan->Clear();

 String^ orderEntry13 = flav13 + "\t\t" + size13 + "\t\t" + qty13 + "\t\t"
 + amount13;

 orderlist->orderListBox->Items->Add(orderEntry13);

}

if (cc4->Checked) {

 cc4->Checked = false; f14Small->Checked = false; f14Medium->Checked
 = false; f14Large->Checked = false; f14quan->Clear();

 String^ orderEntry14 = flav14 + "\t\t\t" + size14 + "\t\t" + qty14 + "\t\t"
 + amount14;

 orderlist->orderListBox->Items->Add(orderEntry14);

}

if (cc5->Checked) {



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
cc5->Checked = false; f15Small->Checked = false; f15Medium->Checked  
= false; f15Large->Checked = false; f15quan->Clear();
```

```
String^ orderEntry15 = flav15 + "\t\t\t" + size15 + "\t\t" + qty15 + "\t\t"  
+ amount15;
```

```
orderlist->orderListBox->Items->Add(orderEntry15);
```

```
}
```

```
//Message Box when order added
```

```
MessageBox::Show("Order has been added!", "Orders",  
MessageBoxButtons::OK, MessageBoxIcon::Information);
```

```
}
```

```
//Function for radio buttons and check boxes
```

```
private: System::Void f1quan_TextChanged(System::Object^ sender,  
System::EventArgs^ e) {
```

```
f1Am->Text = f1quan->Text;
```

```
try {
```

```
int quanty1 = Int32::Parse(f1quan->Text);
```

```
if (mtFlavor1->Checked) {
```

```
if (f1Small->Checked) {
```

```
order = 15;
```

```
sum = order * quanty1;
```

```
f1Am->Text = sum.ToString();
```

```
}
```

```
else if (f1Medium->Checked) {
```

```
order = 20;
```

```
sum = order * quanty1;
```



```
    f1Am->Text = sum.ToString();  
}  
  
else  
  
    order = 25;  
  
    sum = order * quan1;  
  
    f1Am->Text = sum.ToString();  
}  
  
}  
  
catch (FormatException^) {  
  
    f1Am->Text = "0";  
}  
  
}  
  
private: System::Void f2quan_TextChanged(System::Object^ sender,  
System::EventArgs^ e) {  
  
    f2Am->Text = f2quan->Text;  
  
try {  
  
    int quan2 = Int32::Parse(f2quan->Text);  
  
    if (mtFlavor2->Checked) {  
  
        if (f2Small->Checked) {  
  
            order = 15;  
  
            sum = order * quan2;  
  
            f2Am->Text = sum.ToString();  
        }  
  
        else if (f2Medium->Checked) {  
  
            order = 20;  
        }  
    }  
}
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
sum = order * quantity2;

f2Am->Text = sum.ToString();

}

else

    order = 25;

    sum = order * quantity2;

    f2Am->Text = sum.ToString();

}

}

catch (FormatException^) {

    f2Am->Text = "0";

}

private: System::Void f3quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

    f3Am->Text = f3quan->Text;

    try {

        int quantity3 = Int32::Parse(f3quan->Text);

        if (mtFlavor3->Checked) {

            if (f3Small->Checked) {

                order = 15;

                sum = order * quantity3;

                f3Am->Text = sum.ToString();

            }

            else if (f3Medium->Checked) {
```



```
order = 20;

sum = order * quan3;

f3Am->Text = sum.ToString();

}

else {

    order = 25;

    sum = order * quan3;

    f3Am->Text = sum.ToString();

}

}

}

catch (FormatException^) {

    f3Am->Text = "0";

}

private: System::Void f4quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

    f4Am->Text = f4quan->Text;

}

try {

    int quan4 = Int32::Parse(f4quan->Text);

    if (mtFlavor4->Checked) {

        if (f4Small->Checked) {

            order = 15;

            sum = order * quan4;

            f4Am->Text = sum.ToString();

        }

    }

}
```



}

```
else if (f4Medium->Checked) {  
  
    order = 20;  
  
    sum = order * quan4;  
  
    f4Am->Text = sum.ToString();  
  
}  
  
else  
  
    order = 25;  
  
    sum = order * quan4;  
  
    f4Am->Text = sum.ToString();  
  
}
```

```
}  
  
catch (FormatException^) {  
  
    f4Am->Text = "0";  
  
}
```

```
private: System::Void f5quan_TextChanged(System::Object^ sender,  
System::EventArgs^ e) {
```

```
    f5Am->Text = f5quan->Text;
```

```
try {  
  
    int quan5 = Int32::Parse(f5quan->Text);
```

```
if (mtFlavor5->Checked) {  
  
    if (f5Small->Checked) {  
  
        order = 15;  
  
        sum = order * quan5;
```



```
f5Am->Text = sum.ToString();  
  
}  
  
else if (f5Medium->Checked) {  
  
    order = 20;  
  
    sum = order * quan5;  
  
    f5Am->Text = sum.ToString();  
  
}  
  
else  
  
    order = 25;  
  
    sum = order * quan5;  
  
    f5Am->Text = sum.ToString();  
  
}  
  
}  
  
catch (FormatException^) {  
  
    f5Am->Text = "0";  
  
}  
  
}  
  
private: System::Void f6quan_TextChanged(System::Object^ sender,  
System::EventArgs^ e) {  
  
    f6Am->Text = f6quan->Text;  
  
    try {  
  
        int quan6 = Int32::Parse(f6quan->Text);  
  
        if (ft1->Checked) {  
  
            if (f6Small->Checked) {  
  
                order = 15;  
  
                sum = order * quan6;
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
f6Am->Text = sum.ToString();  
}  
  
else if (f6Medium->Checked) {  
  
    order = 20;  
  
    sum = order * quanty6;  
  
    f6Am->Text = sum.ToString();  
}  
  
else  
  
    order = 25;  
  
    sum = order * quanty6;  
  
    f6Am->Text = sum.ToString();  
}  
  
}  
  
catch (FormatException^) {  
  
    f6Am->Text = "0";  
}  
  
}  
  
private: System::Void f7quan_TextChanged(System::Object^ sender,  
System::EventArgs^ e) {  
  
    f7Am->Text = f7quan->Text;  
  
    try {  
  
        int quanty7 = Int32::Parse(f7quan->Text);  
  
        if (ft2->Checked) {  
  
            if (f7Small->Checked) {  
  
                order = 15;  
  
                sum = order * quanty7;  
  
                f7Am->Text = sum.ToString();  
            }  
        }  
    }  
}
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

}

```
else if (f7Medium->Checked) {  
    order = 20;  
    sum = order * quan7;  
    f7Am->Text = sum.ToString();  
}  
else  
    order = 25;  
    sum = order * quan7;  
    f7Am->Text = sum.ToString();  
}  
}  
catch (FormatException^) {  
    f7Am->Text = "0";  
}  
}  
  
private: System::Void f8quan_TextChanged(System::Object^ sender,  
System::EventArgs^ e) {  
    f8Am->Text = f8quan->Text;  
    try {  
        int quan8 = Int32::Parse(f8quan->Text);  
  
        if (ft3->Checked) {  
            if (f8Small->Checked) {  
                order = 15;  
                sum = order * quan8;  
                f8Am->Text = sum.ToString();  
            }  
        }  
    }
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
else if (f8Medium->Checked) {  
    order = 20;  
    sum = order * quanty8;  
    f8Am->Text = sum.ToString();  
}  
else  
    order = 25;  
    sum = order * quanty8;  
    f8Am->Text = sum.ToString();  
}  
}  
catch (FormatException^) {  
    f8Am->Text = "0";  
}  
}  
  
private: System::Void f9quan_TextChanged(System::Object^ sender,  
System::EventArgs^ e) {  
    f9Am->Text = f9quan->Text;  
    try {  
        int quanty9 = Int32::Parse(f9quan->Text);  
  
        if (ft4->Checked) {  
            if (f9Small->Checked) {  
                order = 15;  
                sum = order * quanty9;  
                f9Am->Text = sum.ToString();  
            }  
            else if (f9Medium->Checked) {  
                if (f9Large->Checked) {  
                    order = 25;  
                    sum = order * quanty9;  
                    f9Am->Text = sum.ToString();  
                }  
            }  
        }  
    }  
}
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
order = 20;

sum = order * quanty9;

f9Am->Text = sum.ToString();

}

else

order = 25;

sum = order * quanty9;

f9Am->Text = sum.ToString();

}

}

catch (FormatException^) {

f9Am->Text = "0";

}

}

private: System::Void f10quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

f10Am->Text = f10quan->Text;

try {

int quanty10 = Int32::Parse(f10quan->Text);

if (ft5->Checked) {

if (f10Small->Checked) {

order = 15;

sum = order * quanty10;

f5Am->Text = sum.ToString();

}

else if (f10Medium->Checked) {
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
order = 20;

sum = order * quanty10;

f10Am->Text = sum.ToString();

}

else

order = 25;

sum = order * quanty10;

f10Am->Text = sum.ToString();

}

}

catch (FormatException^) {

f10Am->Text = "0";

}

}

private: System::Void f11quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

f11Am->Text = f11quan->Text;

try {

int quanty11 = Int32::Parse(f11quan->Text);

if (cc1->Checked) {

if (f11Small->Checked) {

order = 15;

sum = order * quanty11;

f11Am->Text = sum.ToString();

}

else if (f11Medium->Checked) {
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
order = 20;

sum = order * quanty11;

f11Am->Text = sum.ToString();

}

else

order = 25;

sum = order * quanty11;

f11Am->Text = sum.ToString();

}

}

catch (FormatException^) {

f11Am->Text = "0";

}

}

private: System::Void f12quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

f12Am->Text = f12quan->Text;

try {

int quanty12 = Int32::Parse(f12quan->Text);

if (cc2->Checked) {

if (f12Small->Checked) {

order = 15;

sum = order * quanty12;

f12Am->Text = sum.ToString();

}

else if (f12Medium->Checked) {
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
order = 20;

sum = order * quanty12;

f12Am->Text = sum.ToString();

}

else

order = 25;

sum = order * quanty12;

f12Am->Text = sum.ToString();

}

}

catch (FormatException^) {

f12Am->Text = "0";

}

}

private: System::Void f13quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

f13Am->Text = f13quan->Text;

try {

int quanty13 = Int32::Parse(f13quan->Text);

if (cc3->Checked) {

if (f13Small->Checked) {

order = 15;

sum = order * quanty13;

f13Am->Text = sum.ToString();

}

else if (f13Medium->Checked) {

order = 20;
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
sum = order * quanty13;

f13Am->Text = sum.ToString();

}

else {

    order = 25;

    sum = order * quanty13;

    f13Am->Text = sum.ToString();

}

}

}

catch (FormatException^) {

    f13Am->Text = "0";

}

private: System::Void f14quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

    f14Am->Text = f14quan->Text;

    try {

        int quanty14 = Int32::Parse(f14quan->Text);

        if (cc4->Checked) {

            if (f14Small->Checked) {

                order = 15;

                sum = order * quanty14;

                f14Am->Text = sum.ToString();

            }

            else if (f14Medium->Checked) {
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
order = 20;

sum = order * quanty14;

f14Am->Text = sum.ToString();

}

else

order = 25;

sum = order * quanty14;

f14Am->Text = sum.ToString();

}

}

catch (FormatException^) {

f14Am->Text = "0";

}

}

private: System::Void f15quan_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

f15Am->Text = f15quan->Text;

try {

int quanty15 = Int32::Parse(f15quan->Text);

if (cc5->Checked) {

if (f15Small->Checked) {

order = 15;

sum = order * quanty15;

f15Am->Text = sum.ToString();

}

else if (f15Medium->Checked) {
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
order = 20;

sum = order * quanty15;

f15Am->Text = sum.ToString();

}

else

order = 25;

sum = order * quanty15;

f15Am->Text = sum.ToString();

}

}

catch (FormatException^) {

f15Am->Text = "0";

}

}

// Computation for Total Amount

private: System::Void totalAm_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

int f1Amount = Int32::Parse(f1Am->Text);

int f2Amount = Int32::Parse(f2Am->Text);

int f3Amount = Int32::Parse(f3Am->Text);

int f4Amount = Int32::Parse(f4Am->Text);

int f5Amount = Int32::Parse(f5Am->Text);

int f6Amount = Int32::Parse(f6Am->Text);

int f7Amount = Int32::Parse(f7Am->Text);

int f8Amount = Int32::Parse(f8Am->Text);

int f9Amount = Int32::Parse(f9Am->Text);

int f10Amount = Int32::Parse(f10Am->Text);
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
int f11Amount = Int32::Parse(f11Am->Text);
int f12Amount = Int32::Parse(f12Am->Text);
int f13Amount = Int32::Parse(f13Am->Text);
int f14Amount = Int32::Parse(f14Am->Text);
int f15Amount = Int32::Parse(f15Am->Text);
int totalAmount = Int32::Parse(totalAm->Text);
total = f1Amount + f2Amount + f3Amount + f4Amount + f5Amount
+ f6Amount + f7Amount + f8Amount + f9Amount + f10Amount
+ f11Amount + f12Amount + f13Amount + f14Amount + f15Amount;
totalAm->Text = totallong.ToString();
orderlist->totalAmt->Text = totallong.ToString();
}

private: System::Void mtCheck_Click(System::Object^ sender, System::EventArgs^ e) {
    orderlist->Show();
}

//Exit Button
private: System::Void mtBackBT_Click(System::Object^ sender, System::EventArgs^ e)
{
    if (MessageBox::Show("Are you sure you want to exit?", "Exit",
    MessageBoxButtons::YesNo, MessageBoxIcon::Question) ==
    System::Windows::Forms::DialogResult::Yes)
    {
        Application::Exit();
    }
}

};
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

List of Order Form

header code:

```
#include "ListofOrders.h"  
  
#include <ctime>  
  
#include <string>  
  
#include <Windows.h>  
  
#include <vcclr.h>
```

```
using namespace System::Text;  
  
using namespace System;  
  
using namespace System::Drawing;  
  
using namespace System::Drawing::Printing;
```

code:

```
#pragma endregion  
  
    // Defining printDocument  
  
    private:  
  
        PrintDocument^ printDocument;  
  
    //Cancel Button/Closes the Application  
  
    private: System::Void cnclordBt_Click(System::Object^ sender, System::EventArgs^ e) {  
  
        System::Windows::Forms::DialogResult result = MessageBox::Show("The orders  
will not be saved, and the application will exit. Are you sure?", "Confirmation",  
MessageBoxButtons::YesNo, MessageBoxIcon::Question);  
  
        if (result == System::Windows::Forms::DialogResult::Yes) {  
  
            orderListBox->Items->Clear();  
  
            Application::Exit();  
        }  
    }
```



}

// For generating random numbers

```
String^ GenerateRandomRefNo() {  
    Random^ rnd = gcnew Random();  
    int randomNumber = rnd->Next(10000, 99999);  
    return randomNumber.ToString();  
}
```

// Shows the Ticket

```
private: System::Void shwrcptBt_Click(System::Object^ sender, System::EventArgs^ e) {  
    PrintDocument^ printDocument = gcnew PrintDocument();  
    printDocument->PrintPage += gcnew PrintPageEventHandler(this,  
&ListofOrders::printDocument_PrintPage);  
    PrintPreviewDialog^ previewDialog = gcnew PrintPreviewDialog();  
    previewDialog->Document = printDocument;  
    previewDialog->>ShowDialog();  
}
```

// Design for Ticket

```
private: System::Void printDocument_PrintPage(System::Object^ sender,  
PrintPageEventArgs^ e) {  
    String^ currentDate = DateTime::Now.ToShortDateString();  
    String^ currentTime = DateTime::Now.ToShortTimeString();  
    String^ randomRefNo = GenerateRandomRefNo();  
    String^ totalAmount = totalAmt->Text;
```

// Set font styles and sizes

```
System::Drawing::Font^ ticketTitleFont = gcnew System::Drawing::Font("Arial",  
24, FontStyle::Bold);
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
System::Drawing::Font^ refNoFont = gcnew System::Drawing::Font("Arial", 14,  
FontStyle::Bold);
```

```
System::Drawing::Font^ amountFont = gcnew System::Drawing::Font("Arial",  
16);
```

```
System::Drawing::Font^ disclaimerFont = gcnew System::Drawing::Font("Arial",  
12, FontStyle::Italic);
```

```
// Set the paper size
```

```
e->PageSettings->PaperSize = gcnew PaperSize("Custom", 680, 340); // 8.5 x  
4.25 inches
```

```
// Set the center alignment
```

```
StringFormat^ centerAlignment = gcnew StringFormat();  
centerAlignment->Alignment = StringAlignment::Center;
```

```
// Set the line separators
```

```
String^ separator =  
"=====";
```

```
// Calculate the x-coordinate for the separators
```

```
int separatorX = (e->MarginBounds.Width -  
e->Graphics->MeasureString(separator, refNoFont).Width) / 2;
```

```
// Draw the ticket layout on the document
```

```
e->Graphics->DrawString("TICKET", ticketTitleFont, Brushes::Black,  
e->MarginBounds.Width / 2, 10, centerAlignment);
```

```
e->Graphics->DrawString("[COUNTER]", refNoFont, Brushes::Black,  
e->MarginBounds.Width / 2, 80, centerAlignment);
```

```
e->Graphics->DrawString(separator, refNoFont, Brushes::Black, separatorX,  
100);
```



COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

```
e->Graphics->DrawString(currentDate + ", " + currentTime, refNoFont,  
Brushes::Black, e->MarginBounds.Width / 2, 120, centerAlignment);  
  
e->Graphics->DrawString(separator, refNoFont, Brushes::Black, separatorX,  
140);  
  
e->Graphics->DrawString("REF NO.: #" + randomRefNo, refNoFont,  
Brushes::Black, e->MarginBounds.Width / 2, 160, centerAlignment);  
  
e->Graphics->DrawString("", refNoFont, Brushes::Black,  
e->MarginBounds.Width / 2, 200, centerAlignment);  
  
e->Graphics->DrawString("Total Amount: P" + totalAmount, amountFont,  
Brushes::Black, e->MarginBounds.Width / 2, 220, centerAlignment);  
  
e->Graphics->DrawString(separator, refNoFont, Brushes::Black, separatorX,  
260);  
  
e->Graphics->DrawString("*THIS TICKET IS FOR ORDER CLAIMS ONLY*",  
disclaimerFont, Brushes::Black, e->MarginBounds.Width / 2, 280, centerAlignment);  
  
e->Graphics->DrawString("Please proceed to counter", disclaimerFont,  
Brushes::Black, e->MarginBounds.Width / 2, 300, centerAlignment);  
  
e->Graphics->DrawString("YummyTea Shop", disclaimerFont, Brushes::Black,  
e->MarginBounds.Width / 2, 320, centerAlignment);  
  
}  
};  
}
```



Republic of the Philippines

RIZAL TECHNOLOGICAL UNIVERSITY

Cities of Mandaluyong and Pasig

COLLEGE OF ENGINEERING, ARCHITECTURE AND TECHNOLOGY

VII. CONCLUSION

In conclusion, the researcher's knowledge of C++ and GUI has helped them in creating this application, and the program also aims to assist those who struggle with placing orders at the counter. The findings of this research have significant implications for practical applications in the food and beverage industry. Our research provides insights into factors that influence purchasing decisions and consumer preferences. Implementing these findings could lead to more targeted marketing campaigns and enhanced customer satisfaction. This type of service or program has proven to be extremely useful as it reduces waiting time for customers, allowing them to avoid queues and simply wait for their order to be served, as proven and tested by fast food chains like McDonald's, where the process of ordering food has been expedited.