

1.INTRODUCTION

The project is on GLOBAL MARKET BILLING SYSTEM. It is the place where customers come to purchase their daily using products and pay for that. So there is a need to calculate how many products are sold and to generate the bill for the customer.

Global market billing system is a windows application designed to help staff or admin to maintain and organized global market system. This application is easy to use for both beginners and advanced users. It features a familiar and well thought-out, an attractive user interface, combined with strong searching insertion and deletion capabilities. It can also calculate GST amount on every product. GST can be issued according to government specified categories of all the products. Total sales can also be Maintain in this project.

The Data will be stored in the Database. Because of this software, paper work will be reduced and the user can spend more time on the monitoring the super market. The project will be user friendly and easy to use.

1.1 ABOUT THE PROJECT

The project global market billing system is developed on visual basic language which mainly focuses on basic operation in a global market like-adding new employee, adding new products, updating information of employee, searching products and employees and facility to generate bills of customers and calculate total sales amount of the global market.

This application is helpful for staff or admins for billing of products and managing products details. This application access by only staff and admin. Members can't access it directly.

OBJECTIVE:

The main objective of this project is to provide a friendly environment to maintain the details of products and employees and generate bill for customers. And replace the manually work. And make software fast in processing, with good user interface so that user can change it and it should be used for a long time without error and maintenance.

1.2 ABSTRACT

Our project "Global Market Billing System" includes registration of customers, suppliers, products, employee and their storage details. Our software has the facility to give a unique id for every customer, supplier and staff automatically.

- It includes a search facility to know the current status of each product.
- The user can search availability of product and their details using the id.

It has main modules like:

- Loading form
- Welcome form
- Admin login form
- MDI form
- Employee details form
- View product form
- Customer details form
- Product details form
- View employee form
- Add new customer form
- Add new employee form
- Bill generation form
- View customer form
- Add new product form

2. SYSTEM ANALYSIS

2.1 EXISTING SYSTEMS

The major drawback of the existing system is that it is very different to receive data. It is difficult to handle the whole system manually and is less accurate to keep the data for future references because it may get destroyed. Moreover, redundancy of data may occur and this may lead to the inconsistency. The manual system is so time consuming.

Features of Existing System are:

- Lack of immediate information storage
- Lack of immediate retrievals
- Error prone manual calculations
- Lack of prompt updating
- Preparation of accurate prompt report

NEED:

- Efficiently maintain the details about the products
- Simultaneously updates changes made to any data item in the entire database
- It should be faster than manual system

2.2 PROPOSED SYSTEM

The proposed system is very easy to operate; speed and accuracy are the main advantages of proposed system. There is no redundancy of data are stored in the computer's secondary memories like hard disk, etc. It can be easily received and used at any time. The proposed system will easily handle all the data and the work done by the existing systems. The proposed systems eliminate the drawbacks of the existing system to a great extent and it provides tight security to data.

FEATURES OF PROPOSED SYSTEM:

- Reduced man power
- Easy retrieval of data
- Consumers less time
- All records can be updated

2.3 FEASIBILITY STUDY

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the feature of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability

to meet their user needs and effective use of resources. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development. The document provides the feasibility of the project that is being designed and lists various area that were considered very carefully during the feasibility study of this project such as Technical, Economic and Operational feasibilities. The following is its feature:

• TECHNICAL FEASIBILITY

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified on outline system, the investigation must go on to suggest the type of equipment, required method in developing the system, and running the system once it has been designed. Technical issues raised during the investigation are:

- Does the existing technology sufficient for the suggested one?
- Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within latest technology. Though the technology may become obsolete after some period of time, due to the fact that newer version of same software supports older versions, the system may still be used. So, there are minimal constraints involved with this project. The system has been developed using VB. The project is technically feasible for development.

• ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is Concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require. The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs of fewer costly errors,

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. All the resources are already available, it gives an indication of the system is economically possible for development.

- **BEHAVIOURAL FEASIBILITY**

This includes the following questions:

1. Is there sufficient support for the users?
2. Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

3. DEVELOPMENT ENVIRONMENT

3.1 HARDWARE SPECIFICATION

Processor	: Intel core TM @ 1.70 GHz
System bus	: 64 bits
RAM	: 4.00GB
HDD	: 1TB
Keyboard	: Standard 101/102 keys

3.2 SOFTWARE REQUIREMENT

OS:	Windows 10
Front end:	Microsoft Visual basic 6.0
Back end:	Microsoft access 2007

3.3 OVER VIEW OF VB AND MICROSOFT ACCESS

MICROSOFT VISUAL BASIC 6.0

Visual Basic 6.0 is Microsoft's latest version of the Visual Basic Programming language. Although writing can be a tedious chore at time. Visual Basic reduce the effort required your part, and makes programming enjoyable.

Visual Basic makes aspects of programming as simple as dragging graphics objects on to screen with your mouse.

Visual Basic 6.0 more than just a programming language; the secret to Visual Basic is in its name "Visual". With two days Windows Operating System, a program must be able to interact with the screen. Keyboard, mouse and printer graphically.

The environment provided by Visual Basic is suitable for any type of application. Using this environment, the user can visually design the object that your application uses. Visual Basic is not just languages, it's an integration development environment in which you can develop, run, test and debug your applications.

The impressive array of programming resources provided by Visual Basic enables to create the objects extemporaneously which can range from pop-up menu to a message box.

A form is a major part of Visual Basic application, which allows the user to enter the data as well as view the result. A control is an object that we draw on a form to enable or enhance user interaction with an application.

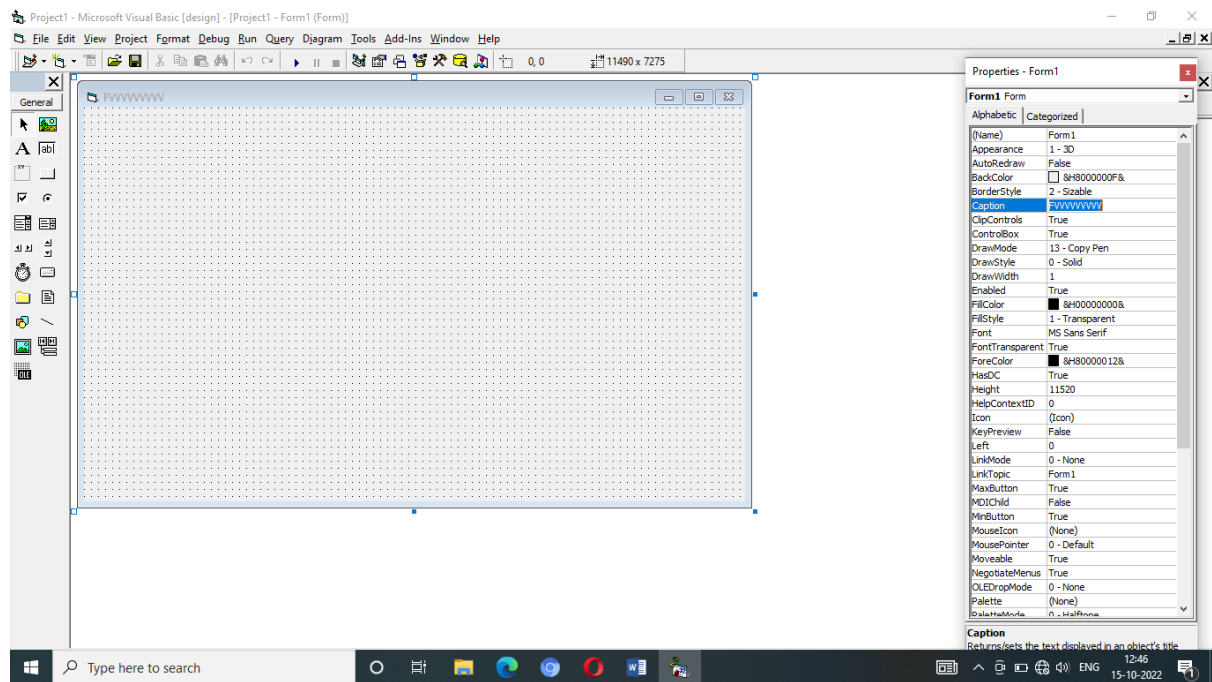
Hence a visual Basic application is a combination of object like forms and controls, procedures that can respond to Event and other general-purpose procedures.

Starting Visual Basic 6.0:

Click on the Start button on the Window task bar.

Select Programs, then Microsoft Visual Basic 6

Click on Visual Basic 6.0



At the top of the screen is the Visual Basic Main Window. At the top of the main window is the Title Bar. The title bar gives us information about what program we're using and what Visual Basic program we are working with. Under the main menu is the Toolbar. Here Title button with pictures also allows us to control Visual Basic, much like the 'tool tip' will pop up and tell you what that particular button does.

Parts of a Visual Basic Project:

There are three major components in a Visual Basic project; the project itself, the form, and the controls. Project are application or program.

- **FROM**

The form is the most important unit in Visual Basic. It is like a canvas on which the user placed certain objects to develop different interfaces. The form can be of any size and color and the user can attach a code to it which will run when the form is loaded for the first time. A simple project may use only one form while certain application may represent different operations and outputting results of different operations. Each form is saved on the disk as a separate file with a .FRM extension.

- **CONTROLS:**

These represent objects or graphical features drawn on forms to allow the user to interact with the system. The objects can be passed on the form and range from text boxes, labels, scroll bars, command buttons and other menus, to file Management utilities and spread sheet type grids. Each object may have certain code attached to it, which identifies the operation or activity to be performed by the control. Form and controls together are sometimes called as objects.

- **PROPERTIES:**

Every object has certain characteristic associated with it. The characteristics identify shape and appearance of the object. These characteristics are termed as properties. Some important properties include names, captions, size, color, positions and contents. When an object is created Visual Basic automatically applies certain default properties. The user can change the properties of objects at design time or at run time.

- **METHODS:**

These are not found in traditional programming. They represent statements or built in procedures that can be invoked to impart some action to a particular object. For example a method can included to draw an object on the form; another method may print certain greeting on the form and son on. Different types of objects have different set of methods that can be used with them.

- **PROCEDURES:**

All the code required for a particular application is written in either procedure or subroutines. Code is normally related to some object or more accurately to the event of an object. The code is executed when a certain event occurs.

- **MODULES:**

The code that is attached to a form is accessible form anywhere on the form. But a program may be used in many forms. A module generally includes a collection of general procedures, variable declarations and constant definitions used by application. There may be several modules in one program and each module is saved as a separate file.

- **PROJECT:**

The project holds together various forms and modules that make up a program. Its purpose is primarily one of convenience. When you want to start work on a program you will have to open a project file. A project file has an extension of VB.

- **FROM WINDOW:**

The Form Window is central to developing Visual Basic applications. It is where you develop your application.

- **PROPERTIES:**

Like all controls, the form has many properties. Fortunately, we only have to know about of them. The properties we will be concerned with are

Property	Description
Name	Name used to identify form. Three letter prefix for form
Caption	Text that appears in the title bar of form.
Icon	Reference to icon that appears in title bar form.
Left form.	Distance from left side of computer screen to left side of
Top form.	Distance from top side of computer screen to top side of
Width	Width of the form
Height	Height of form.
Back Color	Background color of form.
Border Style	Form can be sizable (can resize using the mouse) or fixed size.

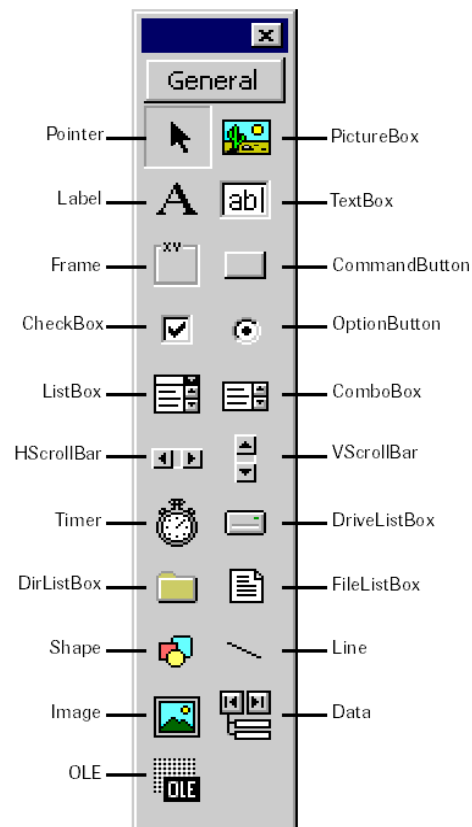
- **FORM EVENTS**

The form primarily acts as ‘container’ for other controls, but it does support events. That is, it can respond to some user interactions. We will only be concerned with two form events in this course:

Events	Description
Click	Event executed when user clicks on the form with the mouse
Load	Event executed when the form first loads the computers

Toolbox Window:

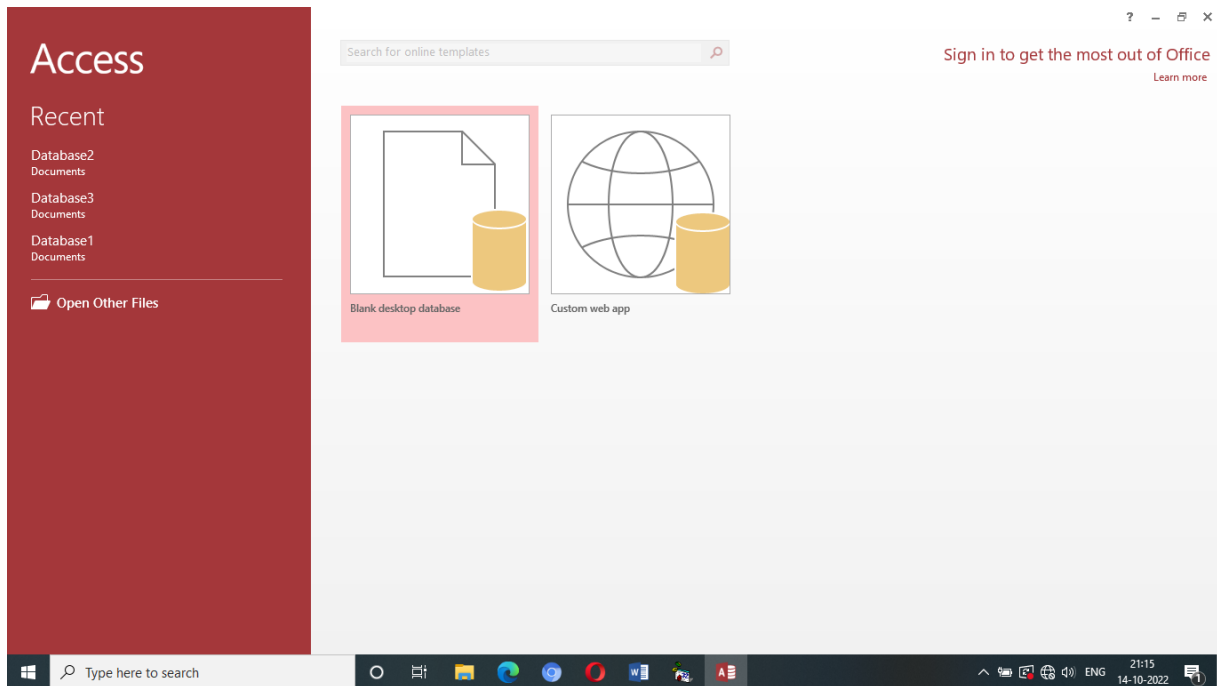
The toolbox window is the selection menu for controls used in your application.



Database Development

MS Access

This activity deals with the design of the physical database. A key is to determine how the access path art to be implemented.



The project is based on the Global Market Billing System, Being the Information System, it requires extensive use of some Database Management System to store, manipulate and handle the huge and complex record.

MS Access – Microsoft Access is an office product that allows the user to create relational databases that have tables of data. MS Access store data in its own format based on the Access Jet Database Engine. It can also import or link directly to data stored in other applications & databases. It's an easy to use tool for quickly creating browser-based database applications.

In every simple term, MS Access is an Information Management tool helps you store information for reference, reporting & analysis. Microsoft Access helps you analyze large amount of information, and manage related data more efficient than MS Excel or other spreadsheet applications.

4.SYSTEM DESIGN

System design is the most creative and challenging phase in the system life cycle. Design the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system design transforms a logic representation of what is required to do into the physical specification. The specification is converted into physical reality during development

4.1 LOGICAL DESIGN

The logical flow of system and define the boundaries of the system. It includes the following steps

- Reviews the current physical system - its data flows, file content, volumes, frequencies etc.,
- Prepares output specification – that is, determines the format, content and frequency of reports.
- Prepares input specification – format, content and most of input functions.
- Prepares edit, security and control specifications.
- Specifies the implementation plan.
- Prepares a logical design walk through of the information flow, output, input, controls and implementation plan.
- Reviews benefits, costs, target date and system constraints.

4.2 PHYSICAL DESIGN

Physical system produces the working systems by define the design specifications that tell the programmers exactly what the candidate system must do. It includes the following steps.

- Design the physical system.
- Specify input and output media.
- Design the database and specify backup procedures.
- Design physical information flow through the system and physical design walk through.
- Plan system implementation.
- Prepare a conversion schedule and target date.
- Determine training procedures, courses and timetable.
- Devise a test and implementation plan and specify any new hardware/software.
- Update benefits, costs and conversion date and system constraints.

Design/specification activities

- Concept formulation.
- Problem understanding.
- High level requirements proposal.

- Feasibility study.
- Requirements engineering.
- Architectural design.

4.3 INPUT DESIGN

Input design deals with what data should be given as input, how the data should be arranged or code, the dialogue to guide the operating personnel in providing input, methods for preparing input validations and steps to follow when error occur. Input design is the process of converting a user – oriented description of the input into a computer – base system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user- friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus, the objective of input design is to create an input layout that is easy to follow.

In this project, the input design consists of log in screen compression / decompression, source and destination browsing button, a menu list for selecting the algorithm, compress / decompress button.

4.4 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. The objective of output design is to convey information about past activities, current status or projections of the future, signal important events, opportunities problems, or warnings, trigger an action etc. efficient, intelligible output design should improve the system's relationship with the user and helps in decisions making. In output design the emphasis is on displaying the output on a CRT screen in a predefined format. The primary consideration in design of output is the information requirements and objectives of the end users. The major formation of the output is to convey the information and so its layout and design the need a careful consideration

4.5 DATA BASE DESIGN

Database name: Empdatabase1

Table name: Employee

Fields:	Name	-	Type
	emp id	-	Integer
	name	-	Text
	dob	-	Text
	address	-	Text
	state	-	Text
	pincode	-	Text
	phoneno	-	Text
	emailed	-	Text
	salary	-	Text
	username	-	Text
	password	-	Text

Indexes (Name – Fields – Unique)

empid - +empid – True

Database name: ProductDetails1

Table name: Productdetails1

Fields:	Name	-	Type
	Barcode ID	-	Long
	Product name	-	Text
	Standard cost	-	Text
	Unit price	-	Text
	Quantity in kg	-	Text
	Stock date	-	Text
	GST	-	Text
	Total	-	Text

Indexes (Name – Fields – Unique)

BarcodeID -+BarcodeID – True77

➤ Customer database

Database name: customer details

Table name: customer

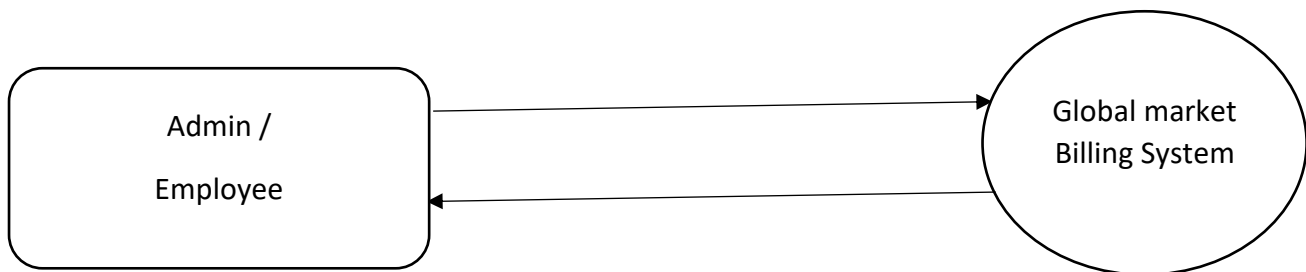
Fields:	Name	-	Type
	First name	-	Text
	Last name	-	Text
	Street address	-	Text
	Street address line	-	Text
	City	-	Text
	State	-	Text
	Pincode	-	Text
	Phone number	-	Text
	Email	-	Text
	Date	-	D/T

Indexes (Name – Fields – Unique)

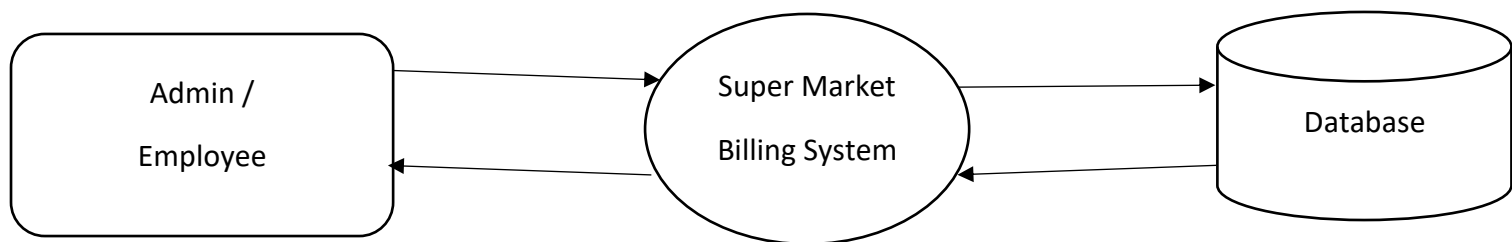
First name - +First name - True

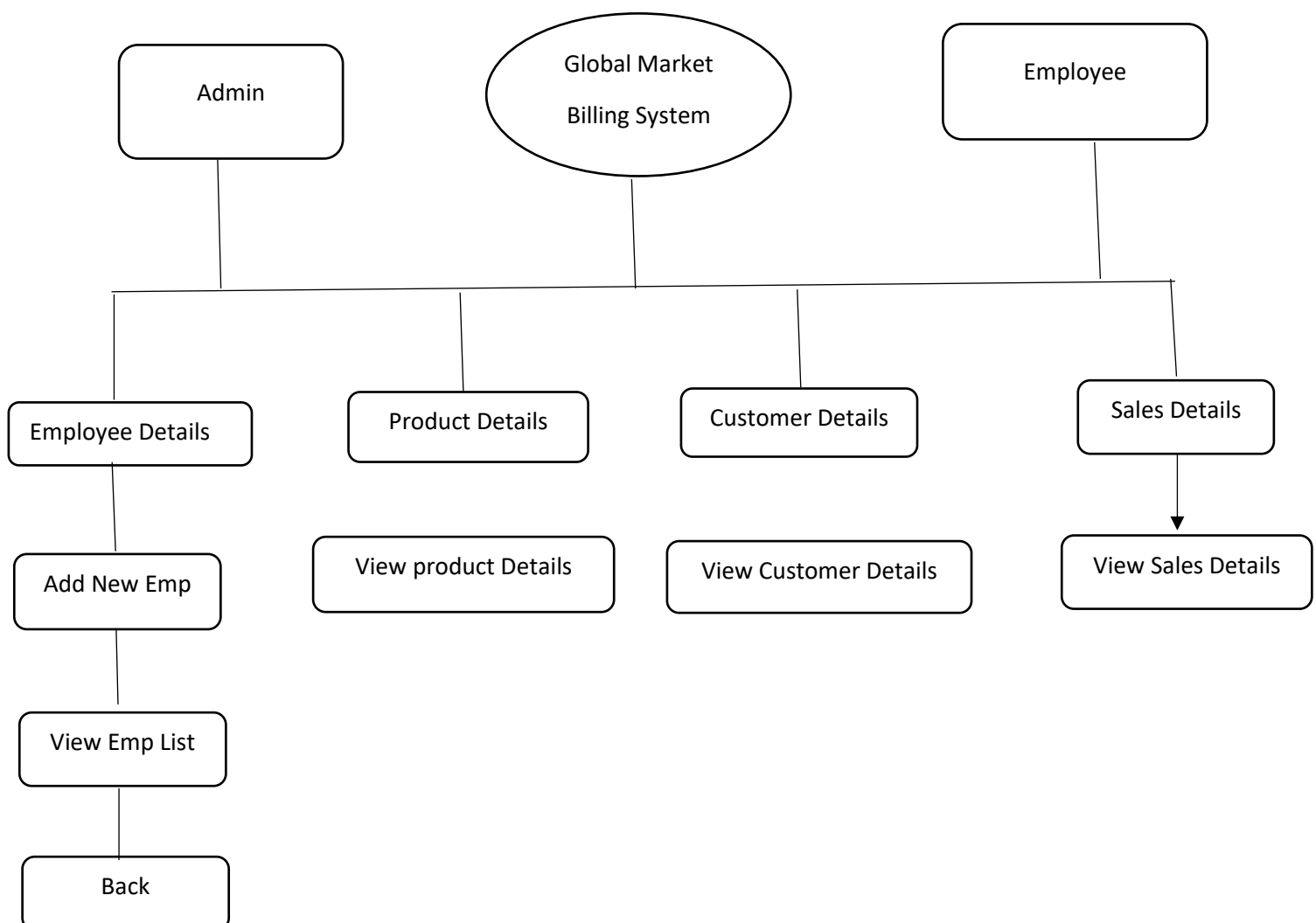
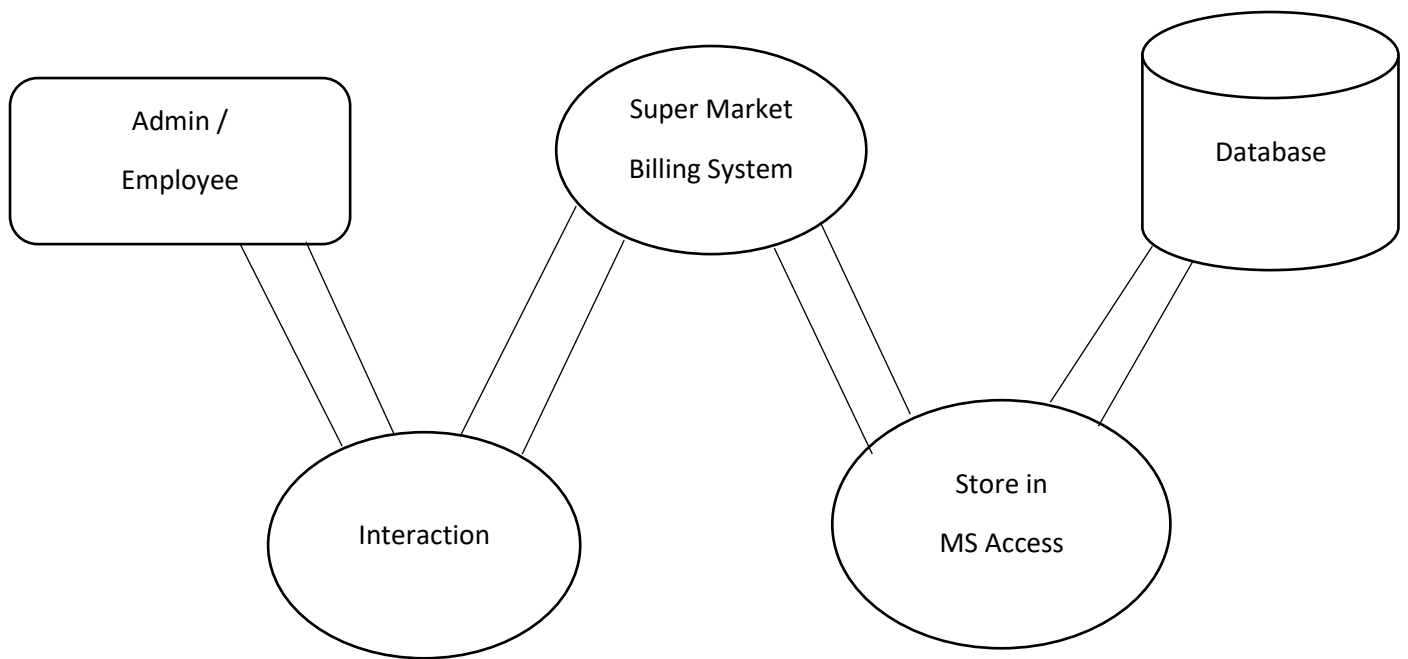
4.6 Data Flow Diagrams Supermarket Billing System:

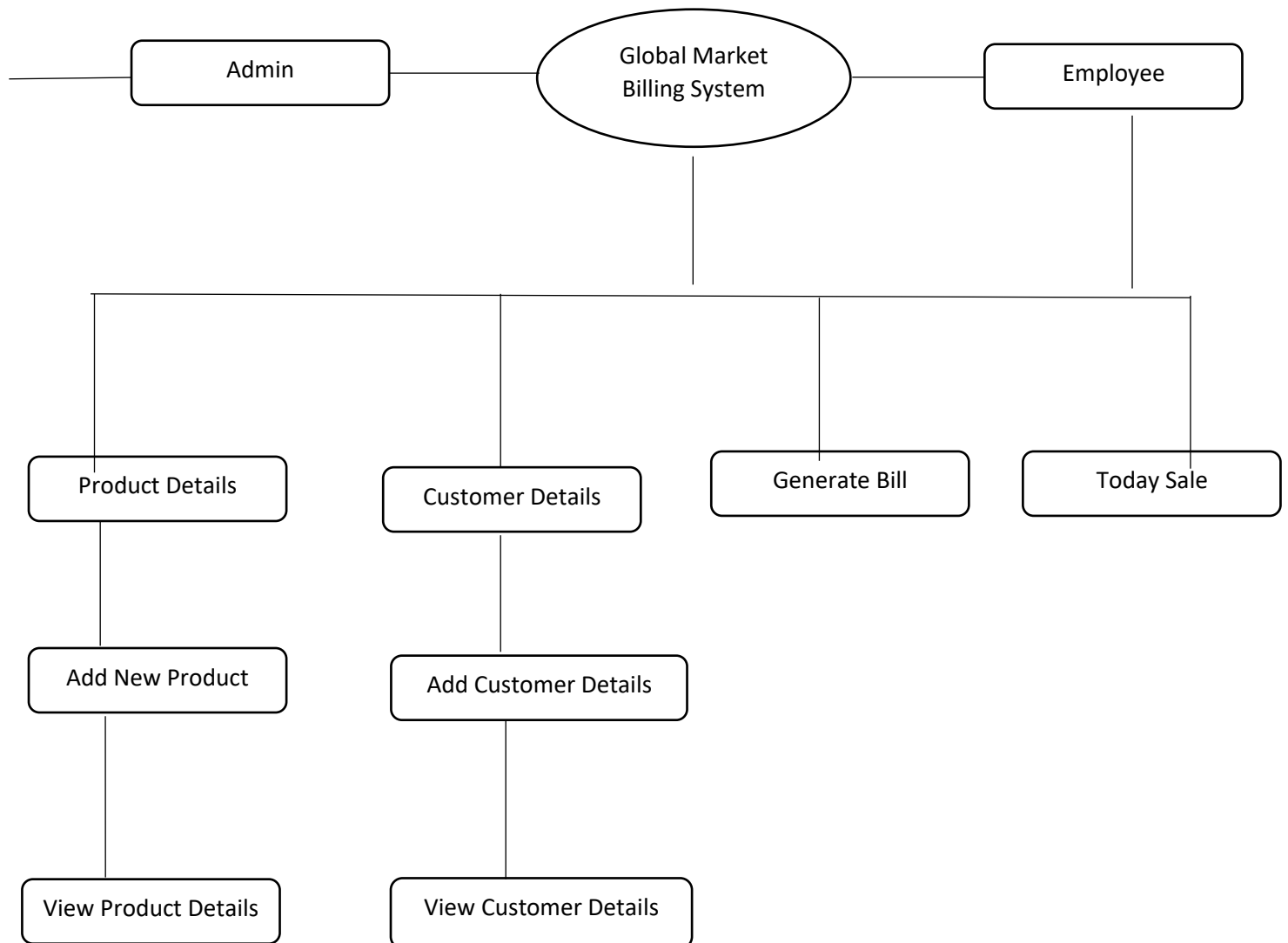
Data Flow Diagram 1



Data Flow Diagram 2

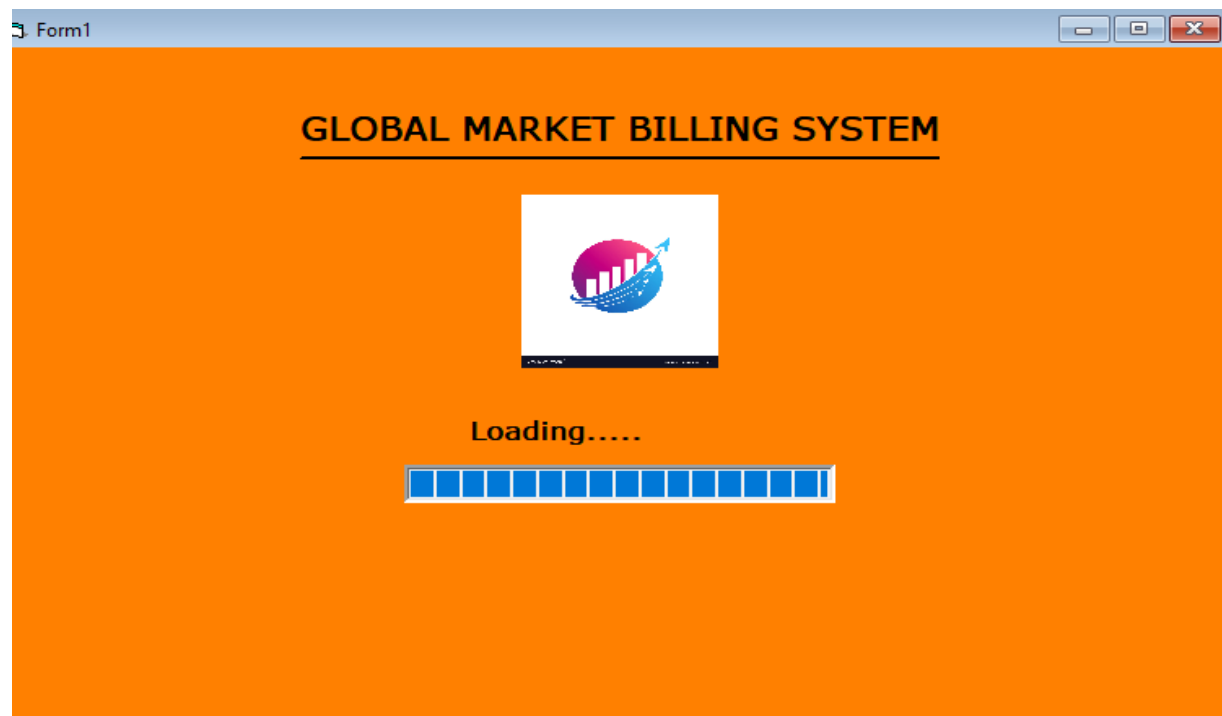


Data Flow diagram 3 and 4

Data Flow Diagram 4

4.7 FORM DESIGN

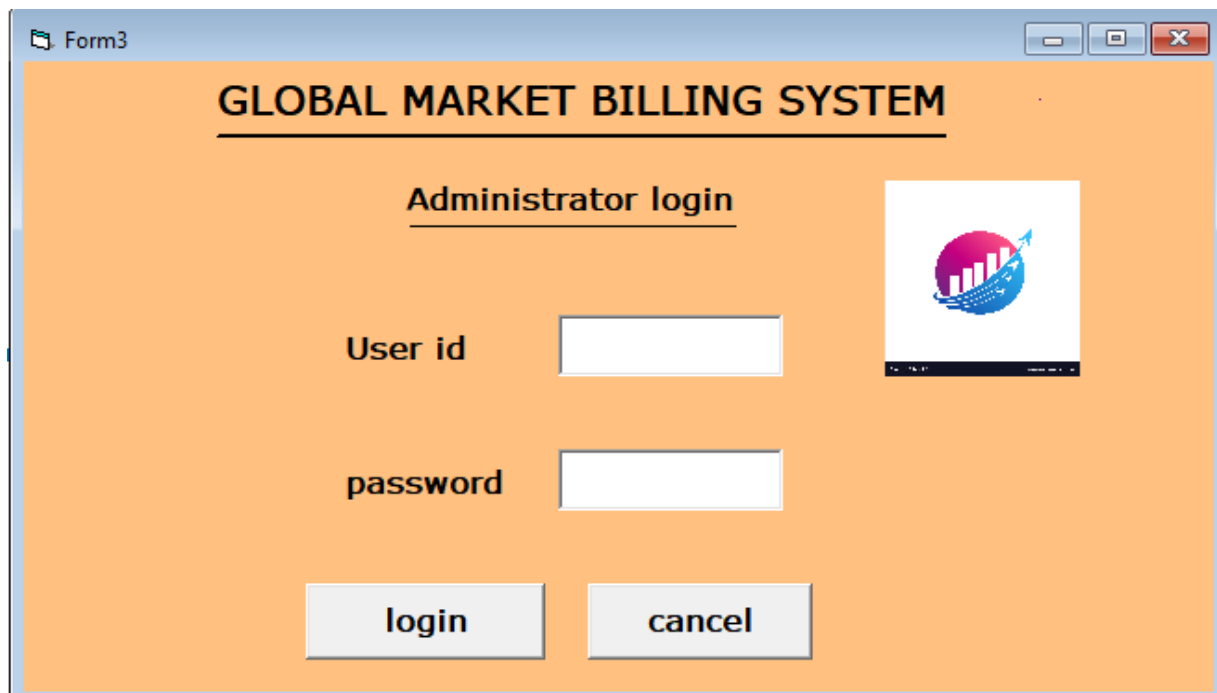
Loading Form



Welcome Form



Admin Login Form



Form3


GLOBAL MARKET BILLING SYSTEM

Administrator login

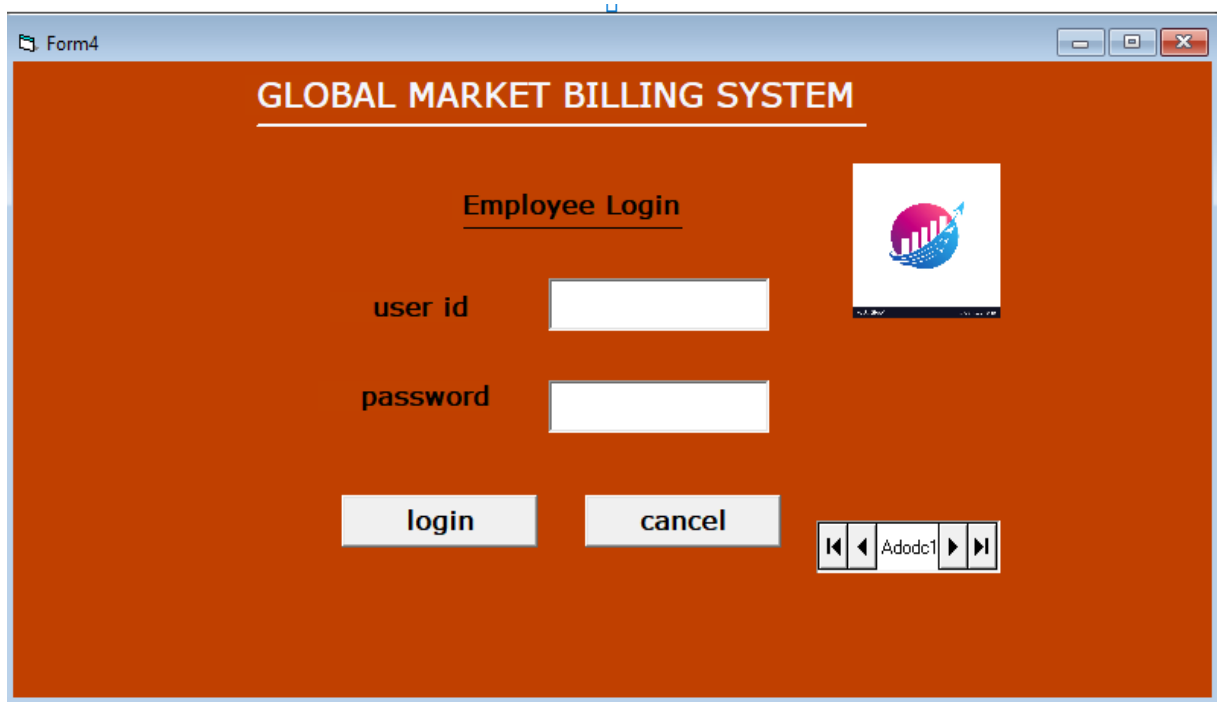
User id

password

login cancel



Employee Login Form



Form4


GLOBAL MARKET BILLING SYSTEM

Employee Login

user id

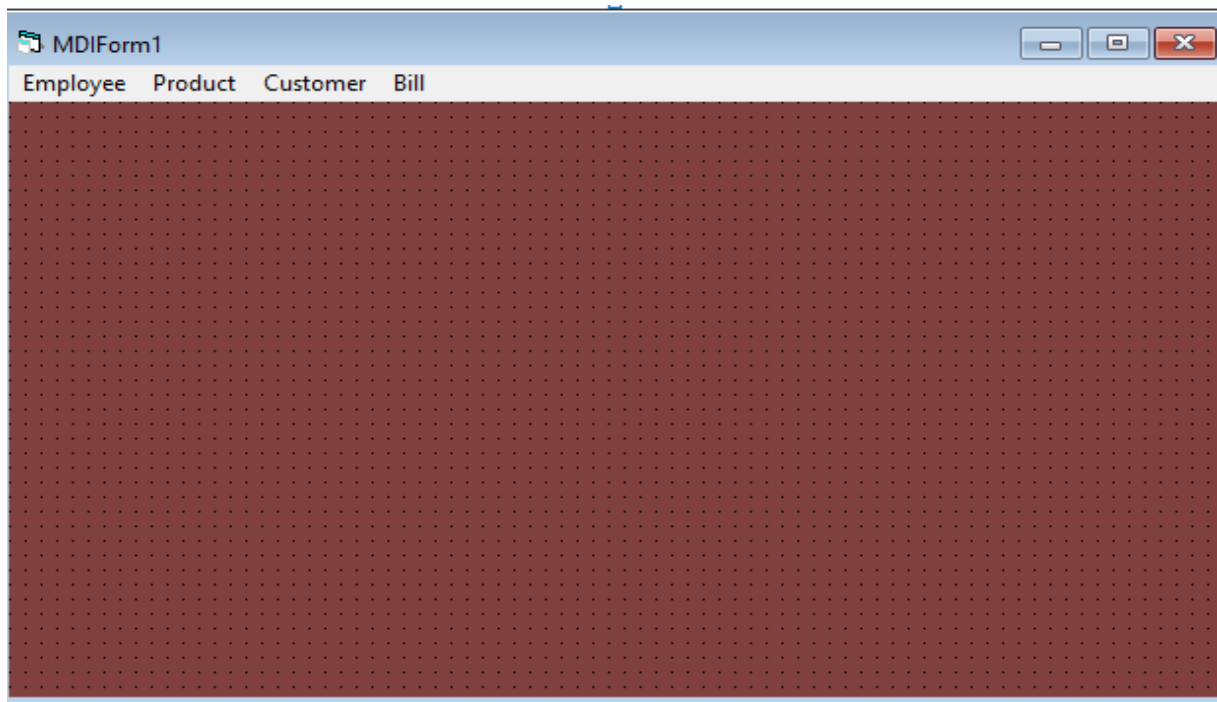
password

login cancel

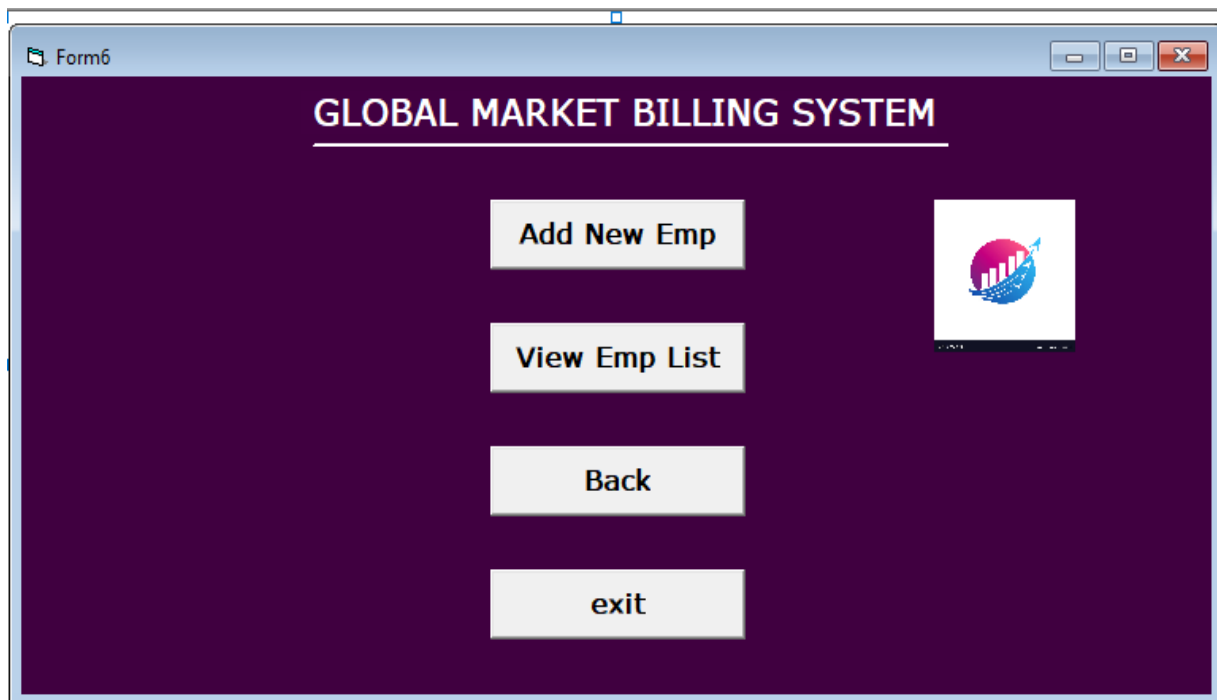


Adodc1

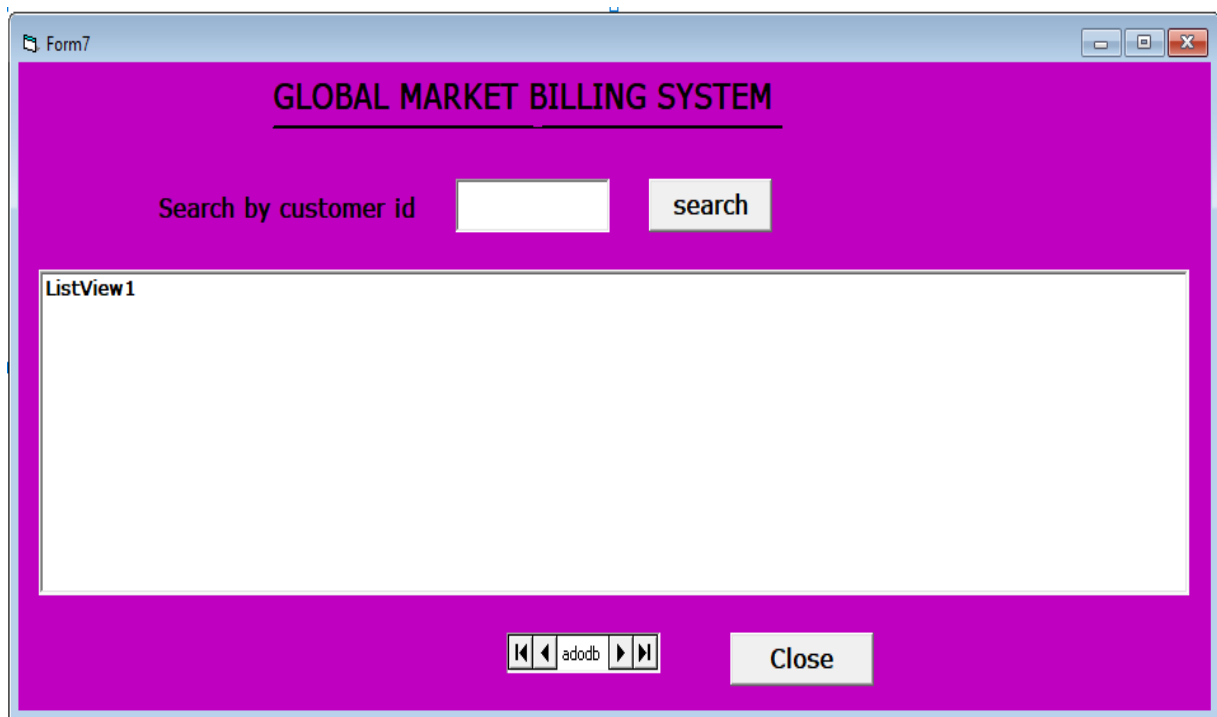
MDI Form



Employee Details Form



View Customer Form



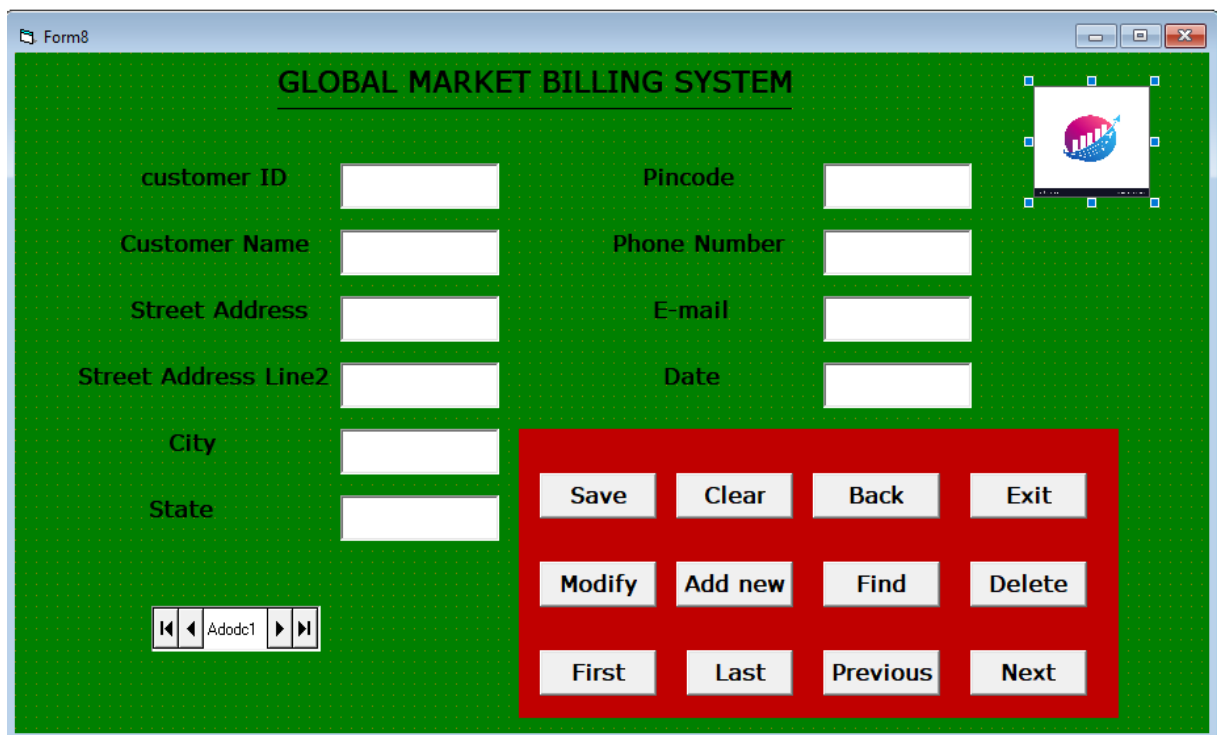
Form7

GLOBAL MARKET BILLING SYSTEM

Search by customer id

ListView1

Customer Details Form



Form8

GLOBAL MARKET BILLING SYSTEM

customer ID Pincode

Customer Name Phone Number

Street Address E-mail

Street Address Line2 Date

City

State

View Product Form

Form10

GLOBAL MARKET BILLING SYSTEM

Search by barcode

ListView1

Adodc1

View Employee Form

Form11

GLOBAL MARKET BILLING SYSTEM

search by employee id

ListView1

Adodc1

Product Details Form



Form12

GLOBAL MARKET BILLING SYSTEM

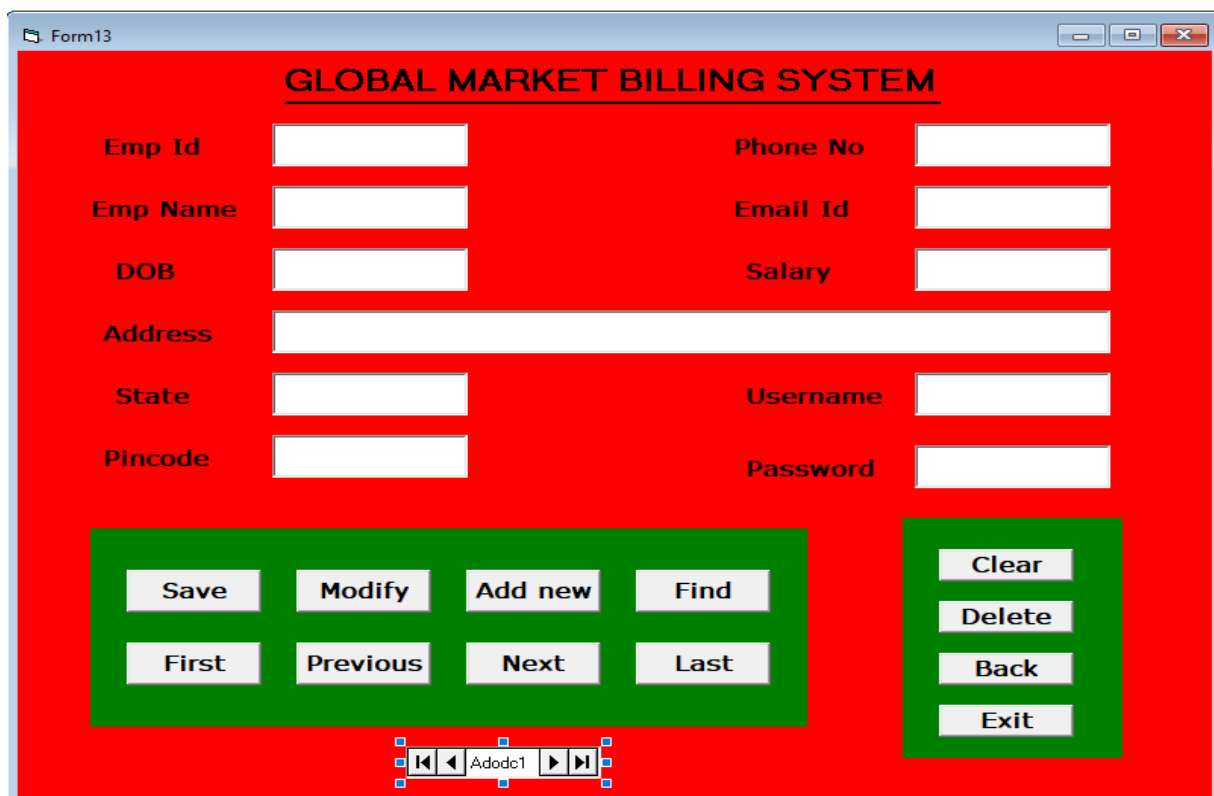
Add new product

View products

Back

Exit

Add New Employee Form



Form13

GLOBAL MARKET BILLING SYSTEM

Emp Id

Emp Name

DOB

Address

State

Pincode

Phone No

Email Id

Salary

Username

Password

Save Modify Add new Find

First Previous Next Last

Clear Delete Back Exit

Bill Generation Form

Form14(bill generation)

GLOBAL MARKET BILLING SYSTEM

<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>
<input type="text"/>	search	<input type="text"/>	<input type="text"/>

total amount

Calculate

Generate Bill

Back

Adodc1

Customer Details Form


form16(customer details)

GLOBAL MARKET BILLING SYSTEM

Add new customer details

View customer list

Back



Add New Product Form

Form17

GLOBAL MARKET BILLING SYSTEM

Product barcode	<input type="text"/>	Quantity	<input type="text"/>
Product name	<input type="text"/>	Current date	<input type="text"/>
Standard cost	<input type="text"/>	Discount in %	<input type="text"/>
GST in %	<input type="text"/>	Total	<input type="text"/>

<input type="button" value="Add new"/>	<input type="button" value="Save"/>	<input type="button" value="Modify"/>	<input type="button" value="Find"/>	<input type="button" value="Clear"/>	<input type="button" value="Delete"/>
<input type="button" value="First"/>	<input type="button" value="Previous"/>	<input type="button" value="Next"/>	<input type="button" value="Last"/>	<input type="button" value="Back"/>	<input type="button" value="Exit"/>

5. SYSTEM IMPLEMENTATION

5.1 CODING

Loading Form

Option Explicit

Private Sub Form_Load()

Timer1.Enabled = True

End Sub

Private Sub Timer1_Timer()

ProgressBar1.Value = ProgressBar1.Value + 5

Label1.Caption = "GLOBAL MARKET BILLING SYSTEM"

Label3.Caption = ProgressBar1.Value & "%"

If (ProgressBar1.Value = ProgressBar1.Max) Then

Timer1.Enabled = False

Unload Me

Form2.Show

End If

End Sub

Welcome Form

Private Sub cmdadminlogin_Click()

Form3.Show

Form2.Hide

End Sub

Private Sub cmdemplogin_Click()

Form4.Show

Form2.Hide

End Sub

Private Sub Form_Load()

Timer1.Interval = 100

End Sub

Private Sub Timer1_Timer()

Label2.Caption

End Sub

Admin Login Form

Dim user As String

Dim pass As String

Private Sub cmdcancel_Click()

Form3.Hide

Form2.Show

End Sub

Private Sub cmdlogin_Click()

user = "Hamsa"

pass = "12345"

If (user = Text1.Text And pass = Text2.Text) Then

MsgBox "login successfully"

Form5.Show

Form3.Hide

Text1.Text = Clear

Text2.Text = Clear

Else

MsgBox "incorrect username and password"

Text1.Text = ""

```
Text2.Text = ""
```

```
Text1.SetFocus
```

```
End If
```

```
End Sub
```

Employee Login Form

```
Dim username As String
```

```
Dim password As Double
```

```
Private Sub cmdcancel_Click()
```

```
Form4.Hide
```

```
Form3.Show
```

```
End Sub
```

```
Private Sub cmdlogin_Click()
```

```
loginado.RecordSource = "select * from employee2 where username='" + txtuser.Text + "' and  
password='" + txtpass.Text + "'"
```

```
loginado.Refresh
```

```
If loginado.Recordset.EOF Then
```

```
MsgBox "invalid username and password,try again....!!!", vbCritical, "please enter correct  
username and password"
```

```
txtuser.Text = ""
```

```
txtpass.Text = ""
```

```
txtuser.SetFocus
```

```
Else
```

```
MsgBox "login successfully!", vbInformation, "successful attempt"
```

```
Form9.Show
```

```
Form4.Hide
```

```
End If
```

```
End Sub
```

MDI Form

```
Private sub mnubillgen_click()
```

```
Form14.show
```

```
MDIForm1.Hide
```

```
End sub
```

```
Private sub mnucustdet_click()
```

```
Form16.show
```

```
MDIForm1.Hide
```

```
End sub
```

```
Private sub mnucustrep_click()
```

```
DataReport2.show
```

```
MDIForm1.Hide
```

```
End sub
```

```
Private sub mnuempdet_click()
```

```
Form6.show
```

```
MDIForm1.Hide
```

```
End sub
```

```
Private sub mnuemprep_click()
```

```
DataReport3.show
```

```
MDIForm1.Hide
```

```
End sub
```

```
Private sub mnuprodrep_click()
```

```
DataReport1.show
```

MDIForm1.Hide

End sub

Private sub mnuexit_click()

Form3.show

MDIForm1.sho

End sub

Employee Details Form

Private Sub cmdaddnewemp_Click()

Form13.Show

Form6.Hide

End Sub

Private Sub cmdback_Click()

Form2.Show

Form6.Hide

End Sub

Private Sub cmdexit_Click()

Form5.Show

Form6.Hide

End Sub

Private Sub cmdviewemplist_Click()

Form11.Show

Form6.Hide

End Sub

View Product Form

```
Dim connect As New adodb.Connection
```

```
Dim rs As New adodb.Recordset
```

```
Private Sub cmdclose_Click()
```

```
Form6.Show
```

```
Form7.Hide
```

```
End Sub
```

```
Private Sub cmdsearch_Click()
```

```
Dim List As ListItem
```

```
Set itmx = ListView1.FindItem(Text1.Text, Ivwtext, , Ivwpartial)
```

```
Text1.Text = ""
```

```
If itmx Is Nothing Then
```

```
MsgBox "record not found", vbCritical
```

```
Else
```

```
ListView1.ListItems(itmx.Index).Selected = True
```

```
ListView1.SetFocus
```

```
End If
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
Form8.Show
```

Customer Details Form

```
Private Sub clear()
```

```
Text1.Text = ""
```

```
Text2.Text = ""
```

```
Text3.Text = ""
```

```
Text4.Text = ""  
Text5.Text = ""  
Text6.Text = ""  
Text7.Text = ""  
Text8.Text = ""  
Text9.Text = ""  
Text10.Text = ""  
  
End Sub
```

```
Private Sub cmdadd_Click()  
Adodc1.Refresh  
Adodc1.Recordset.AddNew  
MsgBox ("enter new record")  
Call clear  
Text1.SetFocus  
  
End Sub
```

```
Private Sub cmdback_Click()  
Form16.Show  
Form8.Hide  
  
End Sub
```

```
Private Sub cmdclear_Click()  
Text1.Text = ""  
Text2.Text = ""  
Text3.Text = ""  
Text4.Text = ""  
Text5.Text = ""  
Text6.Text = ""
```

```
Text7.Text = ""
```

```
Text8.Text = ""
```

```
Text9.Text = ""
```

```
Text10.Text = ""
```

```
Adodc1.Refresh
```

```
Adodc1.Recordset.AddNew
```

```
Text1.SetFocus
```

```
End Sub
```

```
Private Sub cmddelete_Click()
```

```
Adodc1.Recordset.Delete
```

```
Call clear
```

```
MsgBox ("records deleted")
```

```
End Sub
```

```
Private Sub cmdexit_Click()
```

```
Form9.Show
```

```
Form8.Hide
```

```
End Sub
```

```
Private Sub cmdfind_Click()
```

```
Dim n As Double
```

```
n = InputBox("enter the record to find")
```

```
Adodc1.Refresh
```

```
Adodc1.Recordset.Find "CustomerID=" & n & ""
```

```
If Adodc1.Recordset.EOF Then
```

```
MsgBox ("records not found")
```

```
Text1.SetFocus
```

```
Else
```

```
Text1.Text = Adodc1.Recordset.Fields("CustomerID")
Text2.Text = Adodc1.Recordset.Fields("CustomerName")
Text3.Text = Adodc1.Recordset.Fields("StreetAddress")
Text4.Text = Adodc1.Recordset.Fields("StreetAddressLine2")
Text5.Text = Adodc1.Recordset.Fields("City")
Text6.Text = Adodc1.Recordset.Fields("State")
Text7.Text = Adodc1.Recordset.Fields("Pincode")
Text8.Text = Adodc1.Recordset.Fields("PhoneNumber")
Text9.Text = Adodc1.Recordset.Fields("Email")
Text10.Text = Adodc1.Recordset.Fields("Date")

End If

End Sub
```

```
Private Sub cmdfirst_Click()

Adodc1.Refresh

Adodc1.Recordset.MoveFirst

End Sub
```

```
Private Sub cmdlast_Click()

Adodc1.Refresh

Adodc1.Recordset.MoveLast

End Sub
```

```
Private Sub cmdmodify_Click()

Text1.SetFocus

End Sub
```

```
Private Sub cmdnext_Click()

Adodc1.Recordset.MoveNext
```

```
If Adodc1.Recordset.EOF Then
Adodc1.Recordset.MoveLast
MsgBox ("this is the last record and no more data")
End If
End Sub
```

```
Private Sub cmdprevious_Click()
Adodc1.Recordset.MovePrevious
If Adodc1.Recordset.BOF Then
Adodc1.Recordset.MoveFirst
MsgBox ("this is the first record")
End If
End Sub
```

```
Private Sub cmdsave_Click()
If Text1.Text = "" Then
MsgBox ("please enter Customer ID")
Text1.SetFocus
ElseIf Text2.Text = "" Then
MsgBox ("please enter Customer name")
Text2.SetFocus
ElseIf Text3.Text = "" Then
MsgBox ("please enter the street adress")
Text3.SetFocus
ElseIf Text4.Text = "" Then
MsgBox ("please enter the street adress Line 2")
Text4.SetFocus
ElseIf Text5.Text = "" Then
MsgBox ("please enter city")
```

```
Text5.SetFocus
ElseIf Text6.Text = "" Then
MsgBox ("please enter state")
Text6.SetFocus
ElseIf Text7.Text = "" Then
MsgBox ("please enter pincode")
Text7.SetFocus
ElseIf Text8.Text = "" Then
MsgBox ("please enter the phone number")
Text8.SetFocus
Else
Adodc1.Recordset.Update
MsgBox ("record saved")
Call clear
Text1.SetFocus
End If
End Sub
```

```
Private Sub Form_Load()
Call clear
End Sub
```

```
Private Sub Text10_Change()
Text10.Text = Format(Now)
End Sub
```

View Product Form

```
Dim connect As New adodb.Connection
Dim rs As New adodb.Recordset
```

```
Private Sub cmdclose_Click()
```

```
Form12.Show
```

```
Form10.Hide
```

```
End Sub
```

```
Private Sub cmdsearch_Click()
```

```
Dim List As ListItem
```

```
Set itmx = ListView1.FindItem(Text1.Text, Ivwtext, , Ivwpartial)
```

```
Text1.Text = ""
```

```
If itmx Is Nothing Then
```

```
MsgBox "record not found", vbCritical
```

```
Else
```

```
ListView1.ListItems(itmx.Index).Selected = True
```

```
ListView1.SetFocus
```

```
End If
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
With ListView1.ColumnHeaders
```

```
.Add , , "BarcodeID", Width / 12, ivwcolumnleft
```

```
.Add , , "Product Name", Width / 12, ivwcolumncentre
```

```
.Add , , "Standard Cost", Width / 12, ivwcolumncentre
```

```
.Add , , "Unit Price", Width / 12, ivwcolumncentre
```

```
.Add , , "Quantity in kg", Width / 12, ivwcolumncentre
```

```
.Add , , "Stock Date", Width / 12, ivwcolumncentre
```

```
.Add , , "GST in %", Width / 12, ivwcolumncentre
```

```
.Add , , "Total Cost", Width / 12, ivwcolumnright
```

```
End With
```

```
loaddata
End Sub

Sub dbconnection()

connect.Open"provider=microsoft.jet.OLEDB.4.0;data
source=C:\Users\hp\Desktop\product3.mdb"

End Sub

Public Sub loaddata()

Dim itm As ListItem

ListView1.ListItems.Clear

dbconnection

rs.Open "select*from product3", connect, adOpenDynamic, adLockOptimistic

Do Until rs.EOF

Set List = ListView1.ListItems.Add( , rs!BarcodeID)

List.SubItems(1) = rs!ProductName
List.SubItems(2) = rs!StandardCost
List.SubItems(3) = rs!UnitPrice
List.SubItems(4) = rs!Quantityinkg
List.SubItems(5) = rs!StockDate
List.SubItems(6) = rs!GST
List.SubItems(7) = rs!Total

rs.MoveNext

Loop

End Sub
```

Employee Details Form

```
Dim connect As New adodb.Connection

Dim rs As New adodb.Recordset
```

```
Private Sub cmdclose_Click()
```

```
Form6.Show
```

```
Form11.Hide
```

```
End Sub
```

```
Private Sub cmdsearch_Click()
```

```
Dim List As ListItem
```

```
Set itmx = ListView1.FindItem(Text1.Text, Ivwtext, , Ivwpartial)
```

```
Text1.Text = ""
```

```
If itmx Is Nothing Then
```

```
MsgBox "record not found", vbCritical
```

```
Else
```

```
ListView1.ListItems(itmx.Index).Selected = True
```

```
ListView1.Refresh
```

```
End If
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
With ListView1.ColumnHeaders
```

```
.Add , , "empid", Width / 15, ivwcolumnleft
```

```
.Add , , "empname", Width / 15, ivwcolumncentre
```

```
.Add , , "dob", Width / 15, ivwcolumncentre
```

```
.Add , , "address", Width / 15, ivwcolumncentre
```

```
.Add , , "state", Width / 15, ivwcolumncentre
```

```
.Add , , "pincode", Width / 15, ivwcolumncentre
```

```
.Add , , "phoneno", Width / 15, ivwcolumncentre
```

```
.Add , , "emailid", Width / 15, ivwcolumncentre
```

```
.Add , , "salary", Width / 15, ivwcolumncentre
```

```
.Add , , "username", Width / 15, ivwcolumncentre
```

```
.Add , , "password", Width / 15, ivwcolumnright
End With

loaddata

End Sub

Sub dbconnection()
connect.Open                                     "provider=microsoft.jet.oledb.4.0;data
Source=C:\Users\hp\Desktop\employee2.mdb"
End Sub

Public Sub loaddata()
Dim itm As ListItem
ListView1.ListItems.Clear

dbconnection

rs.Open "select*from employee2", connect, adOpenDynamic, adLockOptimistic

Do Until rs.EOF
Set List = ListView1.ListItems.Add(, , rs!empid)
List.SubItems(1) = rs!empname
List.SubItems(2) = rs!dob
List.SubItems(3) = rs!address
List.SubItems(4) = rs!State
List.SubItems(5) = rs!Pincode
List.SubItems(6) = rs!phoneno
List.SubItems(7) = rs!emailid
List.SubItems(8) = rs!salary
List.SubItems(9) = rs!username
List.SubItems(10) = rs!password
rs.MoveNext
```

Loop

End Sub

Product Details Form

Private Sub cmdaddnewprod_Click()

Form17.Show

Form12.Hide

End Sub

Private Sub cmdback_Click()

Form9.Show

Form12.Hide

End Sub

Private Sub cmdexit_Click()

Form2.Show

Form12.Hide

End Sub

Private Sub cmdview_Click()

Form10.Show

Form12.Hide

End Sub

Add New Employee Form

Private Sub cmdaddnew_Click()

adoemp.Refresh

adoemp.Recordset.AddNew

MsgBox ("enter new record")

Call clear

Text1.SetFocus

End Sub

Private Sub cmdback_Click()

Form6.Show

Form13.Hide

End Sub

Private Sub cmdclear_Click()

Text1.Text = ""

Text2.Text = ""

Text3.Text = ""

Text4.Text = ""

Text5.Text = ""

Text6.Text = ""

Text7.Text = ""

Text8.Text = ""

Text9.Text = ""

Text10.Text = ""

Text11.Text = ""

adoemp.Refresh

adoemp.Recordset.AddNew

Text1.SetFocus

End Sub

Public Sub clear()

Text1.Text = ""

Text2.Text = ""

Text3.Text = ""

```
Text4.Text = ""  
Text5.Text = ""  
Text6.Text = ""  
Text7.Text = ""  
Text8.Text = ""  
Text9.Text = ""  
Text10.Text = ""  
Text11.Text = ""  
End Sub
```

```
Private Sub cmddelete_Click()  
If Text1.Text = "" Then  
MsgBox ("unable to delete fields are empty")  
Text1.SetFocus  
Else  
adoemp.Recordset.Delete  
Call clear  
MsgBox ("records deleted")  
Text1.SetFocus  
End If  
End Sub
```

```
Private Sub cmdexit_Click()  
Form5.Show  
Form13.Hide  
End Sub
```

```
Private Sub cmdfind_Click()  
Dim n As Double
```

```
n = InputBox("enter the record to find")
adoemp.Refresh
adoemp.Recordset.Find "empid=" & n & ""
If adoemp.Recordset.EOF Then
MsgBox ("records not found")
Else
Text1.Text = adoemp.Recordset.Fields("empid")
Text2.Text = adoemp.Recordset.Fields("empname")
Text3.Text = adoemp.Recordset.Fields("dob")
Text4.Text = adoemp.Recordset.Fields("address")
Text5.Text = adoemp.Recordset.Fields("state")
Text6.Text = adoemp.Recordset.Fields("pincode")
Text7.Text = adoemp.Recordset.Fields("phoneno")
Text8.Text = adoemp.Recordset.Fields("emailid")
Text9.Text = adoemp.Recordset.Fields("salary")
Text10.Text = adoemp.Recordset.Fields("username")
Text11.Text = adoemp.Recordset.Fields("password")
End If
End Sub
```

```
Private Sub cmdfirst_Click()
adoemp.Refresh
adoemp.Recordset.MoveFirst
End Sub
```

```
Private Sub cmdlast_Click()
adoemp.Refresh
adoemp.Recordset.MoveLast
End Sub
```

```
Private Sub cmdmodify_Click()  
    adoemp.Refresh  
    Text1.SetFocus  
End Sub
```

```
Private Sub cmdnext_Click()  
    adoemp.Recordset.MoveNext  
    If adoemp.Recordset.EOF Then  
        adoemp.Recordset.MoveLast  
        MsgBox ("this is the last record and no more data")  
    End If  
End Sub
```

```
Private Sub cmdprevious_Click()  
    adoemp.Recordset.MovePrevious  
    If adoemp.Recordset.BOF Then  
        adoemp.Recordset.MoveFirst  
        MsgBox ("this is the first record")  
    End If  
End Sub
```

```
Private Sub cmdsave_Click()  
    If Text1.Text = "" Then  
        MsgBox ("please enter empid")  
        Text1.SetFocus  
    ElseIf Text2.Text = "" Then  
        MsgBox ("please enter name")  
        Text2.SetFocus
```

```
ElseIf Text3.Text = "" Then
MsgBox ("please enter dob")
Text3.SetFocus
ElseIf Text4.Text = "" Then
MsgBox ("please enter address")
Text4.SetFocus
ElseIf Text5.Text = "" Then
MsgBox ("please enter state")
Text5.SetFocus
ElseIf Text6.Text = "" Then
MsgBox ("please enter pincode")
Text6.SetFocus
ElseIf Text7.Text = "" Then
MsgBox ("please enter phoneno")
Text.SetFocus
ElseIf Text8.Text = "" Then
MsgBox ("please enter emailid")
Text8.SetFocus
ElseIf Text9.Text = "" Then
MsgBox ("please enter salary")
Text9.SetFocus
ElseIf Text10.Text = "" Then
MsgBox ("please enter username")
Text10.SetFocus
ElseIf Text11.Text = "" Then
MsgBox ("please enter password")
Text11.SetFocus
Else
adoemp.Recordset.Update
```


MsgBox ("record saved")

Call clear

Text1.SetFocus

End If

End Sub

Private Sub Form_Load()

Call clear

End Sub

Bill Generation Form

Private Sub cmdback_Click()

Form9.Show

Form14.Hide

End Sub

Private Sub cmdcalculate_Click()

Label21.Caption = Val(Label9.Caption) + Val(Label10.Caption) + Val(Label11.Caption) +
Val(Label12.Caption) + Val(Label13.Caption) + Val(Label14.Caption) +
Val(Label15.Caption) + Val(Label18.Caption) + Val(Label19.Caption)

End Sub

Private Sub cmdgenerate_Click()

Form15.Show

Form14.Hide

End Sub

Private Sub Command1_Click()

Dim n As Long

n = Text1.Text

```
adoemp.Refresh  
adoemp.Recordset.Find "BarcodeID=" & n & ""  
If adoemp.Recordset.EOF Then  
MsgBox ("records not found")  
Text1.Text = ""  
Text1.SetFocus  
Else  
Text2.SetFocus  
Label2.Caption = adoemp.Recordset.Fields("ProductName")  
Label9.Caption = adoemp.Recordset.Fields("Total")  
End If  
End Sub
```

```
Private Sub Command2_Click()  
Dim n As Long  
n = Text2.Text  
adoemp.Refresh  
adoemp.Recordset.Find "BarcodeID=" & n & ""  
If adoemp.Recordset.EOF Then  
MsgBox ("records not found")  
Text2.Text = ""  
Text2.SetFocus  
Else  
Text2.SetFocus  
Label3.Caption = adoemp.Recordset.Fields("ProductName")  
Label10.Caption = adoemp.Recordset.Fields("Total")  
End If  
End Sub
```

```
Private Sub Command3_Click()
Dim n As Long
n = Text3.Text
adoemp.Refresh
adoemp.Recordset.Find "BarcodeID=" & n & ""
If adoemp.Recordset.EOF Then
MsgBox ("records not found")
Text3.Text = ""
Text3.SetFocus
Else
Text3.SetFocus
Label4.Caption = adoemp.Recordset.Fields("ProductName")
Label11.Caption = adoemp.Recordset.Fields("Total")
End If
End Sub
```

```
Private Sub Command4_Click()
Dim n As Long
n = Text4.Text
adoemp.Refresh
adoemp.Recordset.Find "BarcodeID=" & n & ""
If adoemp.Recordset.EOF Then
MsgBox ("records not found")
Text4.Text = ""
Text4.SetFocus
Else
Text4.SetFocus
Label5.Caption = adoemp.Recordset.Fields("ProductName")
Label12.Caption = adoemp.Recordset.Fields("Total")
```

End If

End Sub

Private Sub Command5_Click()

Dim n As Long

n = Text5.Text

adoemp.Refresh

adoemp.Recordset.Find "BarcodeID=" & n & ""

If adoemp.Recordset.EOF Then

MsgBox ("records not found")

Text5.Text = ""

Text5.SetFocus

Else

Text5.SetFocus

Label6.Caption = adoemp.Recordset.Fields("ProductName")

Label13.Caption = adoemp.Recordset.Fields("Total")

End If

End Sub

Private Sub Command6_Click()

Dim n As Long

n = Text6.Text

adoemp.Refresh

adoemp.Recordset.Find "BarcodeID=" & n & ""

If adoemp.Recordset.EOF Then

MsgBox ("records not found")

Text6.Text = ""

Text6.SetFocus

Else

```
Text6.SetFocus  
Label7.Caption = adoemp.Recordset.Fields("ProductName")  
Label14.Caption = adoemp.Recordset.Fields("Total")  
End If  
End Sub
```

```
Private Sub Command7_Click()  
Dim n As Long  
n = Text7.Text  
adoemp.Refresh  
adoemp.Recordset.Find "BarcodeID=" & n & ""  
If adoemp.Recordset.EOF Then  
MsgBox ("records not found")  
Text7.Text = ""  
Text7.SetFocus  
Else  
Text7.SetFocus  
Label8.Caption = adoemp.Recordset.Fields("ProductName")  
Label15.Caption = adoemp.Recordset.Fields("Total")  
End If  
End Sub
```

```
Private Sub Command8_Click()  
Dim n As Long  
n = Text8.Text  
adoemp.Refresh  
adoemp.Recordset.Find "BarcodeID=" & n & ""  
If adoemp.Recordset.EOF Then  
MsgBox ("records not found")
```

```
Text8.Text = ""  
Text8.SetFocus  
Else  
Text8.SetFocus  
Label16.Caption = adoemp.Recordset.Fields("ProductName")  
Label18.Caption = adoemp.Recordset.Fields("Total")  
End If  
End Sub
```

```
Private Sub Command9_Click()  
Dim n As Long  
n = Text9.Text  
adoemp.Refresh  
adoemp.Recordset.Find "BarcodeID=" & n & ""  
If adoemp.Recordset.EOF Then  
MsgBox ("records not found")  
Text9.Text = ""  
Text9.SetFocus  
Else  
Text9.SetFocus  
Label17.Caption = adoemp.Recordset.Fields("ProductName")  
Label19.Caption = adoemp.Recordset.Fields("Total")  
End If  
End Sub
```

Customer Details Form

```
Private Sub cmdaddnewcust_Click()  
Form8.Show  
Form16.Hide
```

End Sub

Private Sub cmdback_Click()

Form9.Show

Form16.Hide

End Sub

Private Sub cmdviewcust_Click()

Form7.Show

Form16.Hide

End Sub

Add New Product Form

Dim n As String

Dim t As String

Private Sub cmdaddnew_Click()

adoemp.Refresh

adoemp.Recordset.AddNew

MsgBox ("enter new record")

Call clear

Text1.SetFocus

End Sub

Private Sub cmdback_Click()

Form12.Show

Form17.Hide

End Sub

```
Private Sub cmdcalculate_Click()
If Text3.Text = "" Then
MsgBox "please enter Standard Cost"
Text3.SetFocus
ElseIf Text4.Text = "" Then
MsgBox "please enter Product GST"
Text4.SetFocus
ElseIf Text7.Text = "" Then
MsgBox "please enter product discount"
Text7.SetFocus
Else
n = Val(Text3.Text) + Val(Text3.Text) / 100 * Val(Text4.Text)
t = n / 100 * Val(Text7.Text)
Text8.Text = (n - t) * Val(Text5.Text)
End If
End Sub
```

```
Private Sub cmdclear_Click()
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
Text7.Text = ""
Text8.Text = ""
adoemp.Refresh
adoemp.Recordset.AddNew
Text1.SetFocus
```


End Sub

Public Sub clear()

Text1.Text = ""

Text2.Text = ""

Text3.Text = ""

Text4.Text = ""

Text5.Text = ""

Text6.Text = ""

Text7.Text = ""

Text8.Text = ""

End Sub

Private Sub cmddelete_Click()

adoemp.Recordset.Delete

Call clear

MsgBox ("records deleted")

End Sub

Private Sub cmdexit_Click()

Form9.Show

Form17.Hide

End Sub

Private Sub cmdfind_Click()

Dim n As Integer

n = InputBox("enter the record to find")

adoemp.Refresh

adoemp.Recordset.Find "BarcodeID=" & n & ""

```
If adoemp.Recordset.EOF Then
MsgBox ("records not found")
Text1.SetFocus
Else
Text1.Text = adoemp.Recordset.Fields("BarcodeID")
Text2.Text = adoemp.Recordset.Fields("ProductName")
Text3.Text = adoemp.Recordset.Fields("StandardCost")
Text4.Text = adoemp.Recordset.Fields("UnitPrice")
Text5.Text = adoemp.Recordset.Fields("Quantityinkg")
Text6.Text = adoemp.Recordset.Fields("StockDate")
Text7.Text = adoemp.Recordset.Fields("GST")
Text8.Text = adoemp.Recordset.Fields("Total")
End If
End Sub
```

```
Private Sub cmdfirst_Click()
adoemp.Refresh
adoemp.Recordset.MoveFirst
End Sub
```

```
Private Sub cmdlast_Click()
adoemp.Refresh
adoemp.Recordset.MoveLast
End Sub
```

```
Private Sub cmdmodify_Click()
Text1.SetFocus
End Sub
```

```
Private Sub cmdnext_Click()  
adoemp.Recordset.MoveNext  
If adoemp.Recordset.EOF Then  
adoemp.Recordset.MoveLast  
MsgBox ("this is the last record and no more data")  
End If  
End Sub
```

```
Private Sub cmdprevious_Click()  
adoemp.Recordset.MovePrevious  
If adoemp.Recordset.EOF Then  
adoemp.Recordset.MoveFirst  
MsgBox ("this is the first record")  
End If  
End Sub
```

```
Private Sub cmdsave_Click()  
If Text1.Text = "" Then  
MsgBox ("please enter the product barcode")  
Text1.SetFocus  
ElseIf Text2.Text = "" Then  
MsgBox ("please enter the product name")  
Text2.SetFocus  
ElseIf Text3.Text = "" Then  
MsgBox ("please enter the standard cost")  
Text3.SetFocus  
ElseIf Text4.Text = "" Then  
MsgBox ("please enter the product discount")  
Text4.SetFocus
```

```
ElseIf Text5.Text = "" Then
MsgBox ("please enter the quantity of the product")
Text5.SetFocus
ElseIf Text6.Text = "" Then
MsgBox ("please enter the stock date")
Text6.SetFocus
ElseIf Text7.Text = "" Then
MsgBox ("please enter GST of the product")
Text7.SetFocus
ElseIf Text8.Text = "" Then
MsgBox ("please calculate the total amount")
Text8.SetFocus
Else
adoemp.Recordset.Update
MsgBox ("record saved")
Call clear
Text1.SetFocus
End If
End Sub
```

```
Private Sub Form_Load()
Call clear
End Sub
```

6. SYSTEM TESTING

Testing performs a very critical role for quality assurance and ensuring the reliability of the software. The success of testing for errors in program depends critically on the test cases. The basic levels of testing are:

1. Unit testing
2. Integrating testing
3. System testing
4. Acceptance testing

SYSTEM TESTING

Software Testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing presents an interesting anomaly for the software. The testing phase involves testing of a system using various test data. Preparation of the test data plays a vital role in the system testing. After preparation the test data, the system under study is tested using those test data. Errors were found and corrected by using the following testing steps and corrections are recorded for future reference. Thus, series of testing is performed on the system before it is ready for implementation.

The development of software system involves a series of production activities where opportunities for injection of human errors are enormous. Errors may begin to occur at the very inception of the process where the objectives may be enormously or imperfectly specified as well in later design and development stages. Because of human inability to perform and communicate with perfection, software development is followed by assurance activity.

Quality assurance is the review of software products and related documentation for completeness, correctness, reliability and maintainability. And of course it includes assurances that the system meets the specification and the requirements for the intended use and performance. The various levels of quality assurance are described in the following sections.

TEST CASE DESIGN

The test case design for software should be done in such a way that the errors can be found out with a minimum amount of time and effort. A test case is a set of data that the system will process as normal as input. Mainly there are two methods for the test case design-Black box tests and White box Testing.

BLACK BOX TESTING

This method focuses on the functional requirements of the software. It attempts to find out the errors of the following categories-incorrect and missing functions, interface errors, errors in data structures or external database access, performance errors and initialization and termination errors.

WHITE BOX TEST

This method is also called path testing, is a test design method that uses the control structure of the procedural design to derive test cases. It is predicted on a close examination of procedural design. Logical paths through which the software are tested by providing test cases that exercise specific sets of conditions and or loops.

In the case of the present projects, both the White box testing and the Black box testing were conducted successfully. The functionality of the software is also tested. All loops and conditional statements of the software were executed correctly within their operations bounds. Test cases which will result in the execution of every instruction in the modules where developed. All the transaction paths were tested to identify and correct the errors occurred. Mainly there are two levels of testing, namely Unit testing and Integration Testing.

TESTING STRATEGY

An initial testing strategy is developed upon completion of the Analysis stage and is refined and updated during the Design and Build stages of the project. The testing strategies followed in the project are as follows:

UNIT TESTING

Unit testing focuses on verification effort on the smallest unit of software design module. Using the unit test plans. Prepared in the design phase of the system as a guide, important control paths are tested. Boundary conditions were checked. All independent paths were exercises to ensure that all statements in the module are executed at least once and all error handling paths were test. Each unit was thoroughly tested to check if it might fall in any possible situation. This testing was carried out during the programming itself. At the end of this testing phase, each unit was found to be working satisfactorily, as regard to the output from the module.

INTEGRATION TESTING

Data can be lost across an interface one module can have an adverse effect on another's sub functions, when combined may not produce the desired major function; global data structures can be present problems. Integration testing is a symmetric technique for

constructing tests to uncover errors associated with the interface. All modules are combined in this testing step. Then the entire program was tested as a whole.

VALIDATION TESTING

At the culmination of the integration testing, the software was completely assembled as a package, interfacing errors have been uncovered and corrected and a final series of software validation began. Here we test the system in a manner that can be reasonably expected by customer, the system was tested against system requirement.

OUTPUT TESTING

After performing validation test the next phase in output test of the system, since no system could be useful if it does not produce the desired output in a desired format. By considering the format of the report/output, output/ report is generated or displayed and is tested. Here output format is considered in two ways: one is the screen and other is a printed form.

TESTING PROCESS

First thing in the testing process is to make plans. In the test plan, test cases are selected to ensure that there is an error in the program then it is executed by one of the test cases. The success of testing depends critically on these test cases selected.

TEST CASE SPECIFICATION

It is the major activity in the testing process. Test cases are specified for testing each unit, the specification gives all the test cases inputs to be used and the outputs expected for these test cases.

7.PERFORMANCE AND LIMITATIONS

PERFORMANCE:

- It can be upgrade in mere future.
- All modules have been tested and it is working properly.
- Basic security feature has been given, data validation is done.
- It supports various reports.
- It does not have any Constraints.

LIMITATIONS:

- Less Reliable due to human Errors.
- Large Storage Area is required.
- No Online Service, but can be done in Future.
- The existing System is time consuming.

8. FUTURE ENHANCEMENT

- The project made here is just to ensure that this product could be valid in today real challenging world.
- The renovation of Our Project would increase the flexibility of the System.
- The programs were coded in an easier and structured manner so that further modifications may be incorporated easily.
- In future more, modules will be introduced.
- More security towards the project will be given in future.
- Automatic billing of purchase order.

9.CONCLUSION

Our project is on Global Market Billing System. We have successfully completed it. In conclusion Global Market Billing System has to do with making appropriate effort to stop the raising problem to all manual Global Market operation in order to enhance the operation of such global market in this project the software or system that can be used to aid all global market that is still operating manually have been successfully developed. The software can be implementing in all types of global market has mentioned.

10.APPENDIX A

10.1 SAMPLE SCREEN

LOADING FORM



WELCOME FORM



ADMIN LOGIN FORM


Form3

GLOBAL MARKET BILLING SYSTEM

Administrator login

User id

password




Form3

GLOBAL MARKET BILLING SYSTEM

Administrator login

User id

password



Project1(global market)

login successfully

OK

EMPLOYEE LOGIN FORM

Form4

GLOBAL MARKET BILLING SYSTEM

Employee Login

user id

password

successful attempt

login successfully!

OK

MDI FORM



EMPLOYEE DETAILS

Form6

GLOBAL MARKET BILLING SYSTEM



Add New Emp

View Emp List

Back

exit

VIEW CUSTOMER FORM

Form7

GLOBAL MARKET BILLING SYSTEM

Search by customer id

CustomerID	CustomerN...	StreetAddr...	City	State	Pincode	Phoneno	Email
1001	anu	kgf	bangalore	karnataka	563122	8088551594	anu@123
1002	thriveni	chokrabanda	kgf	karnataka	563124	978653421	thiru@123
1234	sneha	allikallu	kgf	karnataka	563122	8088456773	sneha@1223
1345	sneha	kgf	bangarpet	karnataka	563211	987564231	sneha@123
2345	anu	kgf	bpt	karnataka	563122	8088551854	anu@123

Close

CUSTOMER DETAILS

Form8

GLOBAL MARKET BILLING SYSTEM

customer ID: 1001 Pincode: 563122

Customer Name: anu Phone Number: 8088551594

Street Address: kgf E-mail: anu@123

City: bangalore Date: 09-09-2022

State: karnataka

Save Clear Back Exit

Modify Add new Find Delete

First Last Previous Next

Adodc1

VIEW PRODUCT DETAILS

Form10

GLOBAL MARKET BILLING SYSTEM

Search by barcode Search

BarcodeID	Product N...	Standard ...	Unit Price	Quantity in...	Stock Date	GST in %	Total Cost
1001	papaya	60	6	5	03-04-2021	4	305.28
1002	pomogran...	200	3	4	20-03-2021	8	838.08
1005	barley	60	3	4	03-04-2021	6	246.768
1012	Rice	90	3	3	20-03-2021	8	282.852
1234	cucumber	50	3	3	02-03-2021	7	155.685
2345	Tamoto	30	5	1	20-08-2022	7	30.495
3456	onion	30	1	2	20-08-2022	10	32.67

Close

EMPLOYEE DETAILS

Form11

GLOBAL MARKET BILLING SYSTEM

search by employee id

empid	empname	dob	address	state	pincode	phoneno	emailid	salary	username	password
1008	nafeesa	06-03-2001	kgf	karnataka	563244	8365638763	naffu@123	50000	naffu	123
1001	hamsa	02-06-2001	donimadagu	karnataka	563129	9844794546	hamsa@123	50000	hamsa	1234
1003	amrtha	03-04-2001	kgf	karnataka	563126	8745364786	ammu@1234	40000	ammu	34232
1005	sneha	15-09-2002	allikallu	karnataka	563125	8967456345	sneha@123	60000	ammu	21223
1007	anupriya	24-02-2002	bangalore	karnataka	563122	8088551594	anu@123	50000	anu	1234
1234	deepika	02-03-2001	kgf	karnataka	563124	987664322	deepu@123	60000	deepu	3456
3456	swathi	02-04-2001	kgf	karnataka	546788	6784999494	swathi@123	50000	swathi	3456

PRODUCT DETAILS

Form12

GLOBAL MARKET BILLING SYSTEM



ADD NEW EMPLOYEE

Form13

GLOBAL MARKET BILLING SYSTEM

Emp Id	1003	Phone No	8745364786
Emp Name	amrtha	Email Id	ammu@1234
DOB	03-04-2001	Salary	40000
Address	kgf		
State	karnataka	Username	ammu
Pincode	563126	Password	34232

SaveModifyAdd newFind

FirstPreviousNextLast

ClearDeleteBackExit

BILL GENERATION

Form14(bill generation)

GLOBAL MARKET BILLING SYSTEM

1001	search	papaya	305.28
1012	search	Rice	282.852
3456	seach	onion	32.67
2345	search	Tamoto	30.495
	search		
	search		
	search		
	search		
	search		

total amount: 651.297

Calculate

Generate Bill

Back

CUSTOMER DETAILS

form16(customer details)

GLOBAL MARKET BILLING SYSTEM

Add new customer details

View customer list

Back



ADD NEW PRODUCT

Form17

GLOBAL MARKET BILLING SYSTEM

Product barcode	1012	Quantity	3
Product name	Rice	Current date	20-03-2021
Standard cost	90	Discount in %	8
GST in %	3	Total	282.852

Calculate

Ado

Add new	Save	Modify	Find	Clear	Delete
First	Previous	Next	Last	Back	Exit

11. APPENDIX B

PRODUCT REPORT

DataReport1	
Zoom 100%	
PRODUCTS REPORT	

BarcodeID:	2345
ProductName:	Tamoto
StandardCost:	30
UnitPrice:	5
QuantityInkg:	1
StockDate:	20-08-2022
GST:	7
Total:	30.495

CUSTOMER REPORT

DataReport2

Zoom 100%

customer report

CustomerID:	2345
CustomerNam	anu
StreetAddress:	kgf
StreetAddress	bangalore
City:	bpt
State:	karnataka
Pincode:	563122
PhoneNumber:	8088551854
Email:	anu@123
Date:	24-08-2022

Pages: 1

EMPLOYEE REPORT

The screenshot shows a window titled "DataReport3" with a zoom level of 100%. The report content is as follows:

EMPLOYEE REPORT	
empid:	1008
empname:	nafeesa
dob:	06-03-2001
address:	kgf
state:	karnataka
pincode:	563244
phoneno:	8365638763
emailid:	naffu@123
salary:	50000
username:	naffu
password:	123

At the bottom of the window, there is a "Pages:" section with a navigation bar showing "1" and various navigation icons.

12.REFERENCE

BIBILOGRAPHY

- VB Black Book (Author: Steven Holzner)
- The Complete Reference Visual Basic 6 (Author: Michael Harverson)
- Visual Programming (Author: K. Murugan)
- College Notes

ONLINE REFERENCES

- www.google.com
- www.youtube.com
- www.wikipedia.com

