[GROCERY MANAGEMENT SYSTEM]

A Project submitted in partial fulfillment of the requirements for
The Award of the degree of

BACHELOR OF COMPUTER SCIENCE

SUBMITTED BY

Gokul.K

[111948013],

[Iswarya.K]

[111948019]

GUIDED BY

[Mr.R.Saravanan]

[Assistant Professor **]**

DEPARTMENT OF COMPUTER SCIENCE AND APPLICATION

SRI CHANDRASEKHARENDRA SARASWATHI VISWA MAHAVIDYALAYA

(Deemed to be University Under Section 3 of UGC Act 1956)
Accredited with "A" Grade by NAAC
Enathur, Kanchipuram – 631 561.

[APRIL 2022]

SRI CHANDRASEKHARENDRA SARASWATHI VISWA MAHAVIDYALAYA

(Deemed to be University Under Section 3 of UGC Act 1956)
Accredited with "A" Grade by NAAC
Enathur, Kanchipuram – 631 561.

BONAFIDE CERTIFICATE

This is to certify that the project entitled **[GROCERY MANAGEMENT SYSTEM]** is the bonafide work carried out by Gokul.K, [111948013] and Iswarya.K, [111948019] during the academic year 2021-2022.

[Mr.R.Saravanan]
[Assistant professor]
Department of Computer Science & Applications
SCSVMV

Dr. T. NIRMAL RAJ
Head of the Department
Department of Computer Science & Applications
SCSVMV.

Submitted for the project work viva-voce examination held on _____

Internal Examiner **External Examiner**

SRI CHANDRASEKHARENDRA SARASWATHI VISWA MAHAVIDYALAYA

(Deemed to be University Under Section 3 of UGC Act 1956)
Accredited with "A" Grade by NAAC
Enathur, Kanchipuram – 631 561.

CERTIFICATE BY THE GUIDE

This is to certify that the project entitled GROCERY MANAGEMENT SYSTEM Submitted for the Degree of B.Sc.(CS) by Gokul.K, [111948013] and Iswarya.K, [111948019]is a record of project work carried out by him/her during the period from December 2021 to April 2022 under my guidance.

Signature of the Guide

Date:

CONTENT

iii. **Alpha testing**
iv. **Beta testing**
v. **Unit testing**
vi. **Integration testing**
vii. **Validation testing**
viii. **System testing**

5. REPORTS:

6. CONCLUSION:

7. SCOPE FOR FURTHER DEVELOPMENT:

8. BIBILOGRAPHY AND REFERENCES:

9. APPENDIX

CHAPTER-1

INTRODUCTION

1.1 Introduction:

- The "grocery management system" has been developed to override the problems prevailing in the practicing manual system.
- This software is supported to eliminate and in some cases reduce the hardships faced by this existing system.
- Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.
- This application is reduced as much as possible to avoid errors while entering the data.
- No formal knowledge is needed for the user to use this system.
- Thus by this all it proves it is user-friendly.
- Grocery management system, as described above, can lead to error free, secure, reliable, and fast management system.

1.2 Abstract:

The purpose of grocery management system is to automate the existing manual system by the help of computerized equipments and full – fledged computer software, fulfilling their requirements, so that their valuable data / information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Grocery management system, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

The aim is to automate its exiting manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data/ information can be stored for a long period with easy accessing and manipulation of the same. Basically the project describes how to manage for good performance and better services for the clients.

1.3 Objective:

The main objective of the project grocery management system is to manage the details of the customer, product, product company, product type. It manages all the information about customer, stock, supplier, customer. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the customer, product, stock, product company. It tracks all the details about the product company, product type, supplier.

1.4 About the organization:

In everyday life, supermarkets are important for most people. What exactly is a supermarket? It is a business enterprise providing services of various types to its customer. Interesting, a supermarket does not produce any physical products in general, on the contrary it acquires products from places that are far away, store them in the storehouse, distribute them to local shops and sell them to their local customer, for whom it could have been difficult to acquire the product from a remote supplier.

1.5 Existing system:

- A grocery store is a retail store that primarily sells food. A grocer is a bulk seller of food.
- Grocery stores often non – perishable food that is packed in cans, bottles and boxes, with some also having fresh produce, butchers, delis and bakeries.
- As pollution around the world has increased buying food on the road side is not advisable. whether in charge of a small individually owned grocery store or one that is part of larger chain maintain a grocery store successfully involves considerable responsibility.
- Grocery store managers must ensure that the store runs smoothly that items are priced comparatively and that customers are satisfied.

1.6 Drawbacks of existing system:

- Manual system faces a lot of inefficiencies.
- Hinders smooth flow of work.
- Lack of security of data.
- Although more man power but less efficiency.
- Time consuming.
- Consumes large volume of paper work.
- Needs manual calculations which are prone to errors.
- Poor communication may lead to serious inadvertent error.

1.7 Proposed system:

Understanding of key concepts involved in effective grocery store management is imperative for any manager dedicated to the success of his store.

Particularly if the manages the small grocery store inventory requires more than simply keeping enough of every item in stock.

This is an internet-based application that is accessed through web. This system can be used to search for all grocery and fresh vegetables which are available in the market.

1.8 Advantages of proposed system:

- ✓ Accuracy in work.
- ✓ Error reduction.
- ✓ Easy to update information.
- ✓ Improved report generation and analysis.
- ✓ Better equipped to meet user requirements.
- ✓ Reduction in use of paper.
- ✓ Reduction in man power.
- ✓ Faster response time.
- ✓ Work becomes very speedy.
- ✓ User accounts to control the access and maintain security.
- ✓ Robust database back-end.
- ✓ It contain better storage capacity.
- ✓ Decrease the load of the person involve in existing manual system.
- ✓ Be easy to understand by the user and the operator.
- ✓ Be easy to operate.
- ✓ Have a good user interface.
- ✓ It satisfy the user requirement.

CHAPTER – 2

SYSTEM REQUIREMENTS

CHAPTER – 2

SYSTEM REQUIREMENTS

2.1 **Software Environment:**

Operating System : Windows 10 & 11

Front End : Visual studio 2010 , Asp.net

Back End : SQL server

2.2 Hardware Environment:

System : Windows 11

Speed : 2.5 GHz

Hard Disk : 20 GB

RAM : 2 GB

CHAPTER – 3

SYSTEM PLANNING

AND

DESIGNING

CHAPTER – 3

SYSTEM PLANNING AND DESIGNING

3.1 OVERALL SYSTEM DESIGN ARCHITECTURE:

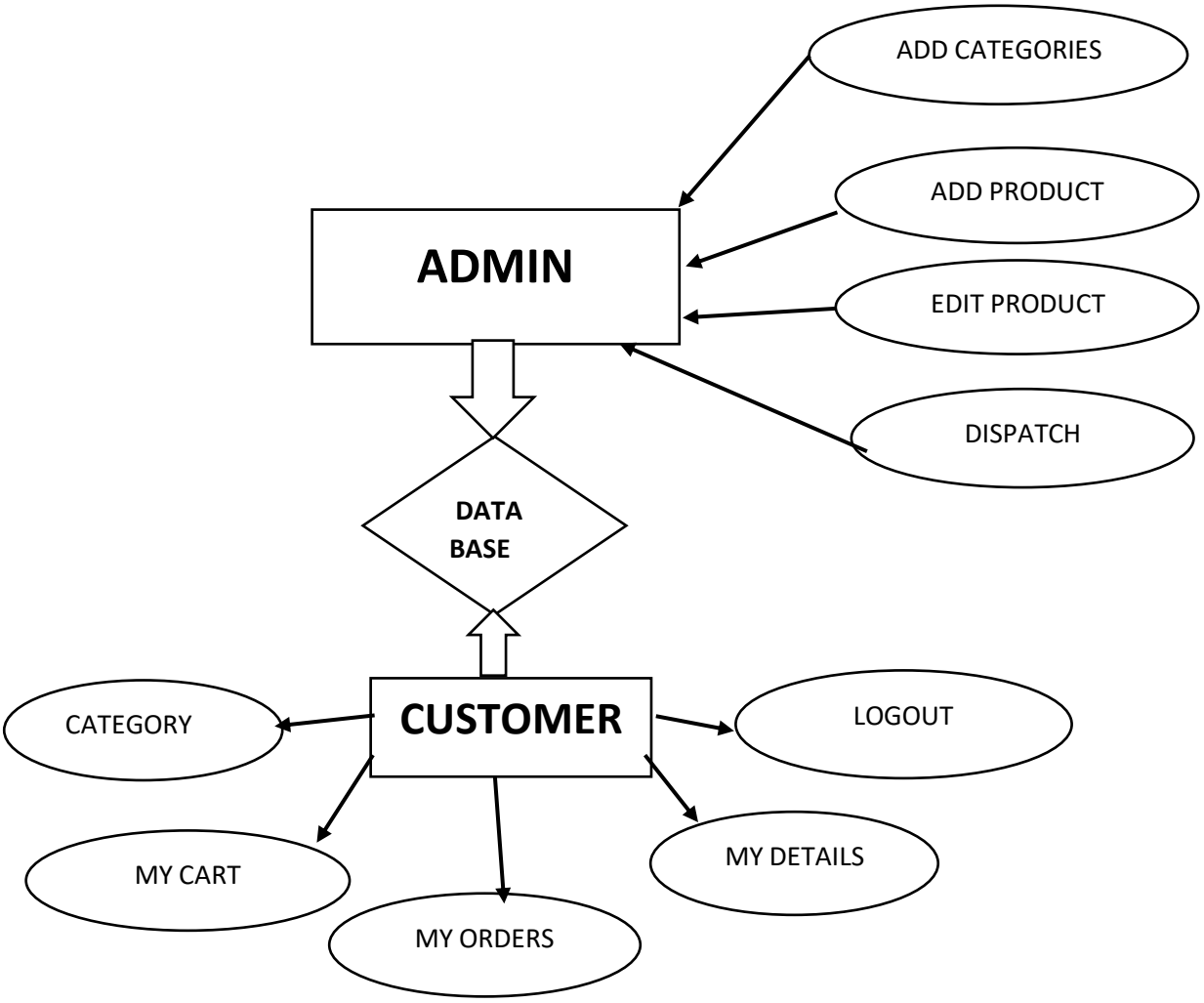

FIG 3.1.1 OVERALL SYSTEM DESIGN ARCHITECTURE

3.2 MODULES DESCRIPTION:

Modules:

These are the Modules of our Project

- Login module: It allows the user to type the username and the password to log in.
- Register module: Used to register.
- User module: Allows user to register, login in and log out.
- Admin module: It allows project admins to manage the products, categories, etc.,

3.3 DATA FLOW DIAGRAM:

Data flow diagram is the starting point of the design phase that functionally decomposes the requirements specification. A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flows in the system. A DFD describes what data flow rather than how they are processed, so it does not hardware,software,data structure.

A data flow diagram(DFD) is a graphical representation of the flow of data through an informatin system .DFDs can also be used for the visualization of data processing.

A data flow diagram (DFD) is a significant modeling technique for analyzing and constructing information processes. DFD literally means an illustration that explains the course or movement of information in a process.

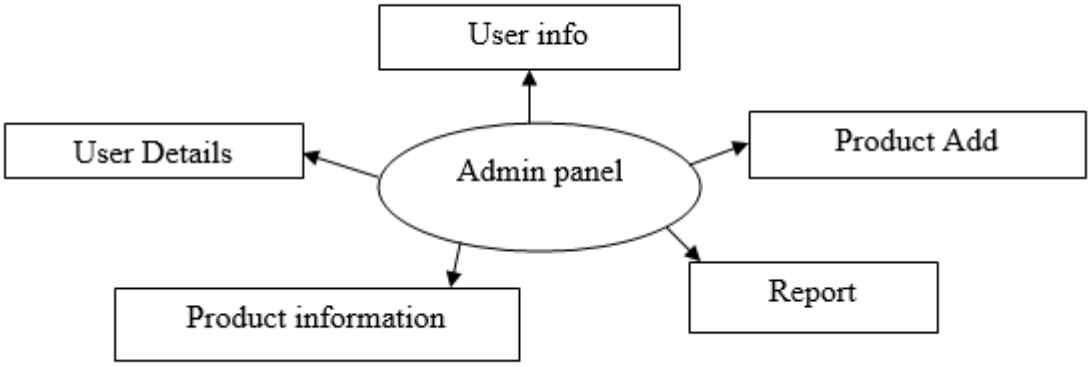

FIG 3.3.1 ADMIN PANEL FORM DFD

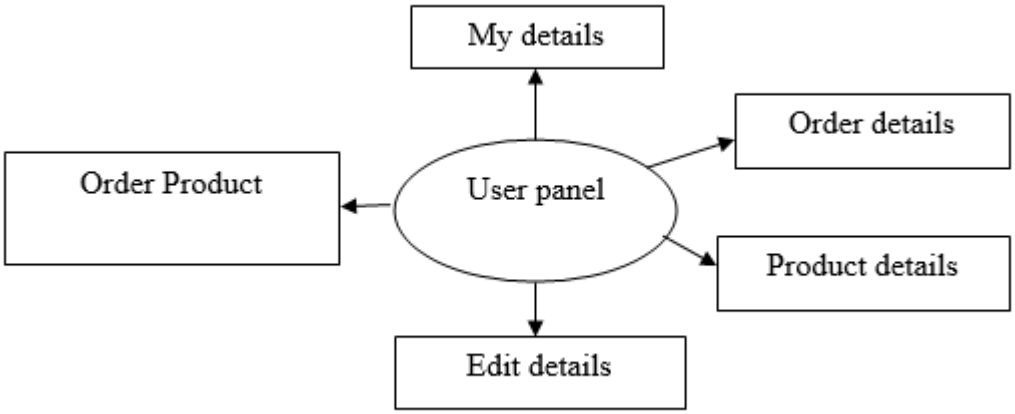

FIG 3.3.2 USER PANEL FORM DFD

3.4 DATABASE DESIGN:

Database is a file composed of records, each containing fields together with a set of operations it helps in organizing data in a logical order for references.

Database contains related data which are organized together in a group of object, table, and file. It can be in form of node. In this project a relational database concept will be used in this appraisal, related data will be store or organize in different table.

The Database design of this system is showed in diagram 3.5.1 – 3.5.6

Column Name	Data Type	Description
Id	nvarchar(50)	It is a unique user id
Password	nvarchar(50)	Used to store password
Name	nvarchar(50)	Used to store the name
Mobileno	nvarchar(50)	Used to store Mobile Number
Address	nvarchar(MAX)	Used to store the Address
Email	nvarchar(50)	Used to store the Mail ID

3.4.1 USER TABLE

Column Name	Data Type	Description
Category	nvarchar(50)	Used to Store the Category of Food
Photo	nvarchar(50)	Used to store the path of photo

3.4.2 CATEGORY TABLE

Column Name	Data Type	Description
Category	nvarchar(50)	Used to Store the Category of Food
foodname	nvarchar(50)	Used to Store the Name of the Food
Amount	Int	Used to store the amount of Food
Photo	nvarchar(50)	Used to store the path of photo

3.4.3 PRODUCT TABLE

Column Name	Data Type	Description
Userid	nvarchar(50)	It is a unique user id
Name	nvarchar(50)	Used to store Name

3.4.4 TEMP TABLE

Column Name	Data Type	Description
Userid	nvarchar(50)	It is a unique user id
Name	nvarchar(50)	Used to store Name
Category	nvarchar(50)	Used to store the category of product
Foodname	nvarchar(50)	Used to store the food name
Amount	nvarchar(50)	Used to store the amount of product
Size	nvarchar(50)	Used to store the size of the product
Cardno	nvarchar(50)	Used to store the Card Number
Address	nvarchar(MAX)	Used to store the Address
Date	nvarchar(50)	Used to store the date of purchase
Time	nvarchar(50)	Used to store the time of Purchase.
Status	nvarchar(50)	Used to store the status of product.

3.4.5 ORDER TABLE

Column Name	Data Type	Description
Userid	nvarchar(50)	It is a unique user id
Name	nvarchar(50)	Used to store Name
Category	nvarchar(50)	Used to store the category of product
Foodname	nvarchar(50)	Used to store the food name
Amount	nvarchar(50)	Used to store the amount of product
Quantity	Int	Used to store the quantity of food

3.4.6 CART TABLE

CHAPTER – 4

SYSTEM IMPLEMENTATION

CHAPTER – 4

SYSTEM IMPLEMENTATION

4.1 SECURITY TESTING FOR THE PROJECT:

Testing is vital role for the success of any software. No system design is ever perfect. Testing is also carried in two phases. First phase is during the software engineering that is during the module creation. Second phase is after the completion of software. This is system testing which verifies the whole set of programs hanged together.

i. WHITE BOX TESTING:

In this technique, the close examination of the logical parts through the software are tested by cases that exercise species sets of conditions or loops, all logical parts of the software checked once. Errors that can be corrected using this technique are typographical errors, logical expressions which should be executed once may be getting executed more than once and error resulting by using wrong controls and loops. When the box testing tests all the independent part within a module a logical-decisions on their true and the false side are exercised, all loops and bounds within their operational bounds were exercised and internal data structure to ensure their validity were exercised once.

ii. BLACK BOX TESTING:

This method enables the software engineer to device sets of input techniques that fully exercise all functional requirements of a program. Black box testing tests the input. The output and the external data. It checks whether the input data is correct and whether we are getting the desired output.

iii. ALPHA TESTING:

Acceptance testing is also sometimes called alpha testing. Be spoke systems are developed for a single customer. The alpha testing proceeds until the system developer and the customer agree that the provided system is an acceptable implementation of the system requirements.

iv. BETA TESTING:

On the other hand, when a system is to be marked as a software product, another process called beta testing is often conducted. During beta testing, a system is delivered among a number of potential users who agree to use it. The customers then report to the developers. This provides the product for real use and detects errors which may not have been anticipated by the system developers.

v. UNIT TESTING:

Each module is considered independently, it focuses on each unit of software as implemented in the source code, it is white box testing.

vi. INTEGRATION TESTING:

Integrating testing aims at constructing the program structure while at the same constructing tests to uncover errors associated with interfacing the modules. Modules are integrated by using the top-down approach.

vii. VALIDATION TESTING:

Validation testing was performed to ensure that all the functional and performance requirements are met.

viii. SYSTEM TESTING:

It is executing programs to check logical changes made in it with intention of finding errors. A system is tested for online response, volume of transaction, recovery from failure etc., system testing is done to ensure that the system satisfies all the user requirements.

CHAPTER – 5

SCREENSHOTS

AND

REPORTS

CHAPTER – 5

REPORTS

5.1 SCREENSHOTS AND REPORTS:

HOME PAGE:

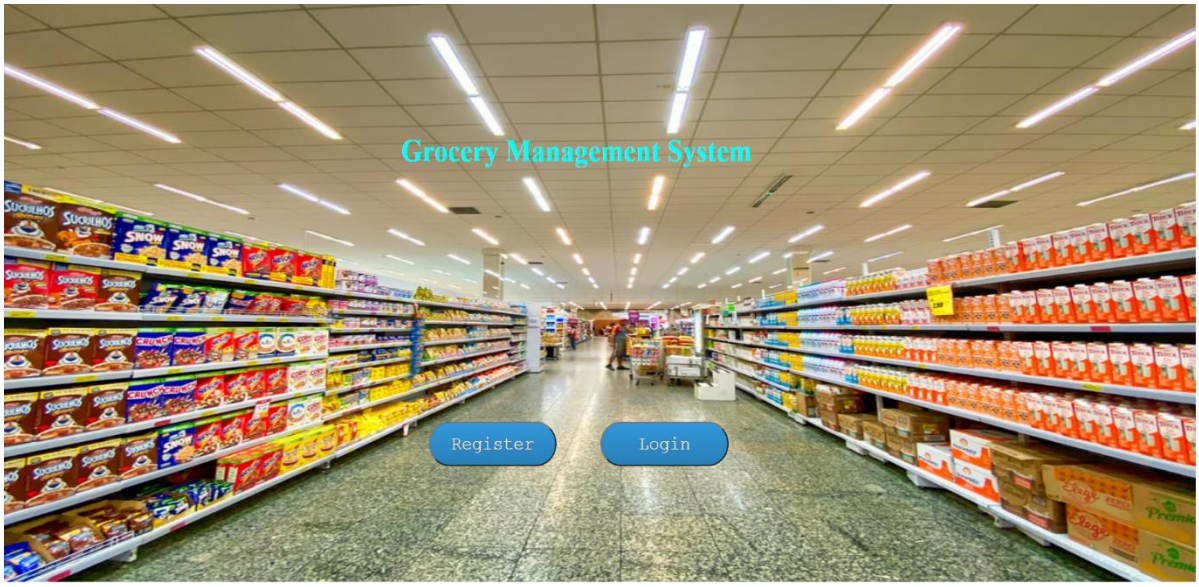

FIG 5.1.1 HOME PAGE

LOGINPAGE

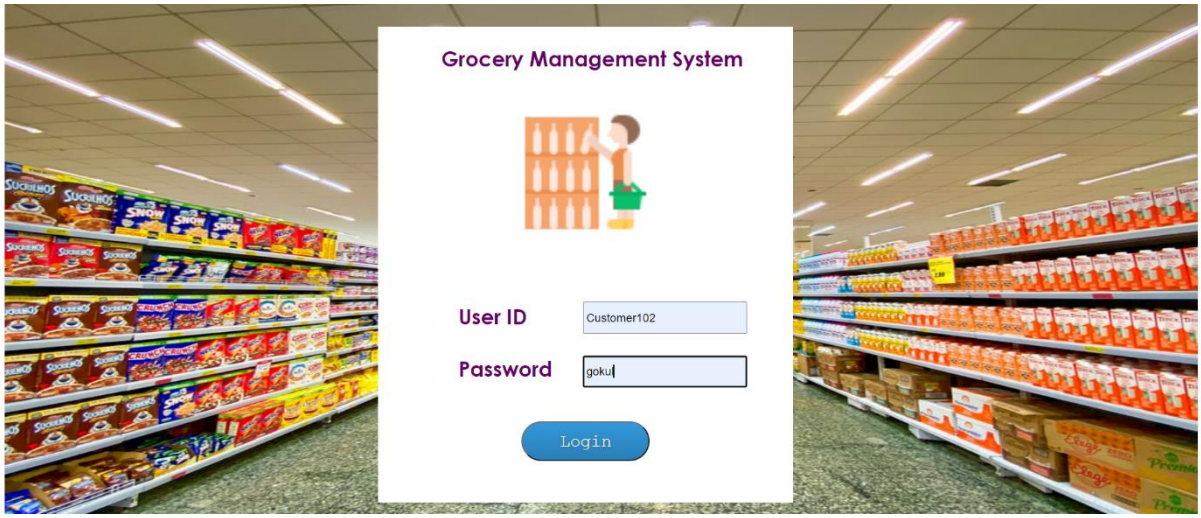

FIG 5.1.2 LOGIN PAGE

REGISTRATION

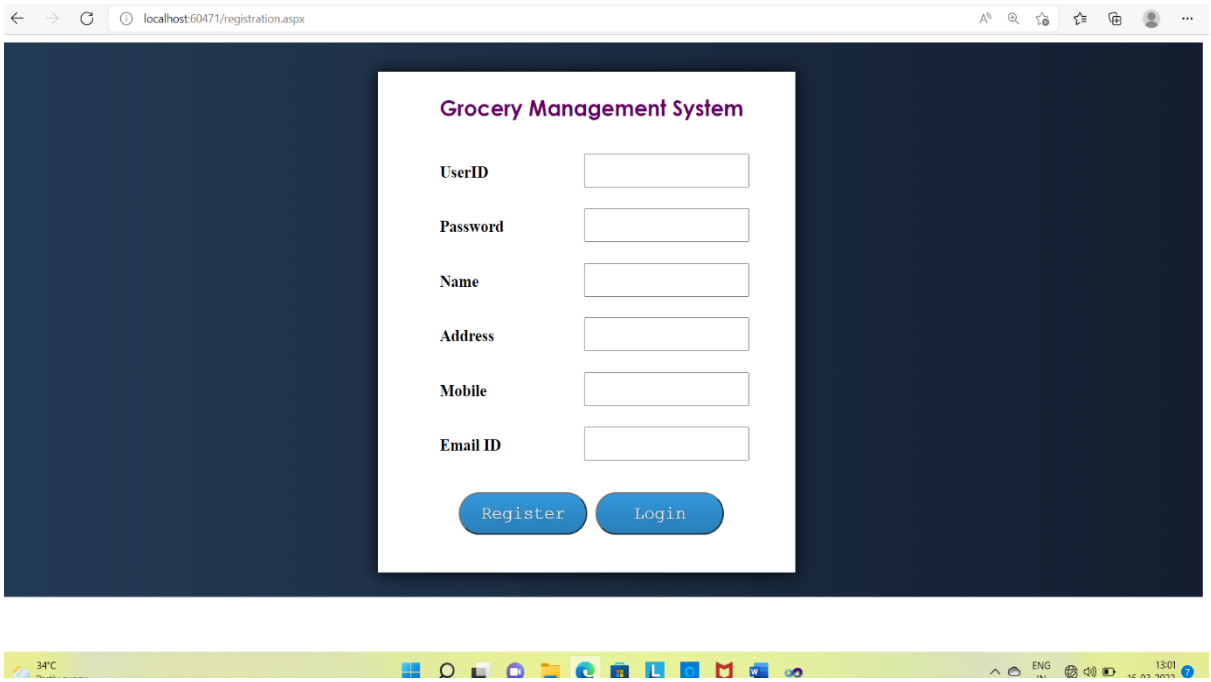

FIG 5.1.3 REGISTRATION PAGE

ADMIN MENU

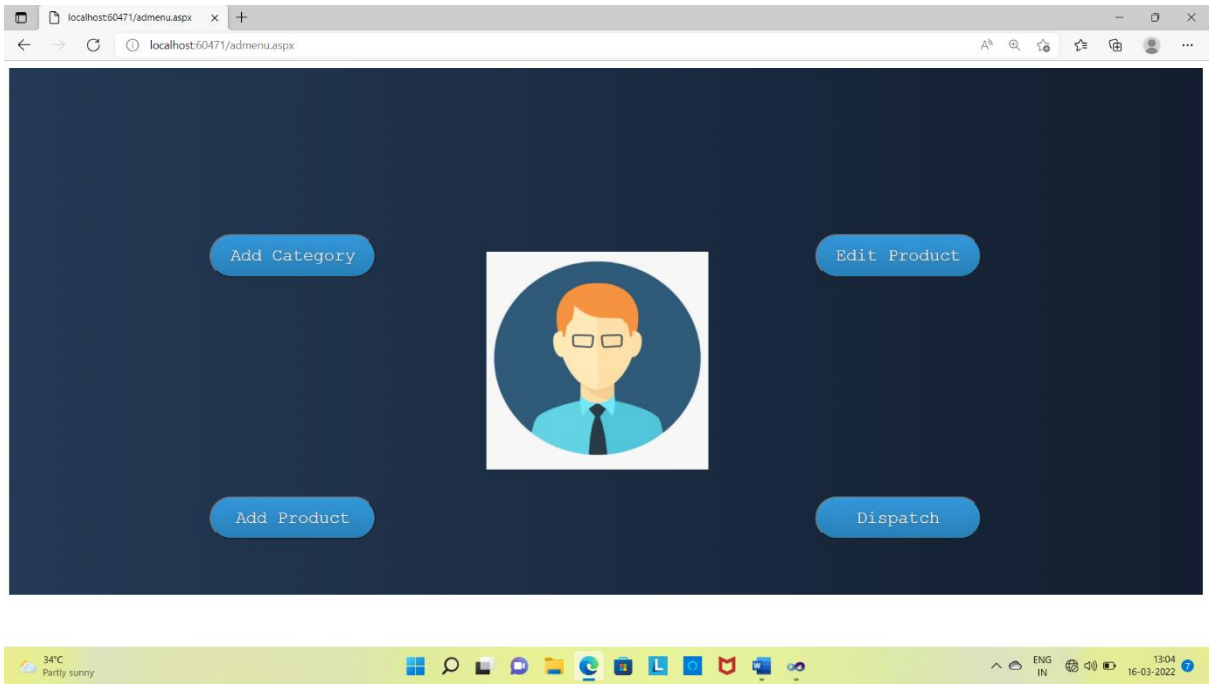

FIG 5.1.4 ADMIN PAGE

ADDCATEGORY

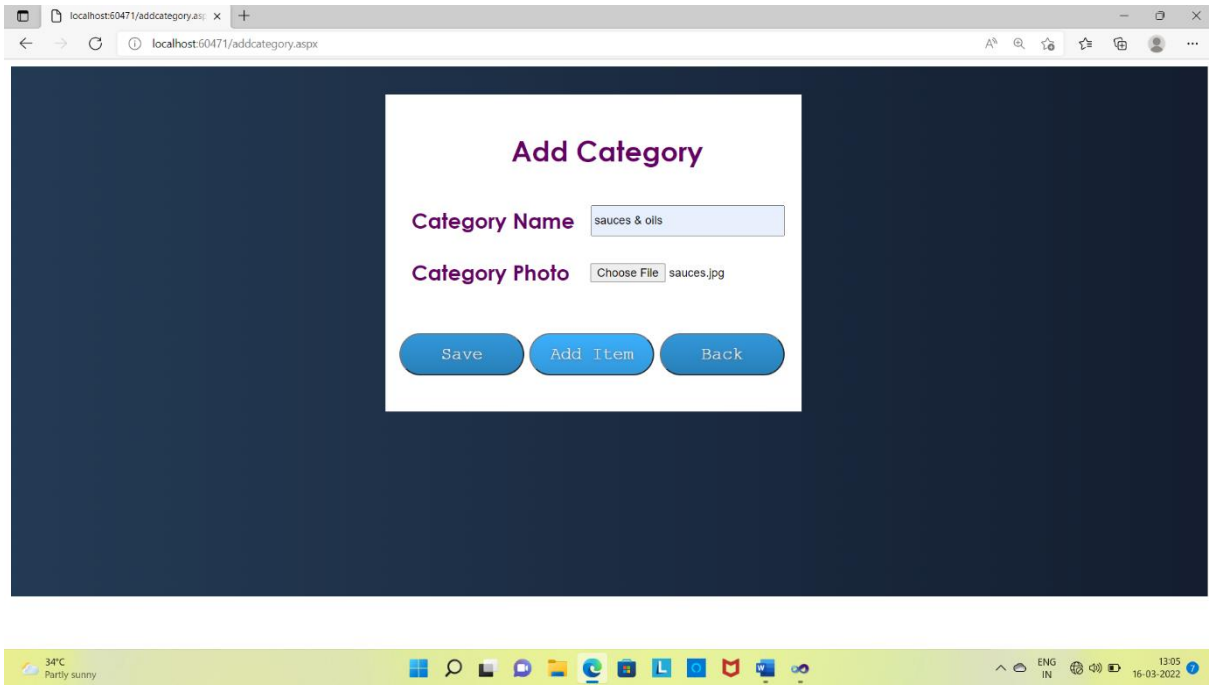

FIG 5.1.5 ADD CATEGORY PAGE

ADD PRODUCT

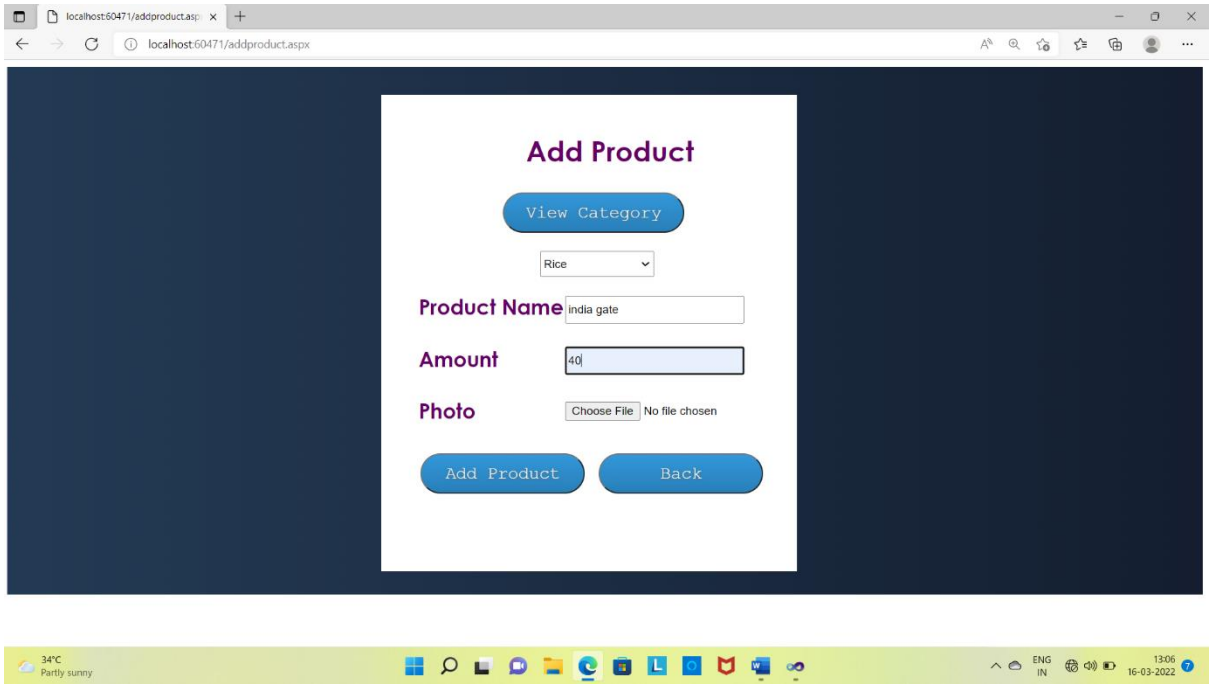

FIG 5.1.6 ADD PRODUCT PAGE

EDIT PRODUCT

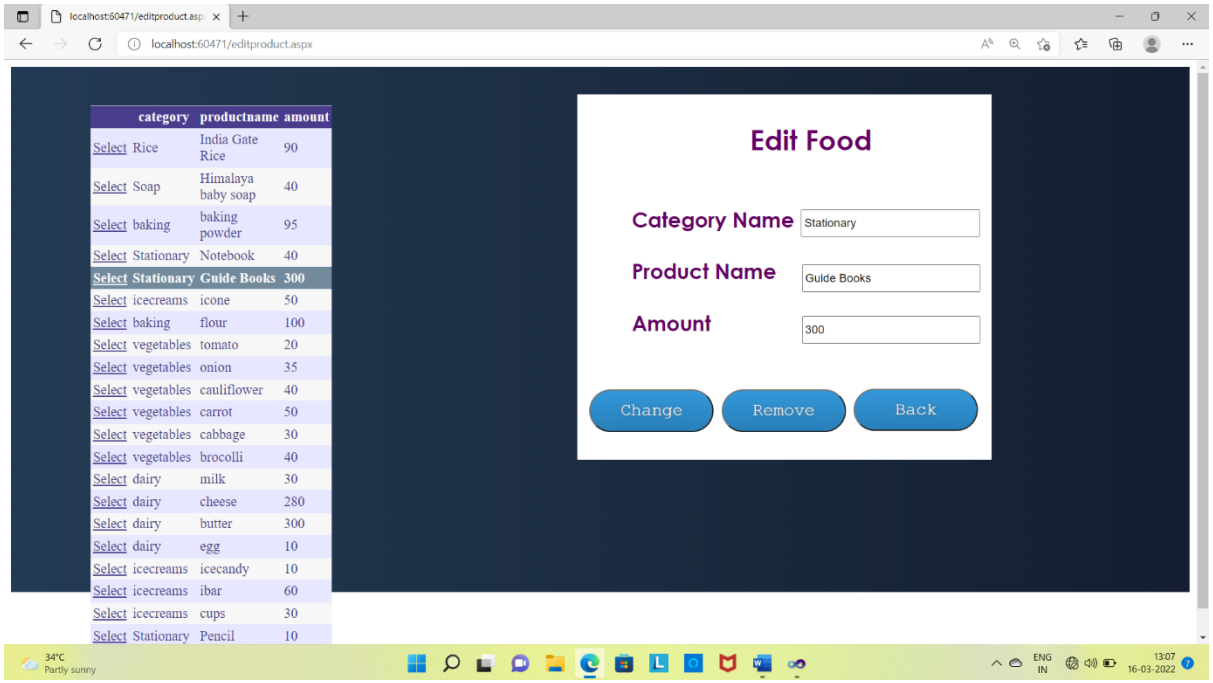

FIG 5.1.7 EDIT PRODUCT PAGE

DISPATCH

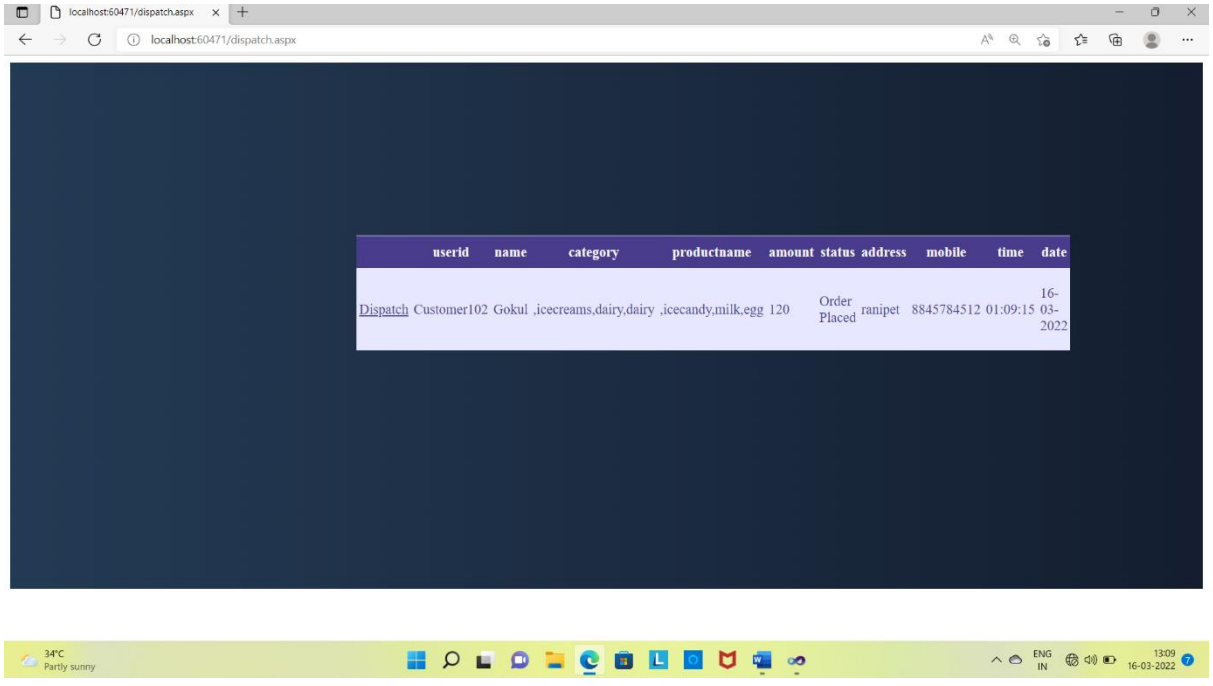

FIG 5.1.8 DISPATCH PAGE

CUSTOMER MENU

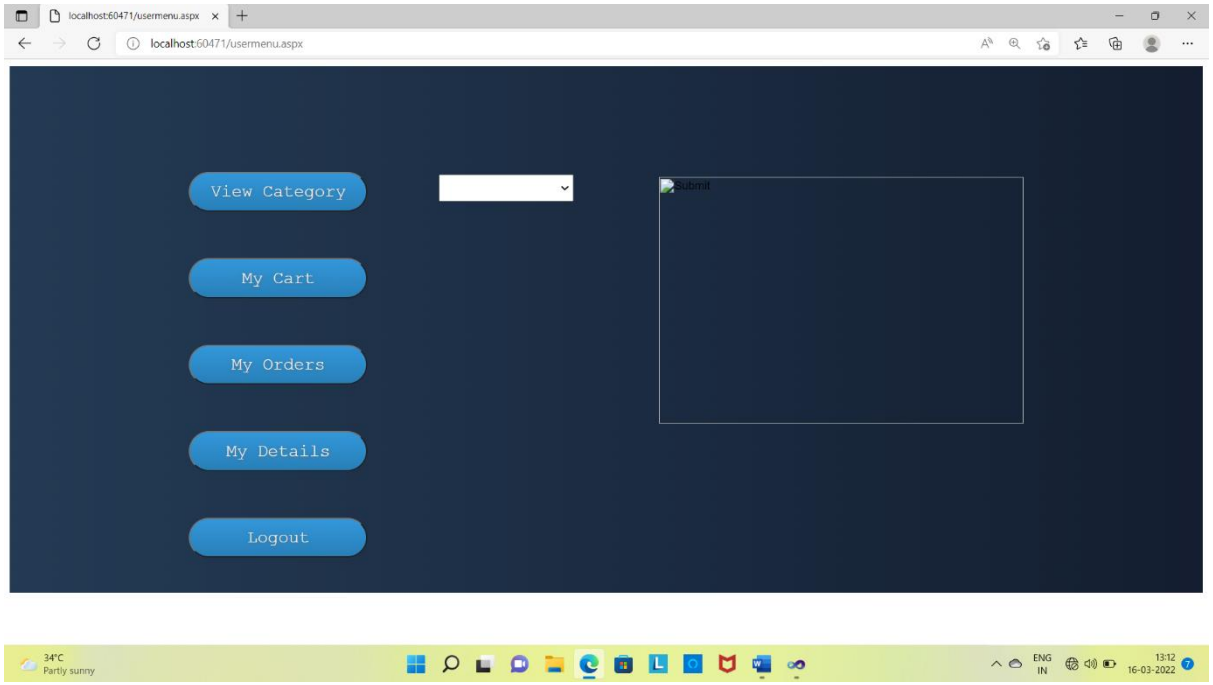

FIG 5.1.9 CUSTOMER PAGE

ORDER PRODUCT

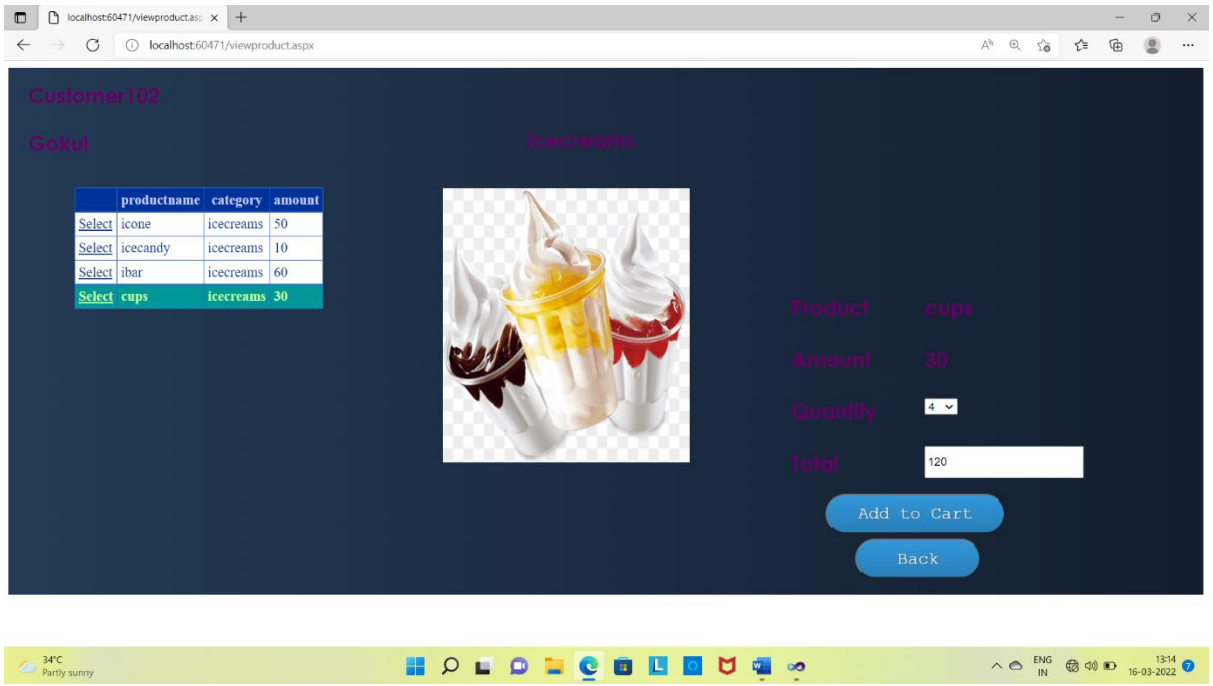

FIG 5.1.10 ORDER PRODUCT PAGE

PLACE ORDER

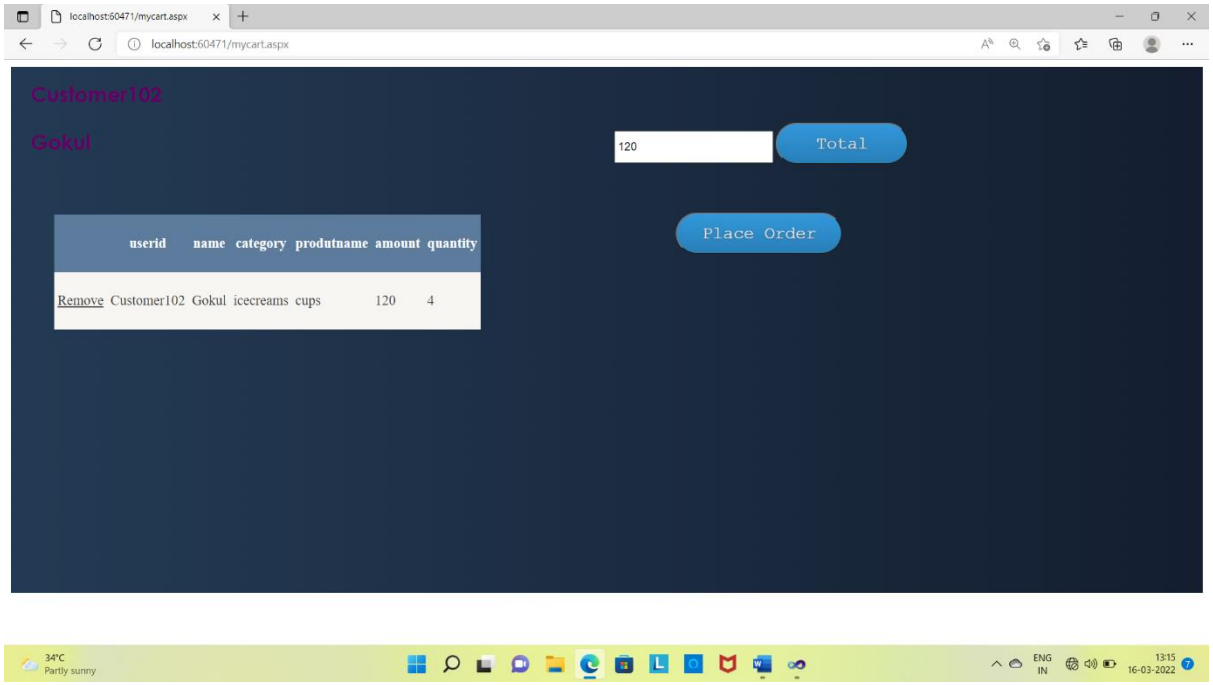

FIG 5.1.11 ORDER PLACING PAGE

VIEW ORDER

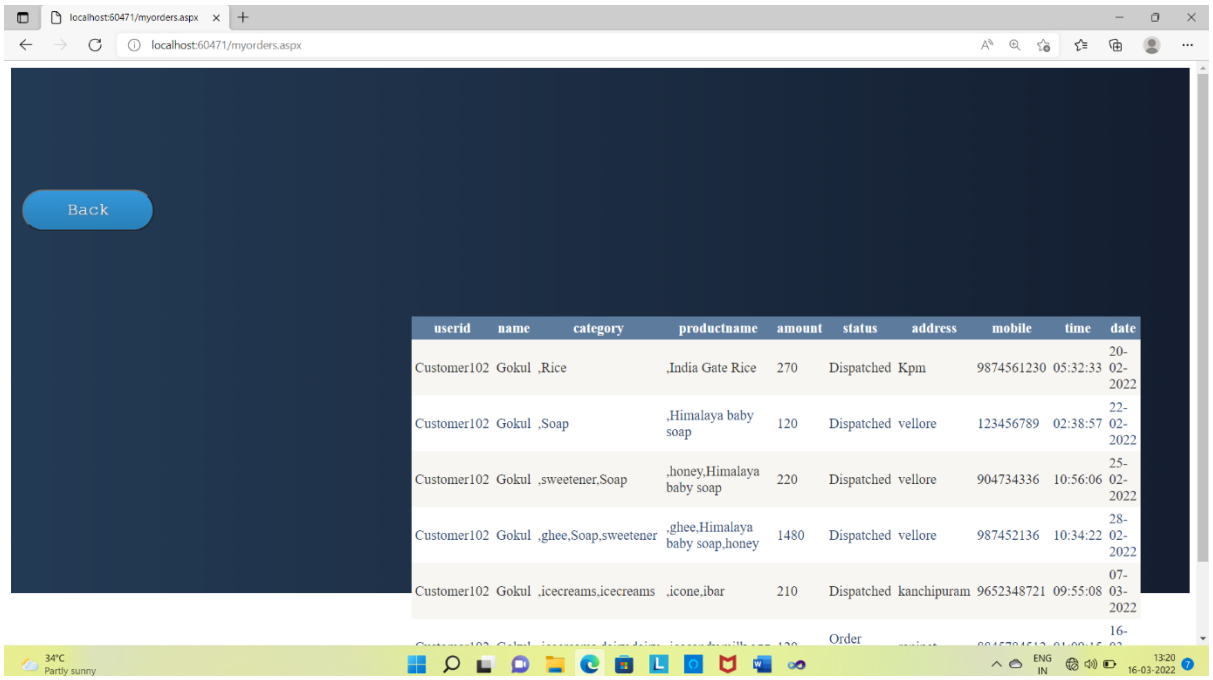

FIG 5.1.12 VIEW ORDER PAGE

MY DETAILS

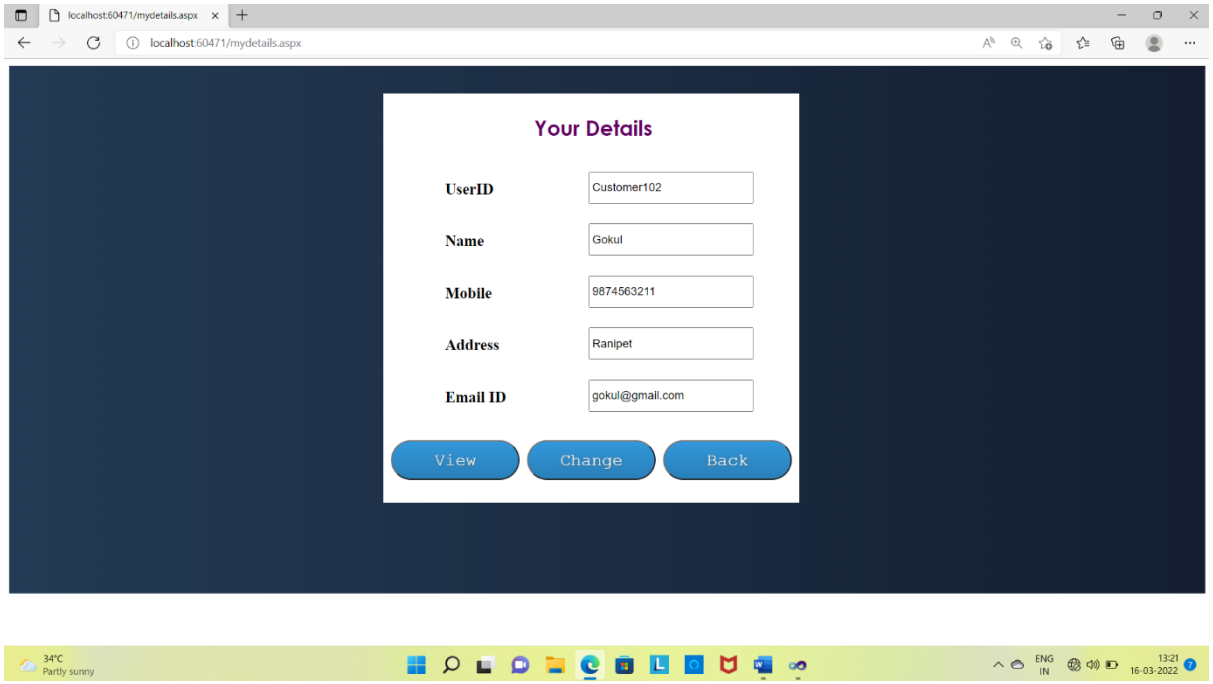

FIG 5.1.13 MY DETAILS PAGE

CHAPTER – 6

CONCLUSION

CHAPTER – 6

CONCLUSION:

Finally, in the grocery management system, we have a system where users order the item according to the wish given by the category he chooses and specify the number of it.

By implementing this grocery management system we are getting the more flexibility for the users. Which can operate from the home itself.

In conclusion, Grocery management system has to do with making appropriate effort to stop the rising problem to all manual operation in order to enhance the operation of such supermarket.

CHAPTER – 7

SCOPE FOR FURTHER DEVELOPMENT

CHAPTER – 7

SCOPE FOR FURTHER DEVELOPMENT:

The project has a very vast scope in future. The project can be implemented on intranet in future. Project can be updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion. With the proposed software of database Space Manager ready and fully functional the client is now able to manage and hence run the entire work in a much better, accurate and error free manner.

CHAPTER – 8

BIBILOGRAPHY AND REFERENCES

CHAPTER – 8

BIBILOGRAPHY AND REFERENCES

References:

- www.w3schools.com
- www.stackoverflow.com
- www.tutorialspoint.com
- www.youtube.com

APPENDIX

SAMPLE CODING:

Front Page:

```
Public Class index
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        Response.Redirect("registration.aspx")
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button2.Click
        Response.Redirect("login.aspx")
    End Sub
End Class
```

Login page

```
Imports System.Data.SqlClient
Public Class login1
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data
Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integra
ted Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        Dim s As String
        s = "select * from usertable where id='" & TextBox1.Text & "' and password='" &
TextBox2.Text & "';"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        If red.Read() = True Then
            If red.Item("password") <> TextBox2.Text Then
                MsgBox("Invalid Password")
            Else
```

```
            red.Close()
            Dim st As String
            st = TextBox1.Text
            If Mid(st, 1, 1) = "A" Then
               Response.Redirect("admenu.aspx")
               MsgBox("Login Successfully")
            ElseIf Mid(st, 1, 1) = "C" Then
               Session("userid") = TextBox1.Text
               Response.Redirect("usermenu.aspx")
               MsgBox("Login Successfully")
            End If
         End If
      End If
      con.Close()
   End Sub
End Class
```

Registration:

```
Imports System.Data.SqlClient
Public Class registration
   Inherits System.Web.UI.Page
   Dim con As New SqlConnection("Data Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True")
   Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

   End Sub

   Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button1.Click
      Dim s, s1 As String
      Dim x As Integer
      s = "select * from usertable where id like 'C%';"
      s1 = "Customer"
      Dim cmd As New SqlCommand(s, con)
      Dim red As SqlDataReader
      con.Open()
      red = cmd.ExecuteReader()
      While (red.Read)
         x = x + 1
      End While
      red.Close()
      s1 = s1 + Convert.ToString(101 + x)
      TextBox1.Text = s1
      s = "insert into usertable(id,password,name,address,mobileno,email) values('" & s1 & "','" & TextBox2.Text & "','" & TextBox3.Text & "','" & TextBox4.Text & "','" & TextBox5.Text & "','" & TextBox6.Text & "');"
      cmd = New SqlCommand(s, con)
```

```
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Your User ID is '" & s1 & "'")
        Response.Redirect("login.aspx")
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button2.Click
        Response.Redirect("login.aspx")
    End Sub
End Class
```

Admin menu:

```
Public Class admenu

    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        Response.Redirect("addcategory.aspx")
    End Sub

    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button3.Click
        Response.Redirect("editproduct.aspx")
    End Sub

    Protected Sub Button4_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button4.Click
        Response.Redirect("dispatch.aspx")
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button2.Click
        Response.Redirect("addproduct.aspx")
    End Sub
End Class
```

Add category:

```
Imports System.Data.SqlClient
Public Class addcategory
```

```
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button1.Click
        Dim s, str As String
        str = FileUpload1.FileName
        FileUpload1.SaveAs(Server.MapPath("~/images/" + str))
        s = "insert into category(category,photo)values('" & TextBox1.Text & "','" & str & "');"
        Dim cmd As New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Category Added Successfully")
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button2.Click
        Response.Redirect("addproduct.aspx")
    End Sub

    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button3.Click
        Response.Redirect("admenu.aspx")
    End Sub
End Class
```

Add product:

```
Imports System.Data.SqlClient
Public Class addproduct
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button2.Click
        Dim s, str As String
        str = FileUpload1.FileName
        FileUpload1.SaveAs(Server.MapPath("~/images/" + str))
        s = "insert into producttable(category,productname,amount,photo)values('" & DropDownList1.SelectedValue & "','" & TextBox1.Text & "','" & TextBox2.Text & "','" & str & "');"
```

```
        Dim cmd As New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Food Added Successfully")
    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        Dim s As String
        s = "select * from category;"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        While red.Read
            DropDownList1.Items.Add(red.GetString(0))
        End While
        con.Close()
        red.Close()
    End Sub

    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button3.Click
        Response.Redirect("admenu.aspx")
    End Sub
End Class
```

Edit product:

```
Imports System.Data.SqlClient
Public Class editproduct
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data
Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integrated
Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load

    End Sub

    Protected Sub GridView1_SelectedIndexChanged(ByVal sender As Object, ByVal e As EventArgs)
Handles GridView1.SelectedIndexChanged
        TextBox1.Text = GridView1.SelectedRow.Cells(1).Text
        TextBox2.Text = GridView1.SelectedRow.Cells(2).Text
        TextBox3.Text = GridView1.SelectedRow.Cells(3).Text
    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        Dim s As String
        s = "update producttable set amount='" & TextBox3.Text & "' where category='" & TextBox1.Text
& "' and productname='" & TextBox2.Text & "';"
```

```
        Dim cmd As New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Details Updated Successfully")

    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button2.Click
        Dim s As String
        s = "delete from producttable where productname='" & TextBox2.Text & "' and category='" & TextBox1.Text & "';"
        Dim cmd As New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Deleted Successfully")
    End Sub

    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button3.Click
        Response.Redirect("admenu.aspx")
    End Sub
End Class
```

Dispatch:

```
Imports System.Data.SqlClient
Public Class dispatch
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

    End Sub

    Protected Sub GridView1_SelectedIndexChanged(ByVal sender As Object, ByVal e As EventArgs) Handles GridView1.SelectedIndexChanged
        Dim s, id, amt, sta As String
        id = GridView1.SelectedRow.Cells(1).Text
        amt = GridView1.SelectedRow.Cells(5).Text
        sta = "Dispatched"
        s = "update ordertable set status='" & sta & "' where userid='" & id & "' and amount='" & amt & "';"
        Dim cmd As New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Done", MsgBoxStyle.OkCancel)
```

```
    End Sub
End Class
```

Customer menu:

```
Imports System.Data.SqlClient
Public Class usermenu
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button1.Click
        Dim s As String
        s = "select * from category;"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        DropDownList1.Items.Add("Select Category")
        While red.Read
            DropDownList1.Items.Add(red.GetString(0))
        End While
        con.Close()
        red.Close()
    End Sub

    Protected Sub DropDownList1_SelectedIndexChanged(ByVal sender As Object, ByVal e As EventArgs) Handles DropDownList1.SelectedIndexChanged
        ImageButton1.Visible = True
        Dim s As String
        s = "select * from category where category='" & DropDownList1.SelectedValue & "';"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        If red.Read Then
            ImageButton1.ImageUrl = "~/images/" + red.GetString(1)
        End If
        con.Close()
        red.Close()
    End Sub

    Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
```

```
        Session("cat") = DropDownList1.SelectedValue
        Response.Redirect("viewproduct.aspx")
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button2.Click
        Response.Redirect("mycart.aspx")
    End Sub

    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button3.Click
        Response.Redirect("myorders.aspx")
    End Sub

    Protected Sub Button4_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button4.Click
        Response.Redirect("mydetails.aspx")
    End Sub

    Protected Sub Button5_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button5.Click
        Response.Redirect("login.aspx")
    End Sub
End Class
```

View product:

```
Imports System.Data.SqlClient
Public Class viewproduct
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load
        Label1.Text = Session("cat").ToString
        Label2.Text = Session("userid").ToString
        Dim s As String
        s = "select * from usertable where id='" & Label2.Text & "';"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        If red.Read Then
            Label3.Text = red.GetString(2)
        End If
        con.Close()
    End Sub
    Dim con As New SqlConnection("Data
Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integra
ted Security=True;Connect Timeout=30;User Instance=True")
```

```
    Protected Sub GridView1_SelectedIndexChanged(ByVal sender As Object, ByVal e As
EventArgs) Handles GridView1.SelectedIndexChanged
        Dim s, c, n As String
        c = GridView1.SelectedRow.Cells(2).Text
        n = GridView1.SelectedRow.Cells(1).Text
        s = "select * from producttable where category='" & c & "' and productname='" & n & "';"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        If red.Read Then
            Image1.ImageUrl = "~/images/" + red.GetString(3)
            Label5.Text = red.GetValue(2)
            Label4.Text = red.GetString(1)
        End If
        con.Close()
        red.Close()

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        If TextBox1.Text = "" Then
            MsgBox("Please press total to view the amount", MsgBoxStyle.Exclamation)
        Else
            Dim s As String
            s = "insert into carttable(userid,name,category,produtname,quantity,amount)values('"
& Label2.Text & "','" & Label3.Text & "','" & Label1.Text & "','" & Label4.Text & "'," &
DropDownList1.SelectedValue & ",'" & TextBox1.Text & "');"
            Dim cmd As New SqlCommand(s, con)
            con.Open()
            cmd.ExecuteNonQuery()
            con.Close()
            MsgBox("Added to Cart")
            Response.Redirect("mycart.aspx")
        End If
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button2.Click
        Response.Redirect("usermenu.aspx")
    End Sub

    Protected Sub DropDownList1_SelectedIndexChanged(ByVal sender As Object, ByVal e
As EventArgs) Handles DropDownList1.SelectedIndexChanged
        Dim amt, qt As Integer
        amt = Convert.ToInt32(Label5.Text)
        qt = Convert.ToInt32(DropDownList1.SelectedValue)
        TextBox1.Text = amt * qt
    End Sub
```

```
End Class
```

My cart:

```
Imports System.Data.SqlClient
Public Class mycart
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
        Label1.Text = Session("userid").ToString
        Dim s As String
        s = "select * from usertable where id='" & Label1.Text & "';"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        If red.Read Then
            Label2.Text = red.GetString(2)
        End If
        con.Close()
        Label3.Visible = False
        Label4.Visible = False
        TextBox2.Visible = False
        TextBox3.Visible = False
        Button3.Visible = False
        Button4.Visible = False
    End Sub

    Protected Sub GridView1_SelectedIndexChanged(ByVal sender As Object, ByVal e As EventArgs) Handles GridView1.SelectedIndexChanged
        Dim s, fn As String
        fn = GridView1.SelectedRow.Cells(4).Text
        s = "delete from carttable where userid='" & Label1.Text & "' and produtname='" & fn & "';"
        Dim cmd As New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Food Removed")
    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button1.Click
        Dim s As String
        Dim total As Integer
        s = "select * from carttable where userid='" & Label1.Text & "';"
        Dim cmd As New SqlCommand(s, con)
```

```
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        While red.Read
            total = total + Convert.ToInt32(red.GetString(4))
        End While
        TextBox1.Text = total
        con.Close()
        red.Close()
        Button2.Visible = True
        If TextBox1.Text = "0" Then
            Button2.Visible = False
            Button3.Visible = False
            Button4.Visible = False
            MsgBox("Please Add the Food to the Cart", MsgBoxStyle.Information)
        End If
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button2.Click
        Button3.Visible = True
        Button4.Visible = True
        Label3.Visible = True
        Label4.Visible = True
        TextBox2.Visible = True
        TextBox3.Visible = True
    End Sub

    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button3.Click
        Dim s, c, fn, sta, time, da As String
        da = Today.Date
        time = DateAndTime.Now.ToString("hh:mm:ss").ToString
        sta = "Order Placed"
        s = "select * from carttable where userid='" & Label1.Text & "';"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        red = cmd.ExecuteReader
        While red.Read
            c = c + "," + red.GetString(2)
            fn = fn + "," + red.GetString(3)
        End While
        con.Close()
        red.Close()
        s                    =                    "insert                    into
ordertable(userid,name,category,productname,amount,status,address,mobile,time,date)values
('" & Label1.Text & "','" & Label2.Text & "','" & c & "','" & fn & "','" & TextBox1.Text & "','"
& sta & "','" & TextBox2.Text & "','" & TextBox3.Text & "','" & time & "','" & da & "');"
        cmd = New SqlCommand(s, con)
```

```
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        s = "delete from carttable where userid='" & Label1.Text & "';"
        cmd = New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Your Order Placed")
        Response.Redirect("usermenu.aspx")

    End Sub

    Protected Sub Button4_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button4.Click
        Response.Redirect("usermenu.aspx")
    End Sub
End Class
```

My orders:

```
Public Class myorders
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        Response.Redirect("usermenu.aspx")
    End Sub
End Class
```

My details:

```
Imports System.Data.SqlClient
Public Class mydetails
    Inherits System.Web.UI.Page
    Dim con As New SqlConnection("Data
Source=.\SQLEXPRESS;AttachDbFilename=C:\grocery\grocery\App_Data\cafe.mdf;Integra
ted Security=True;Connect Timeout=30;User Instance=True")
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Me.Load

    End Sub
```

```
    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button1.Click
        Dim s As String
        s = "select * from usertable where id='" & Session("userid") & "';"
        Dim cmd As New SqlCommand(s, con)
        Dim red As SqlDataReader
        con.Open()
        TextBox1.Text = Session("userid").ToString
        red = cmd.ExecuteReader
        If red.Read Then
            TextBox2.Text = red.GetString(2)
            TextBox3.Text = red.GetString(3)
            TextBox4.Text = red.GetString(4)
            TextBox5.Text = red.GetString(5)
        End If
        con.Close()
    End Sub

    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button3.Click
        Dim s As String
        s = "update usertable set address='" & TextBox3.Text & "', mobileno='" & TextBox4.Text
& "',email='" & TextBox5.Text & "' where id='" & TextBox1.Text & "';"
        Dim cmd As New SqlCommand(s, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        MsgBox("Details Modified")
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
Button2.Click
        Response.Redirect("usermenu.aspx")
    End Sub
End Class
```