**WASTE FOOD MANAGEMENT SYSTEM**

**Food In Need Donate It – Don’t Waste**

**FindIt – Don’t Waste**

**Submitted in partial fulfillment of the requirements**

**for the award of the degree**

**of**

**BACHELOR OF COMPUTER APPLICATIONS**

**Of**

**Bangalore University**

**By**

**BHAIRESH.M (15BUSB7009)**

****

**May 2018**

**SCHOOL OF GRADUATE STUDIES**

**DEPARTMENT OF COMPUTER APPLICATIONS**

**SSMRV Degree College**

**BENGALURU - 560041**

**Rashtreeya Shikshana Samithi Trust**

**SSMRV Degree College**

4th T Block, Jayanagar, Bengaluru-560041.

****

**Department of Computer Applications**

**CERTIFICATE**

This is to certify that this project entitled “WASTE FOOD MANAGEMENT SYSTEM ” is a bonafide work done by BHAIRESH.M (15BUSB7009), submitted to Bangalore University, Bangalore for the award of Bachelor of Computer Applications during the academic year 2017-18.

**Internal Project Guide Head of the Department**

**Mrs.Suprita M Nayak Mrs. Shanthi Krishna**

**Submitted for the Viva-Voce Examination held on 9-05-18.**

**ACKNOWLEDGEMENT**

We are privileged to express our sincere gratitude to all those who helped us in various capacities in undertaking this project and devising the report.

We the student of SSMRV Degree College with reverence and profound gratitude, place on accord our indebtedness to esteem for having **Dr. S.Anil Kumar, Principal**, for giving us the support to do the Project. We also thank **Mrs.R.Shanthi Krishna,HOD, Department of BCA,** for having granted us the permission to proceed with our project.

We hereby take immense pleasure of thanking **Mrs. Suprita M Nayak, Assistant Professor, Department of BCA ,** who presented us with his valuable and resentful guidance during the development of the project work done. We also express our heartfelt thanks for their persistent encouragement throughout the progress of the work. We are greatly obliged to all the assistants for having made all the lab facilities available, to complete our project work on time.

Last but not the least, we thank all other faculties of our department who extended their helping hands at the difficult times.

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Chapter** | **Title** | **Page No.** |
|  | **Title Page..** |  |
|  | **Bonafied Certificate** |  |
|  | **Acknowledgement..** |  |
| **1** | **INTRODUCTION** | **2** |
|  | **1.1 About the Project.** | **2** |
|  | **1.2 Existing System And its Limitations..** | **2** |
|  | **1.3 Proposed System.** | **2** |
| **2** | **SYSTEM REQUIREMENTS** | **3** |
|  | **2.1 Hardware Specification.** | **3** |
|  | **2.2 Software Specification.** | **3** |
|  | **2.2.1 Operating System.** | **3** |
|  | **2.2.2 Language / Packages.** | **6** |
| **3** | **SYSTEM DESIGN.** | **10** |
|  | **3.1 Code Design.** | **11** |
|  | **3.2 Input Design.** | **11** |
|  | **3.3 Output Design.** | **12** |
|  | **3.4 Database Design.** | **13** |
|  | **3.4.1 Tables.** | **13** |
|  | **3.4.2 E-R Diagram.** | **15** |
|  | **3.5 Data Flow Diagram.** | **18** |
| **4** | **SYSTEM IMPLEMENTATION AND TESTING.** | **21** |
|  | **4.1 System Implementation.** | **21** |
|  | **4.2 System Testing.** | **21** |
| **5** | **CONCLUSION.** | **24** |
| **6** | **FUTURE ENHANCEMENTS** | **25** |
|  | **BIBLIOGRAPHY.** | **25** |
|  | **APPENDIX-I : SOURCE CODE.** | **25** |
|  | **APPENDIX-II : SNAP SHOTS.** | **61** |

**SYNOPSIS**

**Title of the Project:**

**Waste Food Management System**

**Team members:**

**Bhairesh M(15BUSB7009)**

**Introduction:**

**Main Objective: This project is used to donate Food, Cloth or Fund(Money) to orphanages and mainly the Waste Food (i.e, Left out food in Hotels and Restuarants) is taken and delivered to the nearest Orphanages , So that the Food will help the people who are in need of it.**

**CHARACTERISTIC OF THE PROPOSED SYSTEM:**

1. **Easy to find Nearest Orphanages.**
2. **Application is user friendly.**
3. **Anyone can donate food, cloths or money which helps the people who are in need .**
4. **Can search the Orphanages in the Bangalore.**

**MODULES:**

**Chapter–1**

**Introduction**

***1.1 About the project***

The main objective of **Billing and Invoice Management** is to allow the shop merchant to bill the products purchased by the customers.

Basically **Billing and Invoice Management** contains three modules they are:

1. New Bill or Invoice: Used to add, delete and preview the data of the products purchased by the customers.
2. Reports: Produce reports from the selected dates. Printing the reports generated is also possible.
3. Bill Copy: Retrieving the bills from the database according to the given bill number.

**1.2 Existing System & Its Limitations**

The present system is complex and complicated to the users who use it for the first time. That is when the user is new to the computer field., and is required to operate the existing system it will be difficult for them.

* Complex and confusing for freshers.
* Time Consuming.

**1.3 Proposed System**

* Strain of manual labor can be reduced
* Keeps track of products ordered
* Easy to handle

**Chapter-2**

**System Requirement**

**2.1 Hardware Requirements(Min)/Configuration:**

Processor: Dual Core

RAM: 512MB

Hard Disk: 50MB

**2.2 Software Requirements/Configuration:**

Operating System: Windows 10.

Front-End: Microsoft Visual Studio 12.

Back-End: MYSQL Server 5.7v

**2.2.1 OPERATING SYSTEM - WINDOWS 10:**

**Windows 10** is a [personal computer](https://en.wikipedia.org/wiki/Personal_computer) [operating system](https://en.wikipedia.org/wiki/Operating_system) developed and released by [Microsoft](https://en.wikipedia.org/wiki/Microsoft) as part of the [Windows NT](https://en.wikipedia.org/wiki/Windows_NT) family of operating systems. It was first released on July 29, 2015. Unlike previous versions of Windows, Microsoft has branded Windows 10 as a "service" that receives ongoing "feature updates". Devices in enterprise environments can receive these updates at a slower pace, or use long-term support milestones that only receive critical updates, such as security patches, over their ten-year lifespan of extended support.

USER INTERFACE AND DESKTOP: A new iteration of the [Start menu](https://en.wikipedia.org/wiki/Start_menu) is used on the Windows 10 desktop, with a list of places and other options on the left side, and tiles representing applications on the right. The menu can be resized, and expanded into a full-screen display, which is the default option in Tablet mode. A new virtual desktop system was added. A feature known as Task View displays all open windows and allows users to switch between them, or switch between multiple workspaces. Windows Store apps, which previously could be used only in full screen mode, can now be used in self-contained windows similarly to other programs. Program windows can now be snapped to quadrants of the screen by dragging them to the corner. When a window is snapped to one side of the screen, Task View appears and the user is prompted to choose a second window to fill the unused side of the screen (called "Snap Assist"). Windows' system icons were also changed.

SYSTEM SECURITY : The enterprise version of Windows 10 offers additional security features; administrators can set up policies for the automatic [encryption](https://en.wikipedia.org/wiki/Data_encryption) of sensitive data, selectively block applications from accessing encrypted data, and enable *Device Guard*‍—‌a system which allows administrators to enforce a high security environment by blocking the execution of software that is not digitally signed by a trusted vendor or Microsoft. Device Guard is designed to protect against [zero-day exploits](https://en.wikipedia.org/wiki/Zero-day_exploit), and runs inside a [hypervisor](https://en.wikipedia.org/wiki/Hypervisor) so that its operation remains separated from the operating system itself.

### STORAGE REQUIREMENTS:

To reduce the storage footprint of the operating system, Windows 10 automatically [compresses](https://en.wikipedia.org/wiki/Data_compression) system files. The system can reduce the storage footprint of Windows by approximately 1.5 GB for [32-bit](https://en.wikipedia.org/wiki/32-bit) systems and 2.6 GB for [64-bit](https://en.wikipedia.org/wiki/64-bit_computing) systems. The level of compression used is dependent on a performance assessment performed during installations or by [OEMs](https://en.wikipedia.org/wiki/Original_equipment_manufacturer), which tests how much compression can be used without harming operating system performance. Furthermore, the Refresh and Reset functions use runtime system files instead, making a separate recovery partition redundant, allowing patches and updates to remain installed following the operation, and further reducing the amount of space required for Windows 10 by up to 12 GB. These functions replace the [WIMBoot mode](https://en.wikipedia.org/wiki/Features_new_to_Windows_8#Installation) introduced on Windows 8.1 Update, which allowed OEMs to configure low-capacity devices with flash-based storage to use Windows system files out of the compressed [WIM image](https://en.wikipedia.org/wiki/Windows_Imaging_Format) typically used for installation and recovery. Windows 10 also includes a function in its Settings app that allows users to view a breakdown of how their device's storage capacity is being used by different types of files, and determine whether certain types of files are saved to internal storage or an [SD card](https://en.wikipedia.org/wiki/SD_card) by default.

SYSTEM REQUIREMENTS: The basic hardware [requirements](https://en.wikipedia.org/wiki/System_requirements) to install Windows 10 are the same as for Windows 8.1 and Windows 8, and only slightly higher than Windows 7. The 64-bit versions require a CPU that supports certain instructions. Devices with low storage capacity must provide a USB flash drive or SD card with sufficient storage for temporary files during upgrades.

**2.2.2 LANGUAGE/PACKAGES**

**MICROSOFT VISUAL STUDIO 12:**

The Visual Studio is a set of freeware integrated development environment (IDE) developed by Microsoft that are lightweight versions of the Microsoft Visual Studio product line.

It consists of the following elements:

Displays the commands you use to work with Visual Basic. Standard file, edit, view, window, and help menus, menus area provided to access functions specific to programming such as project, format or debug.

Contain shortcuts to frequently performed actions. To open a context menu, click the right mouse button on the object you are using. The specific list of shortcuts available from context menus depends on the part of the environment where you click a right mouse button. For example, the context menu displayed when you right click on the Toolbox lets you display the components dialog box, hide the Toolbox, dock or undock the Toolbox, or add a custom tab to the toolbox.

Provide quick access to commonly used commands in the programming environment. You click a button on the toolbar once to carry out the actions represented by that button. By default, the standard toolbar is displayed when you start visual basic. Additional toolbars for editing, form design debugging can be toggled on or off from the toolbars command on the view menu.

Toolbars can be docked beneath the menu bar or can “float” if you select the vertical bar on the edge end drag it away from the menu bar.

Provides a set of tools you use at design time to place control on a form. In addition to the default toolbox layout, you can create your own custom layouts by selecting Add Tab from the context menu and adding controls menu adding controls to the resulting tab.

List the forms and modules in your current project. A project is the collection of files you use to build an application.

List the property settings for the selected form or control. A property is a characteristic of an object, such as size, caption, or color.

Serves as a window that you customize to design the interface of your application. You add controls, graphics, and pictures to a form to create the look you want. Each form in your application has its own form designer window.

Serves as an editor for entering application code. A separate code editor window is created for each form or code module in your application.

The form layout window allows you to position the forms in your application using a small graphical representation of the screen.

**SDI or MDI Interface**

Two different styles are available for the visual basic IDE:

Single document interface (SDI) or multiple document interface (MDI). With the SDI option, all of the IDE windows are free to be moved anywhere on screen as long as Visual Basic is the current application, they will remain on top of any other application. With the MDI option, all of the IDE windows are contained within a single resizable parent window.

**VB Coding**

The first step for creating an application with Visual Basic is to create the interface, the visual parts of the application with which the user will interact. Forms and controls are the basic building blocks used to create the interface; they are the objects that you will work with to build your application.

Forms are objects that expose properties, which define their appearance, methods, which define their behavior, and events, which define their application with the user. By setting the properties of the form and writing Visual Basic code to respond to its events, you customize the object to meet the requirements of your application.

Controls are objects that are contained within form objects. Each type of control has its own set properties, methods and events that make it suitable for a particular purpose. Some of the controls you can use in your applications are best suited for entering or displaying text. Other controls let you access other applications and process data as if the remote application was of your code.

This is the coding for the loading of the form. Here the progress bar loads as the timer value increases and the specified form is displayed.

**MYSQL Server 5.7 Edition**

**Microsoft MYSQL Server** is a relational database server, developed by Microsoft. It is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). There are at least a dozen different editions of Microsoft MYSQL Server aimed at different audiences and for different workloads (ranging from small applications that store and retrieve data on the same computer, to millions of users and computers that access huge amounts of data from the Internet at the same time).

MYSQL Server 5.7 helps developers build robust and reliable applications. Too often database systems are overly complex for building simple applications. Microsoft Visual Studio 12 and MY SQL Server reduces this complexity by providing a simple but powerful development environment for building data-driven applications. Developers can design schemas, add data, and query local databases, all inside the Visual Studio 2012 environment. If developers need more advanced database features, then MYSQL Server can be seamlessly upgraded to more sophisticated versions of MYSQL Server.

Supporting and maintaining software is a huge expense for application software developers. An embedded database helps reduce this cost by providing automated tuning and management, automated patching and servicing, and support for embedded setup and installation. MYSQL Server was specifically designed with these goals in mind. Using the MYSQL Server 5.7 database engine, MYSQL Server Express can be configured to dynamically tune database parameters to respond to changing usage characteristics. Additionally, software developers do not have to worry about directly deploying database updates because MYSQL Server uses Microsoft Update to directly deliver updates to end users. For enterprise users, MYSQL Server easily plugs in to most enterprise management environments. MYSQL Server also provides support for either silent installs or GUI-based setup and installations. This gives the software developer ultimate control and flexibility over the installation and setup of the embedded database.

**Chapter 3 SYSTEM DESIGN**

The most creative and challenging face of the system development is System Design. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Design goes through the logical and physical stages of development.

In designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. The first step is to determine how the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of the proposed output. The operational phases are handled through program construction and testing.

Design of a system can be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Thus system design is a solution to “how to” approach to the creation of a new system. Thus important phase provides the understanding and the procedural details necessary for implementing the system recommended in the feasibility study. The design step provides a data design, architectural design, and a procedural design.

**3.1 CODE DESIGN**

First phase of implementation is coding. Coding can be done in two ways. One by automatic program code and other by programmer’s manually written code. A code generator is a suite of programs that matches the input to an appropriate code template and from these produces modules of code. The code is made simple in such a way that another programmer can easily understand and work on that in future. The crucial phase in the system development life cycle is the successful implementation of the new system design. The process of converting as new or revised system into an operational one is known as system implementation. This includes all those activities that take place to convert from an old system to a new system. The system can be implemented only after a through testing is done and if it is found to work according to the specifications. The most crucial stage in achieving a new successful system and giving confident on the new system for the users is that it will work effectively and efficiently. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover.

**3.2 INPUT DESIGN**

In the input design, user-oriented inputs are converted into a computer based system format. It also includes determining the record media, method of input, speed of capture and entry on to the screen. Online data entry accepts commands and data through a keyboard. The major approach to input design is the menu and the prompt design. In each alternative, the user’s options are predefined. The data flow diagram indicates logical data flow, data stores, source and destination. Input data are collected and organized into a group of similar data. Once identified input media are selected for processing.

In this software, importance is given to develop Graphical User Interface (GUI), which is an important factor in developing efficient and user-friendly software. For inputting user data, attractive forms are designed. User can also select desired options from the menu, which provides all possible facilities.

Also the important input format is designed in such a way that accidental errors are avoided. The user has to input only just the minimum data required, which also helps in avoiding the errors that the users may make. Accurate designing of the input format is very important in developing efficient software. The goal or input design is to make entry as easy, logical and free from errors.

**3.3 OUTPUT DESIGN**

In the output design, the emphasis is on producing a hard copy of the information requested or displaying the output on the CRT screen in a predetermined format. Two of the most output media today are printers and the screen. Most users now access their reports from a hard copy or screen display. Computer’s output is the most important and direct source of information to the user, efficient, logical, output design should improve the systems relations with the user and help in decision-making.

As the outputs are the most important source of information to the user, better design should improve the system’s relation and also should help in decision-making. The output device’s capability, print capability, print capability, response time requirements etc should also be considered form design elaborates the way output is presented and layout available for capturing information. It’s very helpful to produce the clear, accurate and speedy information for end users.

**3.4 DATABASE DESIGN**

Database design is about the logically implied data. Each and every data in the form can be designed in such a manner to understand the meaning. Database designing should give a clear understanding and idea about the related data used to construct a form.

**3.4.1 TABLES**

**1: registeration\_form**

|  |  |
| --- | --- |
| Field name | Data type |
| shop\_name | Varchar(50) |
| Gstin | Varchar(20) |
| tin\_no | Varchar(15) |
| address | Varchar(150) |
| Phone | Varchar(20) |
| State | Varchar(25) |
| City | Varchar(15) |
| pincode | Int(11) |
| Email | Varchar(50) |

**2:owner\_details**

|  |  |
| --- | --- |
| Field name | Data type |
| Name | Varchar(20) |
| Password | Varchar(6) |
| Securityquestion1 | Varchar(100) |
| Ans1 | Varchar(20) |
| Securityquestion2 | Varchar(100) |
| Ans2 | Varchar(20) |
|  |  |

**3: orderform**

|  |  |
| --- | --- |
| Field Name | Data Type |
| productname | varchar(100) |
| Uom | varchar(15) |
| quantity | int(20) |
| Price | float |
| Gst | float |
| subtotal | float |
| Dates | date |
| Billno | Int(11) |

**4:addproduct**

|  |  |
| --- | --- |
| **Field name** | **Data type** |
| **productname** | **varchar(100)** |
| **Price** | **float** |
| **quantity** | **int(10)** |
| **Uom** | **varchar(15)** |
| **Tax** | **int(11)** |
| **Dates** | **date** |
| **Pcode** | **int(3)** |

* + 1. **E-R Diagrams**

E-R-Diagram is a diagrammatic representation of the logical design of the database. In E-R diagram entities, attributes and relationship is represented.

**Entity:** An entity is a real time object. In databases the entity refers to the tables.

**Attribute:** An attribute is an characteristic representation of a column in a table as known as Fields.

**Relationship:** It is the association between one entity to another.

Different symbols used in drawing E-R Diagrams are:

Billing

has

has

generates

New Bill

Stock List

Bill Copy

New Bill

Ordering Items

Reports

Saves in

generates

Stock List

Add products

Modify Stock

Delete Stock

has

has

has

**3.5 DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. It’s a structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification,

specifies the implementation plan etc.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called node.

**Context Level DFD**

Customer

Bill

0

Billing and Invoice Management

**Context Level 1**

1.0

Billing and Invoice

2.0

Create New Bill

3.0

Stock Updation

4.0

Registration

5.0

Owner Details

6.0

Generate Bills

D2 Add Product

D4 Owner Details

D3 Registration Form

D1 Order Form

Customer

Bill

**Context Level2 DFD**

3.0

Stock Updation

3.1

Add Product

3.2

Delete Product

3.3

Update Product

D2 Add Products

Customer

Bill

**Chapter 4**

**SYSTEM IMPLEMENTATION AND TESTING**

**4.1 SYSTEM IMPLEMENTATION**

Implementation is the stage, which is crucial in the life cycle of the new system designed. Planning, training and system testing are the main stages in the implementation. Converting a new or revised system into an operational one is called implementation.

Implementation includes all those activities involving the conversion of an old system into a new system. The new system may be in a totally new concept or a revision of the old one. A proper implementation is required for reliable system, but still does not guarantee a successful system. Chances are there that if implementation is not proper the whole system may become a failure.

Conversion an important aspect of implementation is the process of change from the old system to the new one. A review is conducted once implementations are over. Information required for maintenance is collected during this phase. The basic review method is data collection methods of interview, observation, sampling and record inspection.

**4.2 SYSTEM TESTING**

Testing is a process to show the correctness of the program. Testing is needed to show completeness, it improve the quality of the software and to provide the maintenance aid. Some testing standards are therefore necessary reduce the testing costs and operation time. Testing software extends throughout the coding phase and it represents the ultimate review of configurations, design and coding. Based on the way the software reacts to these testing, we can decide whether the configuration that has been built is study or not. All components of an application are tested, as the failure to do so many results in a series of bugs after the software is put to use.

**Black box Testing:**

Black box testing, also called behavioral testing, focuses on the functional requirements of software. This testing approach enables the software engineer to derive the input conditions that will fully exercise all requirements for a program. Blackbox testing attempts to find the errors like

* Incorrect or missing functions
* Interface errors
* Errors in data structures or external database access
* Behavior or performance errors
* Initialization and termination errors

In Black box testing software is exercised over a full range of inputs and outputs are observed for correctness.

**White box Testing:**

White box testing is also called Glass box testing is a test case design control; structure of the procedural design to derive test cases using White box testing method, the software engineer can derive the test cases that guarantee that all independent paths within the module have been exercised at least once. Exercise all logic decisions on their true or false sides. Execute all loops at their boundaries and within their operational bounds. Exercise internal data structure to ensure their validity.

**Software Testing Strategies:**

Testing involves

* Unit testing
* Integration testing
* Acceptance testing

**UNIT TESTING:**

The first level of test is unit testing. The purpose of unit testing is to ensure that each program is fully tested.

Unit testing focuses verification effort on the smallest unit of software design – the module. The unit test is always white box oriented. The tests that occur as part of unit testing are testing the module interface, examining the local data structures, testing the boundary conditions, executing all the independent paths and testing error-handling paths.

**INTEGRATION TESTING:**

The second step is integration testing. In this individual program units or programs are integrated and tested as a complete system to ensure that the software requirements are met.

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. Scope of testing summarizes the specific functional, performance, and internal design characteristics that are to be tested. It employs top-down testing and bottom-up testing methods for this case.

**ACCEPTANCE TESTING:**

This is a system testing done by the user of the applications. The only emphases is functionality testing as the user is not aware of the technical aspect of the system. The testing is also done in a controlled environmental with logging off all error based on the error found in the system, the user has to accept or reject the system.

**5. CONCLUSION**

This project entitled “Billing and Invoice Management” has been developed efficiently to meet the needs of the users. It is developed using VB 2012and MYSQL SERVER 5.7. The system can be easily modified or expanded. The software provides all facilities required in the finished system, and the reports are user-friendly. The software requires less effort to operate. The assistance of computers will reduce the work done by the staff. The system provides flexibility for incorporating new features, which may be necessary in future. The system has been tested and implemented. Finally, we have concluded that system has capacity to do what it was expected to do.

**6. SCOPE OF THE FURTHER DEVELOPMENT**

Now the organization staffs can bill the purchases very easily. The software is very easy to access. This project can further be implemented as the online system so that the billing can be done online.

The application can also have user accounts, that is, the customers can also be given the access to their particular account to view the products, their prices and bill accordingly.

**BIBILIOGRAPHY**

* Websites, Stackoverflow
* DreamInCode
* CodeProject

**Source Code:**

**For New Bill**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class order\_form

Dim mysql As MySqlConnection

Dim cmd As MySqlCommand

Private Sub addbtn\_Click(sender As Object, e As EventArgs) Handles addbtn.Click

Dim reader As MySqlDataReader

Dim mysql As New MySqlConnection

Dim ada As New MySqlDataAdapter

Dim bgs As New BindingSource

Dim dt As New DataTable

Dim total As String = 0

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

Dim query1 As String

Dim query2 As String

Dim query3 As String

If ComboBox1.Text = "" Or TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or TextBox6.Text = "" Or TextBox4.Text = "" Then

MsgBox("Enter all the details")

Else

mysql.Open()

query = "insert into orderform(productname,uom,quantity,price,gst,subtotal,dates,billno) values('" & ComboBox1.Text & "','" & TextBox1.Text & "','" & TextBox2.Text & "','" & TextBox3.Text & "','" & TextBox6.Text & "','" & TextBox4.Text & "','" & DateTimePicker1.Text & "','" & TextBox7.Text & "')"

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

reader.Close()

'MsgBox("data saved", vbOKOnly)

query1 = "update addproduct set quantity=quantity-'" & TextBox2.Text & "' where productname='" & ComboBox1.Text & "'"

cmd = New MySqlCommand(query1, mysql)

reader = cmd.ExecuteReader

'MsgBox("updated succesfully")

reader.Close()

query3 = "select productname,uom,quantity,price,gst,subtotal from orderform where billno='" & TextBox7.Text & "'"

cmd = New MySqlCommand(query3, mysql)

ada.SelectCommand = cmd

ada.Fill(dt)

bgs.DataSource = dt

DataGridView1.DataSource = bgs

ada.Update(dt)

mysql.Close()

End If

ComboBox1.Text = ""

TextBox6.Text = ""

TextBox1.Text = ""

TextBox2.Text = ""

TextBox3.Text = ""

TextBox4.Text = ""

ComboBox1.Focus()

Catch ex As Exception

MsgBox(ex.Message)

End Try

End Sub

Private Sub Order\_form\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Dim mysql As New MySqlConnection

Dim reader As MySqlDataReader

Dim billno As Integer

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

Dim query1 As String

Dim query2 As String

mysql.Open()

query1 = "select max(billno) from orderform"

cmd = New MySqlCommand(query1, mysql)

If IsDBNull(cmd.ExecuteScalar) Then

billno = 1

TextBox7.Text = billno

Else

billno = cmd.ExecuteScalar + 1

TextBox7.Text = billno

End If

query = "select \* from billing.addproduct"

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

While reader.Read

Dim product = reader.GetString("productname")

ComboBox1.Items.Add(product)

End While

query2 = "select SUM(subtotal) from orderform where billno='" & TextBox7.Text & "'"

cmd = New MySqlCommand(query2, mysql)

reader = cmd.ExecuteReader

TextBox5.Text = reader.GetInt32("SUM(subtotal)")

mysql.Close()

Catch ex As Exception

End Try

Me.Show()

ComboBox1.Focus()

End Sub

Private Sub ComboBox1\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ComboBox1.SelectedIndexChanged

Dim mysql As New MySqlConnection

Dim reader As MySqlDataReader

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

query = "select \* from billing.addproduct where productname='" & ComboBox1.Text & "'"

mysql.Open()

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

While reader.Read

TextBox1.Text = reader.GetString("uom")

TextBox3.Text = reader.GetFloat("price")

TextBox6.Text = reader.GetInt32("tax")

End While

mysql.Close()

Catch ex As Exception

End Try

End Sub

Private Sub TextBox2\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox2.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

End If

End If

End Sub

Private Sub TextBox2\_TextChanged(sender As Object, e As EventArgs) Handles TextBox2.TextChanged

Dim GST As Double

Dim cmd As MySqlCommand

Dim reader As MySqlDataReader

Dim query As String

Dim mysql As New MySqlConnection

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

mysql.Open()

query = "select quantity from addproduct where productname = '" & ComboBox1.Text & "'"

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

While reader.Read

If reader(0) <= 0 Then

MsgBox("Out of stock", vbOKOnly)

ComboBox1.Text = ""

TextBox1.Text = ""

TextBox2.Text = ""

TextBox3.Text = ""

TextBox4.Text = ""

TextBox6.Text = ""

ComboBox1.Focus()

reader.Close()

mysql.Close()

Exit Sub

Else

TextBox4.Text = Val(TextBox2.Text) \* Val(TextBox3.Text)

GST = (TextBox6.Text / 100) \* TextBox4.Text

TextBox4.Text += GST

End If

End While

End Sub

Private Sub DataGridView1\_CellContentClick(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick

Dim i As Integer

i = DataGridView1.CurrentCell.RowIndex

TextBox8.Text = DataGridView1.Item(0, i).Value

End Sub

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim mysql As New MySqlConnection

Dim query As String

Dim reader As MySqlDataReader

Dim qty As Double

Dim i As Integer

Dim query1 As String

Dim query2 As String

Dim ada As New MySqlDataAdapter

Dim bgs As New BindingSource

Dim dt As New DataTable

Dim qty1 As Integer

i = DataGridView1.CurrentCell.RowIndex

qty = DataGridView1.Item(5, i).Value

qty1 = DataGridView1.Item(2, i).Value

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

mysql.Open()

query = "delete from orderform where subtotal='" & qty & "'"

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

reader.Close()

MsgBox("Item has been deleted")

TextBox8.Text = ""

TextBox8.Focus()

query1 = "update addproduct set quantity=quantity+'" & qty1 & "' where productname='" & TextBox8.Text & "' "

cmd = New MySqlCommand(query1, mysql)

reader = cmd.ExecuteReader

reader.Close()

query2 = "select productname,uom,quantity,price,gst,subtotal from orderform where billno='" & TextBox7.Text & "'"

cmd = New MySqlCommand(query2, mysql)

ada.SelectCommand = cmd

ada.Fill(dt)

bgs.DataSource = dt

DataGridView1.DataSource = bgs

ada.Update(dt)

mysql.Close()

End Sub

End Class

**For Stock list**

-**Addproduct form:**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class addproduct

Dim mysql As MySqlConnection

Dim cmd As MySqlCommand

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles listbtn.Click

stocklistfrm.Show()

Me.Hide()

End Sub

Private Sub addbtn\_Click(sender As Object, e As EventArgs) Handles addbtn.Click

Dim reader As MySqlDataReader

Dim mysql As New MySqlConnection

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

If TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or ComboBox1.Text = "" Or ComboBox2.Text = "" Then

If MsgBox("Enter all the details", vbOKOnly) Then

TextBox4.Focus()

End If

Else

query = "insert into addproduct(productname,price,quantity,uom,tax,dates,Pcode) values('" & TextBox1.Text & "','" & TextBox2.Text & "','" & TextBox3.Text & "','" & ComboBox1.Text & "','" & ComboBox2.Text & "','" & DateTimePicker1.Text & "','" & TextBox4.Text & "')"

mysql.Open()

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

reader.Close()

mysql.Close()

MsgBox("Product Added Successfully", MsgBoxStyle.Information)

TextBox1.Text = ""

TextBox2.Text = ""

TextBox3.Text = ""

ComboBox1.Text = ""

ComboBox2.Text = ""

TextBox4.Text = ""

Me.Show()

TextBox4.Focus()

End If

Catch ex As Exception

If MsgBox("Check your product code", vbOKOnly) Then

TextBox4.Focus()

End If

End Try

End Sub

Private Sub addproduct\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Me.Show()

TextBox4.Focus()

End Sub

Private Sub TextBox4\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox4.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

End If

End If

End Sub

Private Sub TextBox2\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox2.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

End If

End If

End Sub

Private Sub TextBox3\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox3.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

End If

End If

End Sub

Private Sub TextBox1\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress

If Not (Asc(e.KeyChar) = 8) Then

If Not ((Asc(e.KeyChar) >= 97 And Asc(e.KeyChar) <= 122) Or (Asc(e.KeyChar) >= 65 And Asc(e.KeyChar) <= 90) Or Asc(e.KeyChar) = 32) Then

e.KeyChar = ChrW(0)

e.Handled = True

End If

End If

End Sub

End Class

**-Modify product**

Private Sub modifybtn\_Click(sender As Object, e As EventArgs) Handles modifybtn.Click

Dim mysql As New MySqlConnection

Dim reader As MySqlDataReader

Dim ada As New MySqlDataAdapter

Dim dt As New DataTable

Dim bgs As New BindingSource

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

Dim query1 As String

If TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or TextBox4.Text = "" Or TextBox5.Text = "" Or TextBox6.Text = "" Then

MsgBox("Please enter valid details", MsgBoxStyle.Exclamation)

Else

query = "update addproduct set price='" & TextBox2.Text & "',quantity='" & TextBox3.Text & "',uom='" & TextBox4.Text & "',tax='" & TextBox5.Text & "' where productname='" & TextBox1.Text & "'"

query1 = "select Pcode,productname,price,quantity,uom,tax,dates from billing.addproduct"

mysql.Open()

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

reader.Close()

If MsgBox("Updated successfully", vbOKOnly) Then

TextBox1.Text = ""

TextBox2.Text = ""

TextBox3.Text = ""

TextBox4.Text = ""

TextBox5.Text = ""

TextBox6.Text = ""

TextBox7.Text = ""

cmd = New MySqlCommand(query1, mysql)

ada.SelectCommand = cmd

ada.Fill(dt)

bgs.DataSource = dt

DataGridView1.DataSource = bgs

ada.Update(dt)

mysql.Close()

End If

End If

Catch ex As Exception

End Try

End Sub

**-Delete product:**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class stockdeletefrm

Dim msyql As MySqlConnection

Dim cmd As MySqlCommand

Private Sub productlist\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Dim query As String

Dim mysql As New MySqlConnection

Dim ada As New MySqlDataAdapter

Dim dt As New DataTable

Dim bgs As New BindingSource

TextBox1.Enabled = False

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

mysql.Open()

query = "select Pcode,productname,uom,quantity,price,tax,dates from billing.addproduct"

cmd = New MySqlCommand(query, mysql)

ada.SelectCommand = cmd

ada.Fill(dt)

bgs.DataSource = dt

DataGridView1.DataSource = bgs

ada.Update(dt)

mysql.Close()

Catch ex As Exception

MsgBox("You have not saved any product")

End Try

End Sub

Private Sub DataGridView1\_CellContentClick(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick

Dim i As Integer

i = DataGridView1.CurrentCell.RowIndex

TextBox1.Text = DataGridView1.Item(1, i).Value

Button1.Focus()

End Sub

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim query As String

Dim mysql As New MySqlConnection

Dim reader As MySqlDataReader

Dim query1 As String

Dim ada As New MySqlDataAdapter

Dim dt As New DataTable

Dim bgs As New BindingSource

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

If TextBox1.Text = "" Then

MsgBox("Select the product to be removed", vbInformation, vbOKOnly)

TextBox1.Focus()

Else

mysql.Open()

query = "delete from addproduct where productname='" & TextBox1.Text & "'"

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

reader.Close()

query1 = "select Pcode,productname,price,quantity,uom,tax,dates from billing.addproduct"

cmd = New MySqlCommand(query1, mysql)

ada.SelectCommand = cmd

ada.Fill(dt)

bgs.DataSource = dt

DataGridView1.DataSource = bgs

ada.Update(dt)

mysql.Close()

MsgBox("Product Deleted Successfully", MsgBoxStyle.Information)

TextBox1.Text = ""

TextBox1.Focus()

End If

End Sub

End Class

**For Registration:**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class Registerationfrm

Dim mysql As MySqlConnection

Dim cmd As MySqlCommand

Private Sub nextbtn\_Click(sender As Object, e As EventArgs) Handles nextbtn.Click

Dim reader As MySqlDataReader

Dim mysql As New MySqlConnection

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

If TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or TextBox4.Text = "" Or TextBox5.Text = "" Or ComboBox1.Text = "" Or ComboBox2.Text = "" Or TextBox6.Text = "" Or TextBox7.Text = "" Then

MsgBox("Please enter all the details", vbOKOnly)

TextBox1.Focus()

Else

mysql.Open()

query = "insert into registeration\_form(shop\_name,gstin,tin\_no,address,phone,state,city,pincode,email) values('" & TextBox1.Text & "','" & TextBox2.Text & "','" & TextBox3.Text & "','" & TextBox4.Text & "','" & TextBox5.Text & "','" & ComboBox1.Text & "','" & ComboBox2.Text & "','" & TextBox6.Text & "','" & TextBox7.Text & "')"

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

mysql.Close()

Owner\_details.Show()

Owner\_details.TextBox1.Focus()

Me.TopMost = False

Me.Close()

End If

Catch ex As Exception

MsgBox(ex.Message)

Finally

mysql.Dispose()

End Try

End Sub

Private Sub TextBox2\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox2.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

Else

e.Handled = False

End If

End If

End Sub

Private Sub TextBox3\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox3.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

Else

e.Handled = False

End If

End If

End Sub

Private Sub TextBox6\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox6.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

End If

End If

End Sub

Private Sub TextBox5\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox5.KeyPress

If Asc(e.KeyChar) <> 8 Then

If Asc(e.KeyChar) < 48 Or Asc(e.KeyChar) > 57 Then

e.Handled = True

End If

End If

End Sub

Private Sub TextBox1\_gotfocus(sender As Object, e As EventArgs) Handles TextBox1.GotFocus

nextbtn.Enabled = False

End Sub

Private Sub TextBox7\_GotFocus(sender As Object, e As EventArgs) Handles TextBox7.GotFocus

nextbtn.Enabled = True

End Sub

Private Sub TextBox1\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress

If Not (Asc(e.KeyChar) = 8) Then

If Not ((Asc(e.KeyChar) >= 97 And Asc(e.KeyChar) <= 122) Or (Asc(e.KeyChar) >= 65 And Asc(e.KeyChar) <= 90) Or Asc(e.KeyChar) = 32) Then

e.KeyChar = ChrW(0)

e.Handled = True

End If

End If

End Sub

End Class

**For Owner Details:**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class Owner\_details

Dim mysql As MySqlConnection

Dim cmd As MySqlCommand

Private Sub savebtn\_Click(sender As Object, e As EventArgs) Handles savebtn.Click

Dim mysql As New MySqlConnection

Dim query As String

Dim reader As MySqlDataReader

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

query = "insert into owner\_details(Name,Password,Securityquestion1,Ans1,Securityquestion2,Ans2) values('" & TextBox1.Text & "','" & TextBox2.Text & "','" & ComboBox1.Text & "','" & TextBox3.Text & "','" & ComboBox2.Text & "','" & TextBox4.Text & "')"

If TextBox1.Text = "" Or TextBox2.Text = "" Or ComboBox1.Text = "" Or TextBox3.Text = "" Or ComboBox2.Text = "" Or TextBox4.Text = "" Then

MsgBox("Enter all the details", vbOKOnly)

TextBox1.Focus()

Else

mysql.Open()

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

MsgBox("data saved")

mysql.Close()

Me.Close()

Main\_page.newbillbtn.Focus()

End If

End Sub

Private Sub backbtn\_Click(sender As Object, e As EventArgs) Handles backbtn.Click

Registerationfrm.Show()

Me.Hide()

End Sub

Private Sub TextBox1\_KeyPress(sender As Object, e As KeyPressEventArgs) Handles TextBox1.KeyPress

If Not (Asc(e.KeyChar) = 8) Then

If Not ((Asc(e.KeyChar) >= 97 And Asc(e.KeyChar) <= 122) Or (Asc(e.KeyChar) >= 65 And Asc(e.KeyChar) <= 90) Or Asc(e.KeyChar) = 32) Then

e.KeyChar = ChrW(0)

e.Handled = True

End If

End If

End Sub

End Class

**For Reports:**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class Reportfrm

Dim mysql As MySqlConnection

Dim cmd As MySqlCommand

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim mysql As New MySqlConnection

Dim ada As New MySqlDataAdapter

Dim bgs As New BindingSource

Dim dt As New DataTable

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

mysql.Open()

query = "select dates,productname,price,quantity,uom,gst,subtotal from orderform where dates between '" & DateTimePickerFrom.Text & "' and '" & DateTimePickerTo.Text & "'"

cmd = New MySqlCommand(query, mysql)

ada.SelectCommand = cmd

ada.Fill(dt)

bgs.DataSource = dt

DataGridView1.DataSource = bgs

ada.Update(dt)

mysql.Close()

Catch ex As Exception

MsgBox("Nothing to display", vbInformation, vbOKOnly)

End Try

End Sub

End Class

**For Bill Copy**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class Bill\_Copy

Dim msyql As MySqlConnection

Dim cmd As MySqlCommand

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim mysql As New MySqlConnection

Dim query As String

Dim ada As New MySqlDataAdapter

Dim bgs As New BindingSource

Dim dt As New DataTable

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

query = "select dates,productname,quantity,uom,price,gst,subtotal from orderform where billno='" & TextBox1.Text & "'"

mysql.Open()

cmd = New MySqlCommand(query, mysql)

ada.SelectCommand = cmd

ada.Fill(dt)

bgs.DataSource = dt

DataGridView1.DataSource = bgs

ada.Update(dt)

mysql.Close()

End Sub

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

Printing\_form.Show()

End Sub

End Class

**For Change Password**

Imports MySql.Data.MySqlClient

Imports System.Data

Public Class changepasswordfrm

Dim mysql As MySqlConnection

Dim cmd As MySqlCommand

Dim ada As MySqlDataAdapter

Dim ds As DataSet

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim query As String

Dim mysql As New MySqlConnection

Dim reader As MySqlDataReader

Dim query1 As String

Dim count As Integer

Dim count1 As Integer

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

mysql.Open()

If RadioButton1.Checked Then

query = "select \* from owner\_details where Securityquestion1='" & TextBox1.Text & "' and Ans1='" & TextBox2.Text & "'"

cmd = New MySqlCommand(query, mysql)

reader = cmd.ExecuteReader

count = 0

While reader.Read

count = count + 1

End While

If count = 1 Then

MsgBox("Password is correct", vbOKOnly)

New\_password.Show()

New\_password.TextBox1.Focus()

Me.Close()

Else

MsgBox("Incorrect Password", vbOKOnly)

TextBox2.Focus()

End If

ElseIf RadioButton2.Checked Then

query1 = "select \* from owner\_details where Securityquestion2='" & TextBox1.Text & "' and Ans2='" & TextBox2.Text & "'"

cmd = New MySqlCommand(query1, mysql)

reader = cmd.ExecuteReader

count1 = 0

While reader.Read

count1 = count1 + 1

End While

If count1 = 1 Then

MsgBox("Password is correct", vbOKOnly)

New\_password.Show()

New\_password.TextBox1.Focus()

Me.close()

Else

MsgBox("Incorrect Password ", vbOKOnly)

TextBox2.Focus()

End If

Else : MsgBox("Selct the option please", vbInformation, vbOK)

End If

mysql.Close()

End Sub

Private Sub RadioButton1\_click(sender As Object, e As EventArgs) Handles RadioButton1.Click

Dim ds As New DataSet

Dim dt As New DataTable

Dim mysql As New MySqlConnection

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

Dim query As String

mysql.Open()

query = "select Securityquestion1 from Owner\_details"

ada = New MySqlDataAdapter(query, mysql)

ada.Fill(ds)

TextBox1.Text = ds.Tables(0).Rows(0).Item(0).ToString()

TextBox2.Text = ""

TextBox2.Focus()

mysql.Close()

Catch ex As Exception

MsgBox("You have not entered Security question", MsgBoxStyle.Critical)

End Try

End Sub

Private Sub RadioButton2\_Click(sender As Object, e As EventArgs) Handles RadioButton2.Click

Dim ds As New DataSet

Dim dt As New DataTable

Dim mysql As New MySqlConnection

Dim query As String

mysql.ConnectionString = "server=localhost;user id=root;password=root;database=billing"

Try

query = "select Securityquestion2 from Owner\_details"

TextBox1.Text = ""

mysql.Open()

TextBox1.Text = ""

ada = New MySqlDataAdapter(query, mysql)

ada.Fill(ds)

TextBox1.Text = ds.Tables(0).Rows(0).Item(0).ToString()

TextBox2.Text = ""

TextBox2.Focus()

mysql.Close()

Catch ex As Exception

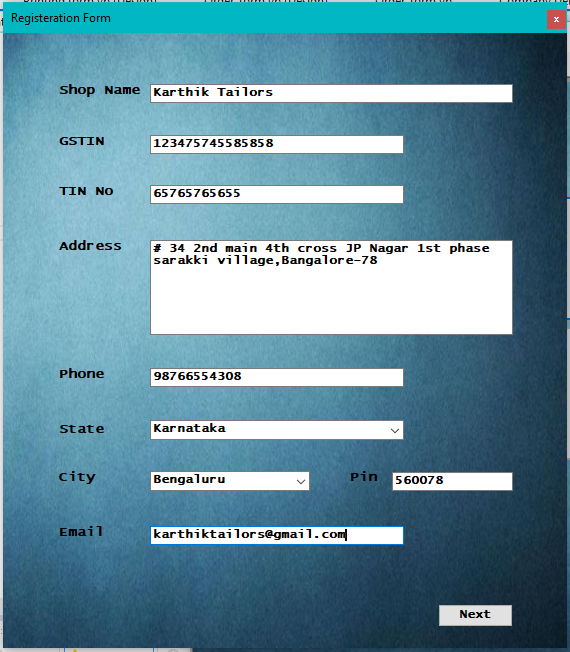
MsgBox("You have not entered Security question", MsgBoxStyle.Critical)

End Try

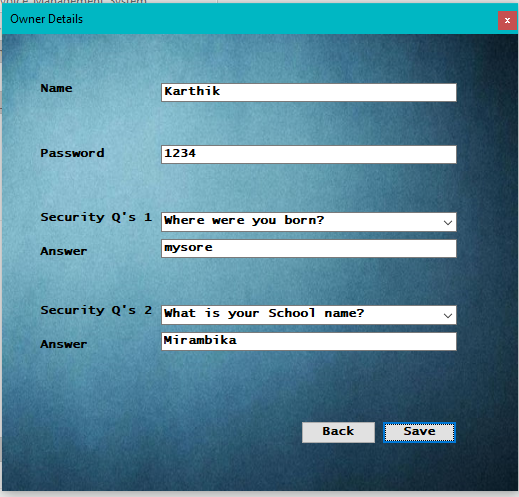
End Sub

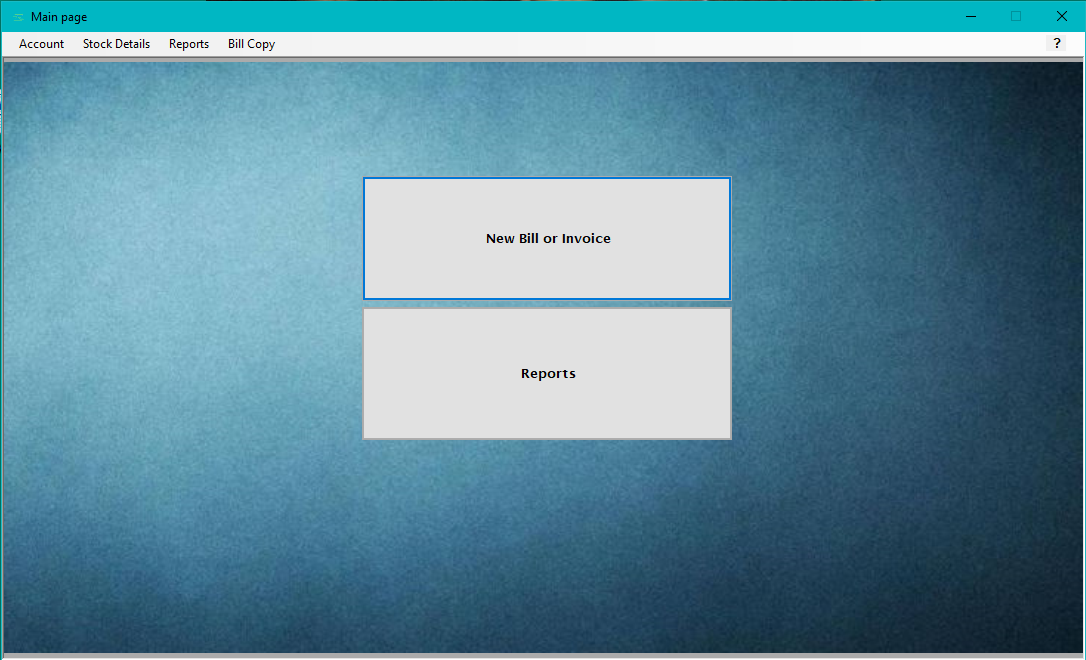
End Class

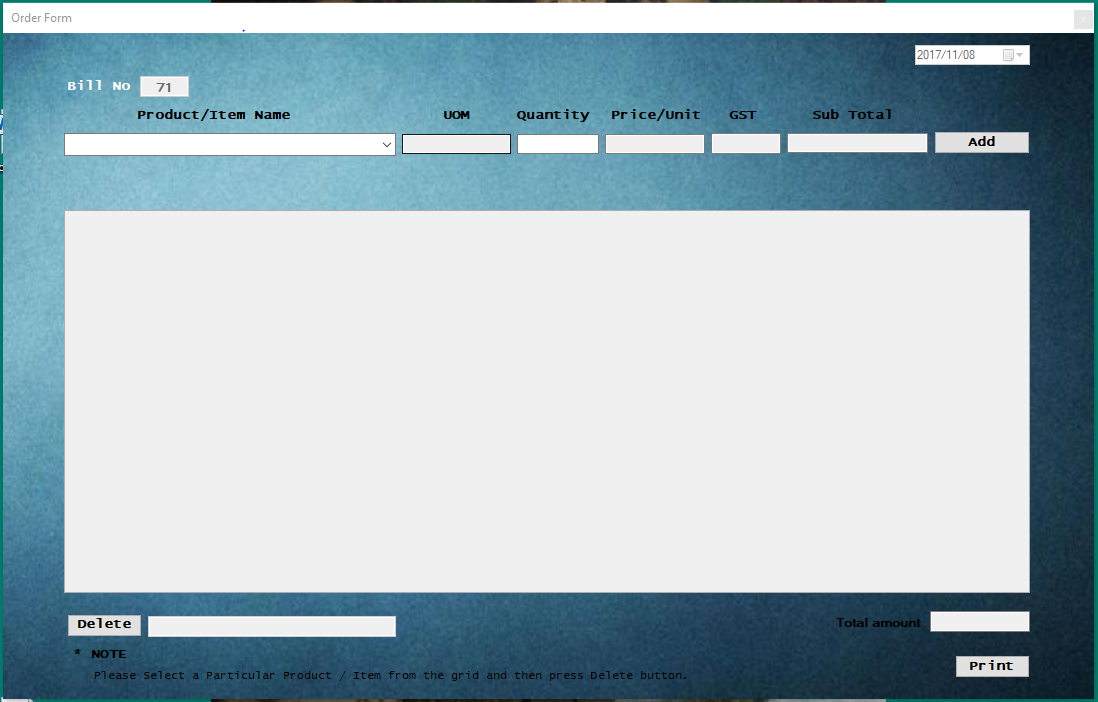
**SNAPSHOTS**

**Registration form**

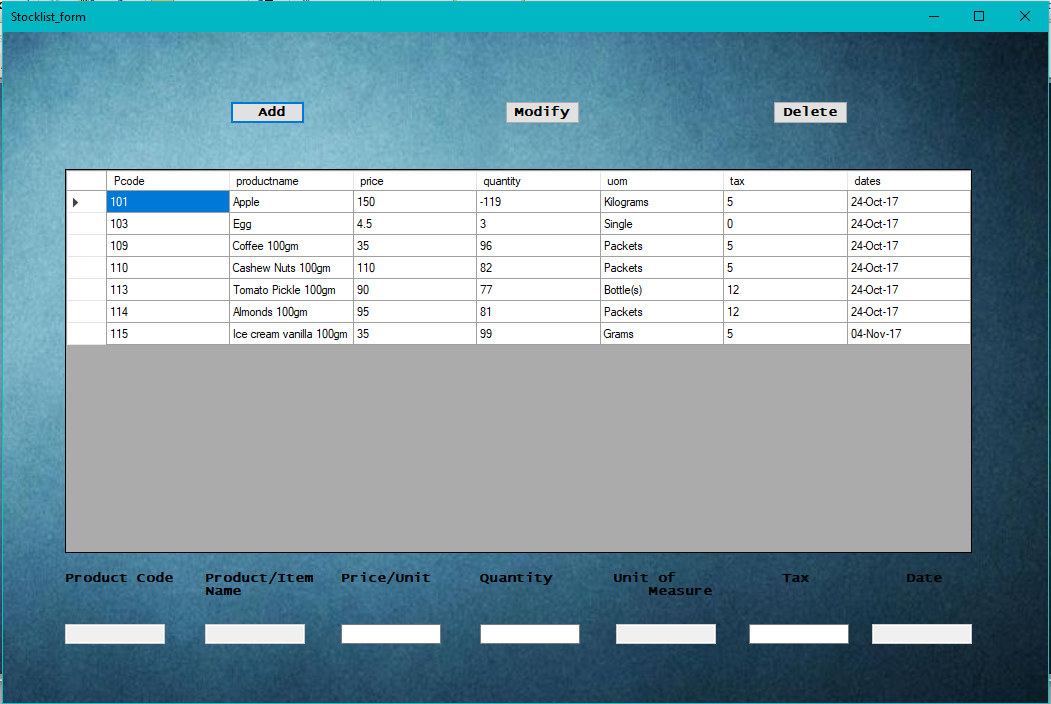
**Owner Details**

****

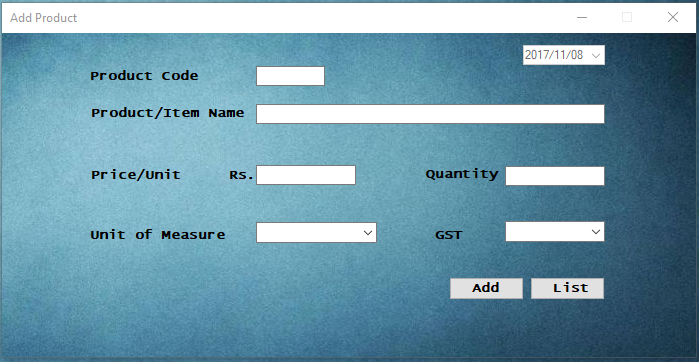
**Main Page**

**New Bill/Invoice**

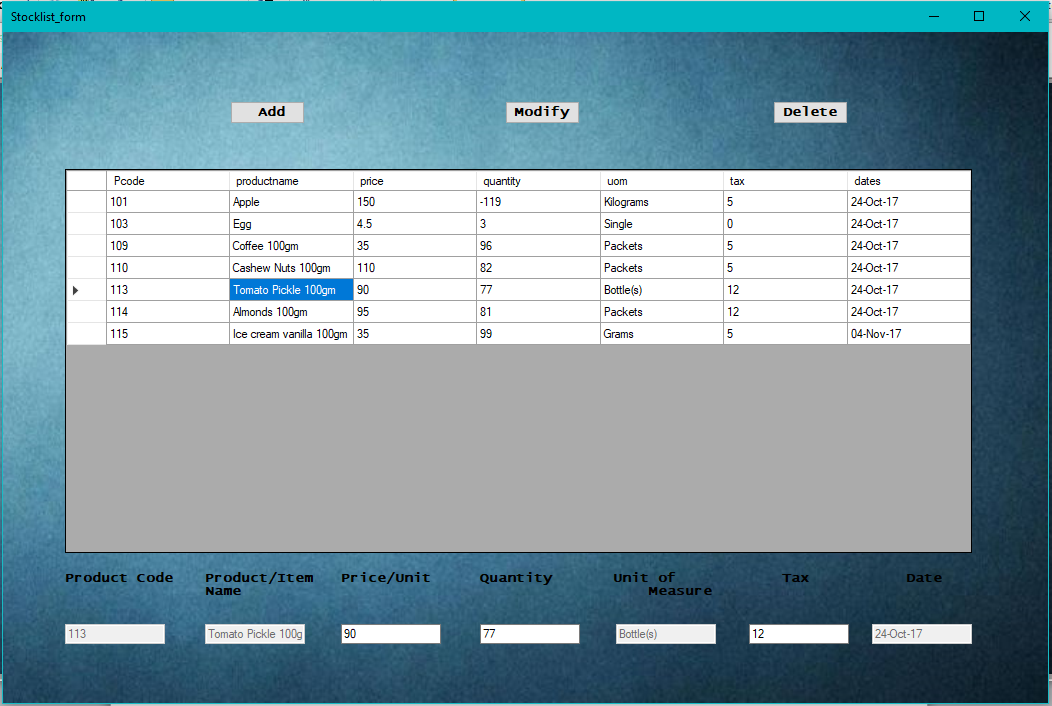
**Stock List**

****

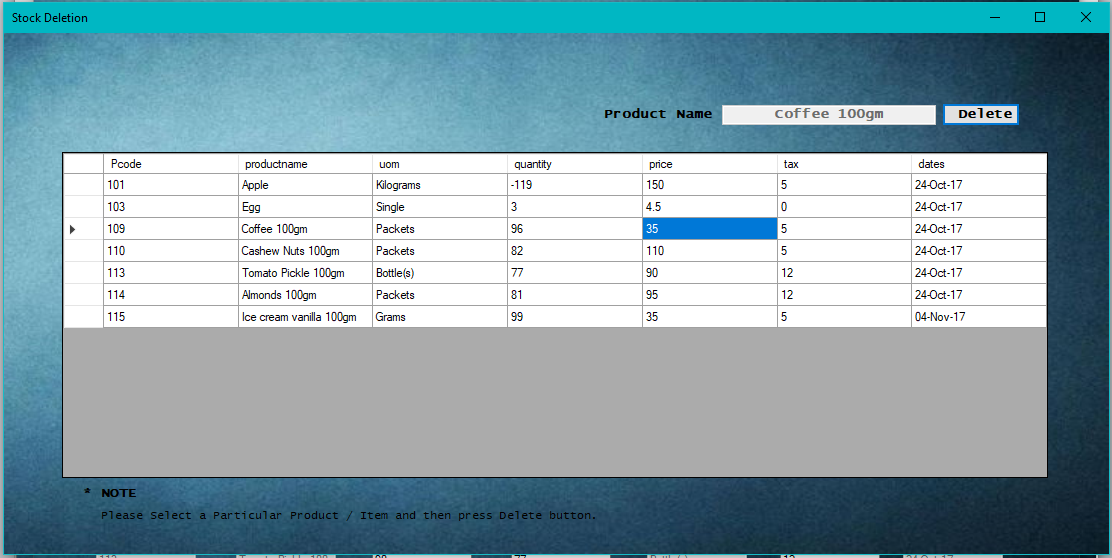
**Add Product**

****

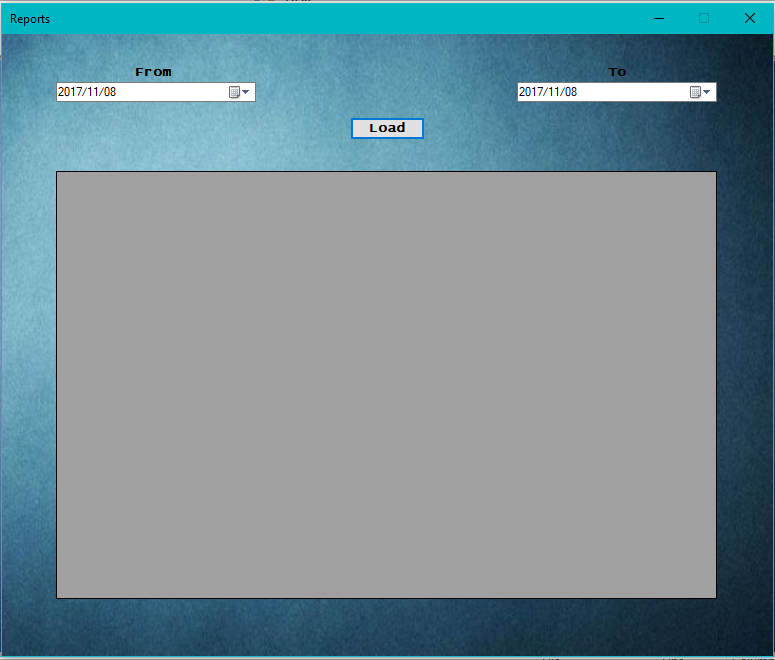
**Modify product**

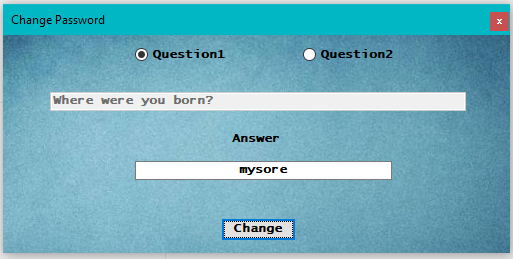
****

**Delete Product**

****

**Reports**

****

**Change password**

**New password**

****