1.INTRODUCTION

The project entitled "**CONTRACTOR AND LABOUR MAINTENANCE SYSTEM**" is developed to manage the construction work. The application will be developed using **VB .NET** and **MS SQL Server** and suits to run on any Windows platform.

Contractor and Labour management system is for homebuilders. It combines project scheduling, customer management, employee management and inventory monitoring in a single suite. It can be used by both production and custom homebuilders, but is also suitable for construction managers, residential remodelers, and design/build firms that work on residential projects as well as commercial builders.

Project scheduling module plans for pre-planning about new construction, site details, raw material availability verification, and engineering design. Customer management module stores the details about customer's registered for site, customer contact details, booking information, and payment details. Employee management and service management includes all other details and the information mandatory for completion and updation of construction process.

This system involves a lot of data processing work, which is difficult to do manually because of lot of computations and transaction it requires. The construction will start only if the required materials in hand are in shortage. Every coustomer's requirements will be recorded for future reference as well as strat the process. The detailed process includes planning, cost estimation, purchase material, billing and receipt to the customer.

1.1.Organization Profile

KICE INFOSYSTEMS is a professional website designing, Customized software development, Business process Outsourcing – IT/ITES & Internet marketing company providing full featured web services including B2B Acquisition & B2C ecommerce solutions and acting as an offshore development center for overseas development firms. KICE Infosystems is an innovative company, based in India that provides a series of Web-based and software applications that have helped their customer to create successful business ventures through online initiatives. KICE Infosystems provide all the services that a company needs to get online from web designing to web hosting and manage leading-edge Web sites and e-business applications. Quality and Client Satisfaction are primarily the telling factors of KICE Infosystems success in the domestic market. Ever since its inception, **KICE Info systems** has accrued continuous growth in all its business functions and this has been possible only due to its commitment, quality training methodologies, the services it offers, knowledge sharing with industry leaders and professional approach.

Our mission is to embrace our students in gaining quality based Knowledge in computer fields with real-time project development, placement oriented courses and traning, extending the knowledge based service and education. We give opportunity to transform his/her life and help them to transform the organization they work for and the communities they live in.

The reputation of KICE INFOSYSTEM is anchored in the professionalism, ethics and excellence of service our people have striven to demonstrate and embrace every day. Every project is adapted to the scale and specific needs of the client: privete or public sector, small, medium or large organizations.

1.2 System Specifications

1.2.1 HARDWARE CONFIGURATION

Processor	: Pentium -IV
Speed	: 1 GHz
Hard Disk Capacity	: 40GB
RAM Capacity	: 1GB RAM
CD-ROM Drive	: 52x speed
Keyboard	: 104 keys
Mouse	: Logitech
Printer	: HP3745 series DeskJet printer

1.2.2. SOFTWARE SPECIFICATION

Operating System	:	Windows XP
Front End	:	VB.NET
Back End	:	MS- SQL Server

2. SYSTEM STUDY

Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

The feasibility of a proposed solution is evaluated in teams of its components. These components are:

- Economic feasibility
- Technical feasibility

Economic Feasibility

The economic feasibility study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development or the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical Feasibility

The technical feasibility study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The will lead to high demands on the available technical resources. This will lead to high demands being places on the client. The developed system must have modest requirements, as only minimal or null changes are required for implementing this system.

2.1. EXISTING SYSTEM

In this system every work is carried out manually which is time consuming and leads to miss entry of details. Searching and updating records is tedious process. Storage of data needs separate and provide to store the books if not so; it leads to loss of data. This system is also much time consuming and expensive.

The above result in these factors leads to formulate a new system, which could help the organization in all aspects to make things easier, faster and efficient then the existing manual system.

2.1.1. DRAWBACKS

- Manual work
- Security of information is low
- Calculations are difficult
- It needs more time
- Errors occur frequently
- Occupies more resources
- Needs a lot of manpower

2.2. PROPOSED SYSTEM

The existing system is not functioning effectively due to manual processing. Computer dominates major roles in banking activities. Primarily computers are used in banking organization for accomplishing activities at faster rate with up-to date information accuracy and cost reduction.

The proposed system is an effective menu driven package. This package is more powerful than the existing system. The package also aims to provide faster service to the management proposed system thus aim at removing all the complexities in the existing system.

2.2.1. FEATURES

- The system is more user friendly
- It is equipped with powerful GUI(Graphical User Interface)
- The interrelated data are grouped into different input screens.
- Provides a high uniformity among all screens format.
- The system works in high speed and accuracy.
- It generates neat formatted report, based on which the decision is possible.
- It handles bulk amount of data.

3. SYSTEM DESIGN AND DEVELOPMENT

Design is concerned with identifying software components specifying relationship Among components. Specifying software structure and providing blue print for the document phase. Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. In such a manner, the interaction between parts is Minimal clearly specified. Design will explain software components in details. This will help the implementation of the system. Moreover, this will guide the further changes in the system to satisfy the further requirements.

The design document describes how to transform, the requirement and the functional design into more technical system design specification. This design involves conceiving and planning out in the mind and making a drawing pattern of sketch of. It includes type of activities, External Design, Architectural Design and Detailed Design. The architectural design and detailed design collectively referred to as internal design.

The external design involves specifying the externally observable characteristics of a software product and the internal design involves specifying the internal structure and processing details of the system. The fundamental concept of the design include abstraction structure, information hiding Modularity, concurrency, verification and design aesthetics.

3.1 FILE DESIGN

In computing, a file design (or file system) is used to control how data is stored and retrieved. Without a file system, information placed in a storage area would be one large body of data with no way to tell where one piece of information stops and the next begins. By separating the data into individual pieces, and giving each piece a name, the information is easily separated and identified. Taking its name from the way paper-based information systems are named, each group of data is called a "file". The structure and logic rules used to manage the groups of information and their names are called a "file system".

Some file systems are used on local data storage devices; others provide file access via a network protocol. Some file systems are "virtual", in that the "files" supplied are computed on request or are merely a mapping into a different file system used as a backing store. The file system manages access to both the content of files and the metadata about those files. It is responsible for arranging storage space; reliability, efficiency, and tuning with regard to the physical storage medium are important design considerations.

Following files are available in this application

- Customer detail
- Material detail
- Order detail
- Billing detail

3.2 INPUT DESIGN

The input design is the process of entering data to the system. The input design goal is to enter to the computer as accurate as possible. Here inputs are designed effectively so that errors made by the operations are minimized.

The inputs to the system have been designed in such a way that manual forms and the inputs are coordinated where the data elements are common to the source document and to the input. The input is acceptable and understandable by the users who are using it.

Input design is the process of converting user-originated inputs to a computer-based format input data are collected and organized into group of similar data. Once identified, appropriate input media are selected for processing.

The input design also determines the user to interact efficiently with the system. Input design is a part of overall system design that requires special attention because it is the common source for data processing error. The goal of designing input data is to make entry easy and free from errors.

Input design is the process of connecting the user-originated inputs into a computer to used format. The goal of the input design is to make the data entry logical & free from errors.

3.3 OUTPUT DESIGN

Output design is the process of converting computer data into hard copy that is understood by all. The various outputs have been designed in such a way that they represent the same format that the office and management used to.

Computer output is the most important and direct source of information to the user. Efficient, intelligible output design should improve the systems relationships with the user and help in decision making. A major form of output is the hardcopy from the printer.

Output requirements are designed during system analysis. A good starting point for the output design is the Data Flow Diagram (DFD). Human factors educe issues for design involves addressing internal controls to ensure readability.

The output form in the system is either by screen or by hard copies. Output design aims at communicating the results of the processing of the users. The reports are generated to suit the needs of the users. The reports have to be generated with appropriate levels.

All reports are output formats, maintained details can be reported over crystal reports, this project sustain following reports

Customer Details

Supplier Details

Material Details

Building Order

Billing

Reports

3.4 DATABASE DESIGN

The most important consideration in designing the database is how information will be used.

The main objectives of designing a database are:

Data Integration

In a database, information from several files are coordinated, accessed and operated upon as through it is in a single file. Logically, the information are centralized, physically, the data may be located on different devices, connected through data communication facilities.

Data Integrity

Data integrity means storing all data in one place only and how each application to access it. This approach results in more consistent information, one update being sufficient to achieve a new record status for all applications, which use it. This leads to less data redundancy; data items need not be duplicated; a reduction in the direct access storage requirement.

Data Independence

Data independence is the insulation of application programs from changing aspects of physical data organization. This objective seeks to allow changes in the content and organization of physical data without reprogramming of applications and to allow modifications to application programs without reorganizing the physical data.

The tables needed for each module were designed and the specification of each and every column was given based on the records and details collected during record specification of the system study.

3.5 SYSTEM DEVELOPMENT

The key to control maintenance costs is to design systems that are easy to change, so the link between development and maintenance is very strong. Many of the analysis and design methodologies, tools, and techniques employed during system development can be applied to system maintenance, but there are significant differences between development and maintenance. Maintainability is the ease with which software can be understood, corrected, adopted and enhanced.

3.5.1 DESCRIPTION OF MODULES

To develop this project several step should be followed. There are various modules in this proposed system they are listed below.

- Customer details
- Supplier details
- Contractor details
- Labor details
- Material details
- Building orders
- Planning and cost estimation
- Purchase material
- Billing details
- Reports

Customer Details:

This module is used to enter the details of the customer while they booking building orders or enquiry.

Supplier Details:

This module is used to enter the supplier details like name of the supplier, company address, contact person, and contact number

contractor Details:

This module is used to enter the contractor details like name of the contractor, company address, contact person, contact number, and so on.

Labor Details:

The company labor details are maintained by the admin through this module. This module holds the labor details such as labor id, labor name, address, contact number, and so on.

Material Details:

This module is used to store the material details like rate, quantity and the material type and supplier details.

Building Order :

This module is used to enter the building order requirement details from the customer. When the customer is given order to this company the order requirements will be stored by this module. The order detail includes room's details, no of rooms and other information about the building.

Planning and Cost Estimation:

When the order is confirmed the planning process will be prepared for the customer the planning process consist of no of day's required, total square feet for the building, construction requirements, cost estimation and other agreements.

Purchase Material :

This module is helps to enter the purchase details of the material from the supplier. The purchase details includes material description, cost of the material and supplier details.

Billing Details:

The billing details module is helps to prepare the billing for the customer after finished the building construction. The billing details includes bill no, bill date, customer details, total square feet of the building and other costs details.

Reports:

This module is used to generate various reports. This module prepares reports for the above module.

4. TESTING AND IMPLEMENTATION

System testing is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because Web-based systems and application reside on a network and interoperate with many different operating system, browsers, hardware platforms, and communication protocols; the search for errors represents a significant challenge for web application.

The distributed nature of client\server environments, the performance issues associated with transaction processing, the potential presence of a number of different hardware platforms, the complexities of network communication, the need to serve multiple clients from a centralized database and the requirements imposed on the server all combine to make testing of client\server architectures.

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer based system. System testing is the state of implementation that is aimed at assuring that the system works accurately and efficiently. Testing is the vital to the success of the system. System testing makes the logical assumption that if all the parts of the system are correct, the goal will be successfully achieved.

Unit Testing

Unit testing focuses verification efforts on the smallest unit of software design of the module. This is also known as “module testing”. This testing is carried out during programming stage itself. In this testing step, each module is found to be working satisfactorily as regards to the expected output of the modules.

In Project, Each module such Customer detail, Order detail, supplier detail modules are tested individually for example, Report details module can contain the more forms to maintain the information so all forms could be tested like entered information store appropriately in database access page or not. If correctly accessed means the testing of registration module successfully completed. Likewise all modules are tested successfully.

Integration Testing

Data can be lost across an interface, one module can have adverse effect on another sub function when combined it may not produce the desired major functions. Integration testing is a systematic testing for constructing test to uncover errors associated within an interface.

The objectives taken from unit tested modules and a program structure is built for integrated testing. All the modules are combined and the test is made.

A correction made in this testing is difficult because the vast expenses of the entire program complicated the isolation of causes. In this integration testing step, all the errors are corrected for next testing process.**In Project,** Integration of two modules can be tested together such as Order module and purchase modules are monitoring for verification purposes by the payment module. The communication of Entry and monitoring module can test and executed successfully.

Validation Testing

After the completion of the integrated testing, software is completely assembled as a package; interfacing error has been uncovered and corrected and a final series of software test validation begins.

Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software function in a manner that can be reasonably expected by the customer. After validation test has been conducted, one of two possible conditions exists:

In this project, Admin login details form Enter without username and password in textbox enter the submit button then Login failed message otherwise checks the both textbox value that is true means valid page displayed. Enter Password Displaying password character *.if it displays the characters security is not availed so testing of software is failed.

Output Testing

The next process of validation testing, is output testing of the proposed system, since no system could be successful if it does not produce the required output in the specified format. Asking the user about the format required, list the output to be generated or displayed by the system under considerations.

Output testing is a different test whose primary purpose is to fully exercise the computer based system although each test has a different purpose all the work should verify that all system elements have been properly integrated and perform allocated functions.

The output format on the screen is found to be corrected as the format was designed in the system design phase according to the user needs for the hard copy also; the output testing has not resulted in any correction in the system.

In project All the forms are tested as it gives the necessary output to the user's search such as view report details.

IMPLEMENTATION

System implementation is the stage of the project that the theoretical design is turned into a working system. If the implementation stage is not properly planned and controlled, it can cause error. Thus it can be considered to be the most crucial stage in achieving a successful new system and in giving the user confidence that the new system will work and be effective.

Normally this stage involves setting up a coordinating committee, which will act as a sounding board for ideas; complaints and problem. The first task is implementation planning; i.e., deciding on the methods and time scale to be adopted. Apart from planning two major task of preparing for implementation are, education takes place much earlier in the project; at the implementation stage the emphasis must be on training in new skills to give staff confidence they can use the system. Once staff has been trained, the system can be tested.

After the implementation phase is completed and the user staff is adjusted to the changes created by the candidate system, evaluation and maintenance is to bring the new system to standards.

5.CONCLUSION

The project title “**Contractor and Labour Maintenance Management System**” is developed successfully with various modules. This project is categorized into some basic modules, they are, customer details, building booking details module, planning module, purchase materials details module and billing calculation module. For every module there should be one report is available .It also generates reports of all needs such as requirement report, order report and construction details report etc., and this system satisfies all the requirements of the company and the application is developed by advanced software Visual Basic.net which is widely used in all applications.The system was tested with all possible samples of data and the performance of the system proves much effective and the data maintenance and manipulation is achieved practically.The system has been developed in Visual Basic .Net to reduce the response time and ensure flexibility. The system also gives opportunity for further development and enhancement of the existing system.

BIBLIOGRAPY

Books Referred :

- Tay Vaughan, “Multimedia Making it work”, Tata Mcgraw Hill, 5th edition.
- Peter Foster & Dr. Thomas B. Schalk , “Speech Recognition the complete Practical Reference Guide”, First Edition, August 1993.
- NITT’s Education Research and Development Group, “Structured System Analysis and Design”, Pace Education PVT Ltd 1993 Edition.
- Bill Evien and Jason Beres(2006), “Visual Basic.NET programming Bible”, wiley-dreamtech, special Edition.
- Roger S. Pressman, “ Software Engineering”, Tata me Graw Hill, 2000, Fifth Edition.
- Davin Reader, “Master SQL server 2000” BPB Publications, Third Edition.
- Elias M. Award, “System Analysis and design", Tata McGraw Hill , 3rd edition.
- Jeffrey A. Hoffer, Joey F. George and Joseph S. Valacich (2005), “Modern systems Analysis & Design”, pearson Education, 3rd Edition.

Websites:

www.e-speaking.com

www.speechapi.com

www.recording-microphones.com

www.sqlserver.com

www.yahoo.com

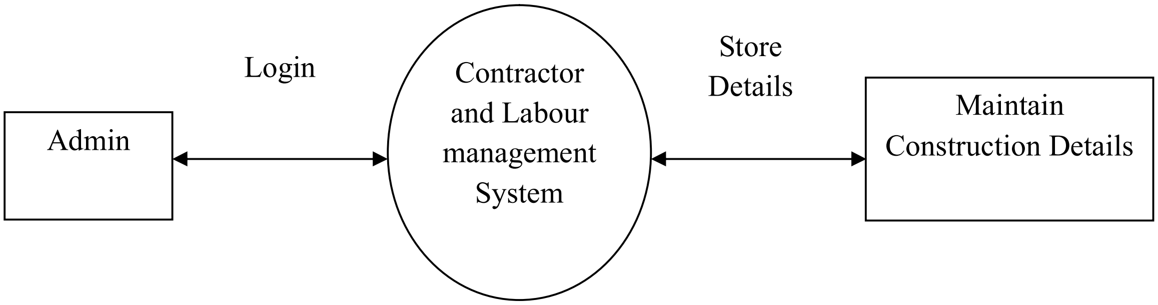
www.google.com

www.dotnetwatch.com

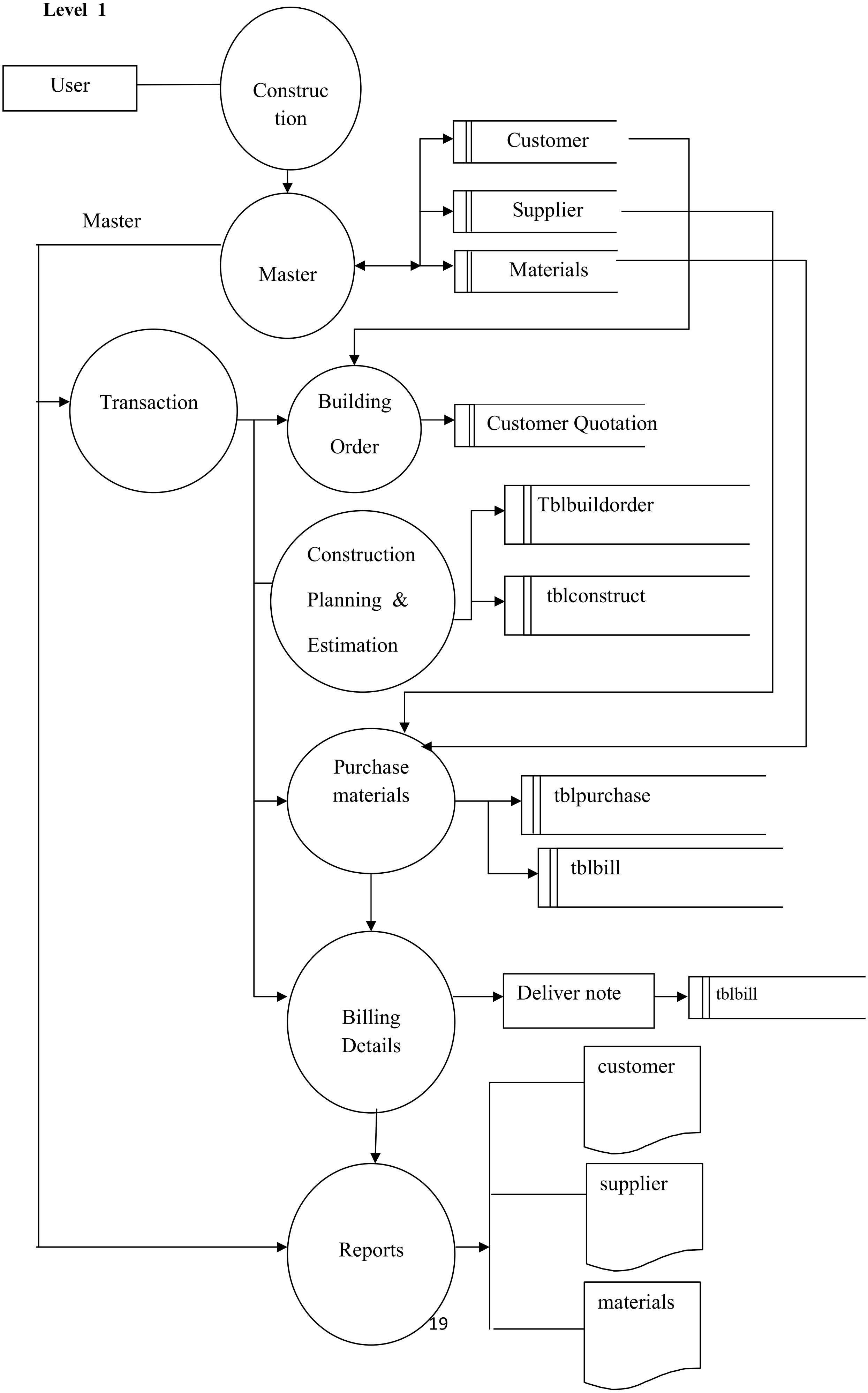
APPENDICES

A. Data Flow Diagram

Level 0


Login
Admin
Contractor and Labour management System
Store Details
Maintain Construction Details

Level 1


User
Construc tion
Master
Master
Customer
Supplier
Materials
Transaction
Building Order
Customer Quotation
Construction Planning & Estimation
Tblbuildorder
tblconstruct
Purchase materials
tblpurchase
tblbill
Billing Details
Deliver note
tblbill
Reports
customer
supplier
materials

B. TABLE STRUCTURE

Table Name : tblCustomer

Primary Key : Customer_id

Table Description : This table is used to maintain the details about customer

S.NO	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1	Customer_id	Int	8	Customer id
2	Customer_Name	Varchar	25	Customer Name
3	Address	Varchar	50	Address
4	Phone_no	Varchar	15	Phone number
5	Mobile_No	Varchar	15	Mobile Number

Table Name : tblSupplier

Primary Key : Supplier_id

Table Description : This table is used to maintain the details about supplier

S.NO	FIELD NAME	DATA TYPE	SIZE	DESCRIPTION
1	Supplier_id	Int	10	Supplier id
2	Supplier_Name	Varchar	25	Supplier Name
3	Address	Varchar	50	Address
4	Phone_no	Varchar	20	Phone number
5	Mobile_No	Varchar	20	Mobile Number

Table Name : Material Master

Primary Key : matCode

Table Description : This table is used to maintain the details about Material Master

S.NO	Field Name	Data Type	Size	Description
1	matCode	Int	10	Material Code
2	matName	Varchar	20	Material Name
3	Category	Varchar	10	Category
4	UOM	Int	20	Unit of Measurement

Table Name : tblorder

Primary Key : orderNo

Foreign Key : cusId

Table Description : Stores Building Order Details

S.NO	Field Name	Data Type	Size	Description
1	orderNo	Int	10	Order Number
2	orderDate	Date/Time	8	Order Date
3	QuotNo	Int	10	Quotation Number
4	cusId	Int	10	Customer id
5	cusName	Varchar	20	Customer Name
6	Sitename	Varchar	25	Site Name
7	Plot_number	Int	10	Plot Number
8	LandSqft	Varchar	25	Total Sqft for Land
9	BuildingSqlft	Varchar	25	Total Sqlft for Building Area
10	No_of_room	Int	10	Total number of room
11	Roomdetails	Varchar	50	Room Details
12	FlooringDet	Varchar	50	FlooringDetails
13	Additionalfacl	Varchar	50	Additional Facility
14	Kitchendet	Varchar	50	Kitchen details
15	Toiletdet	Varchar	50	Toilet Details
16	OtherDet	Varchar	50	Other Requirements

Table Name **:** tblplanning

Primary Key **:** PlanningID

Foreign Key **:** OrderID

Table Description **:** This table is used to maintain the details about planning

S.NO	**Field Name**	**Data Type**	**Size**	**Description**
1	PlanningID	Int	10	Planning ID
2	PlanningDate	Date/Time	8	Planning Date
3	OrderID	Int	10	Order ID
4	cusId	Int	10	Customer id
5	TotalSqft	Varchar	25	Total Sqft for Building
6	BuildingDet	Varchar	25	Building Details
7	Sup_name	Varchar	25	Supervisor name
8	MaterialsNeed	Varchar	50	Material Requiremetns
9	Daysneeed	Varchar	50	Total days need
10	Othedesc	Varchar	50	Other description

Table Name : tblPurchase

Primary Key : Purhcase_ID

Foreign Key : Supid

Table Description : This table is used to maintain the details about purchase

S.N0	**Field Name**	**Data Type**	**Size**	**Description**
1	PurchaseID	Int	10	Purchase ID
2	PurhaseDate	Date/Time	8	Date of Purchase
3	Supid	Int	10	Supplier ID
4	Supplier Name	Varchar	25	Supplier Name
5	matCode	Int	10	Material Code
6	matName	Varchar	10	Material Name
7	UOM	Int	10	Unit of Measurement
8	Rate	Varchar	15	Rate
9	Qty	Int	10	Quantity
10	Amount	Double	10	amount

Table Name **:** tblbilling

Primary Key **:** billNo

Foreign Key **:** cusId

Table Description **:** This table is used to maintain the details about billing

S.NO	**Field Name**	**Data Type**	**Size**	**Description**
1	billNo	Int	10	Bill number
2	billDate	Date/Time	10	Billing Date
3	cusId	Int	10	Customer ID
4	cusName	Varchar	10	CusotmerName
5	Address	Varchar	10	Address
6	Plotnumber	Int	25	Plotnumber
7	Ordered	Int	10	Orderid
8	tsqft	Varchar	25	Total Square Feet of Building
9	Amtpersqft	Int	10	Amount per Sqft
10	Totamt	Double	10	Total Amount
11	Extraamt	Double	10	Additional Amount charged
12	Netamt	Double	10	Net Amount

B. Sample Coding

```
Imports System.Data.SqlClient

Public Class frmBilling

    Sub auto()

        qry = "select max(billid) from tbl_bill"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        If dr.Read Then

            If IsDBNull(dr(0)) Then

                txtbillid.Text = 4000

            Else

                txtbillid.Text = dr(0) + 1

            End If

            dr.Close()

        End If

        dr.Close()

    End Sub

    Sub clear()

        Txtamtsqft.Clear()

        Txtnetamt.Clear()

        txtbillid.Clear()

        txtCustomerName.Clear()

        Txtextraamt.Clear()

        Txtnetamt.Clear()

        Txttotal.Clear()

        Txttotsqft.Clear()
        Txtcusid.Clear()
        Cmborderid.Text = ""
    End Sub
    Sub order()
        qry = "Select * from tbl_border"
        cmd = New SqlCommand(qry, con)
        Dim dr As SqlDataReader
        dr = cmd.ExecuteReader
        Cmborderid.Items.Clear()
        While dr.Read
            Cmborderid.Items.Add(dr("oid"))
        End While
        dr.Close()
    End Sub

    Private Sub frmBilling_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        If con.State = ConnectionState.Closed Then
            con.ConnectionString = constr
            con.Open()
        End If
        auto()
        order()
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        clear()
        auto()
        Cmborderid.Focus()
    End Sub


    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
        Try
            If Txtamtsqft.Text = "" Or Cmborderid.Text = "" Or txtCustomerName.Text = "" Or txtbillid.Text =
"" Or Txtcusid.Text = "" Or Txtextraamt.Text = "" Or Txtnetamt.Text = "" Or Txttotal.Text = "" Or
Txttotsqft.Text = "" Then
                MsgBox("Please Fill All The Details")
            Else
                qry = "insert into tbl_bill values('" & txtbillid.Text & "','" & DateTimePicker1.Text & "','" &
Cmborderid.Text & "','" & Txtcusid.Text & "','" & txtCustomerName.Text & "','" & Txttotsqft.Text & "','" &
Txtamtsqft.Text & "','" & Txttotal.Text & "','" & Txtextraamt.Text & "','" & Txtnetamt.Text & "')"
                cmd = New SqlCommand(qry, con)
                Dim i As Integer = cmd.ExecuteNonQuery()
                If i > 0 Then
                    MsgBox("Sucessfully Inserted")
                    clear()
                    auto()
                Else
                    MsgBox("Cannot Insert ")
                End If
            End If
        Catch ex As Exception


        End Try
    End Sub


    Private Sub Cmborderid_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Cmborderid.SelectedIndexChanged
        qry = "Select * from tbl_border where oid='" & Cmborderid.Text & "'"
        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        If dr.Read = True Then

            txtCustomerName.Text = dr("cname")

            Txtcusid.Text = dr("cid")

            Txttotsqft.Text = dr("bsqft")

        End If

        dr.Close()

    End Sub


    Private Sub Txtamtsqft_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Txtamtsqft.TextChanged

        Txttotal.Text = Val(Val(Txttotsqft.Text) * Val(Txtamtsqft.Text))

    End Sub


    Private Sub Txtextraamt_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Txtextraamt.TextChanged

        Txtnetamt.Text = Val(Val(Txttotal.Text) + Val(Txtextraamt.Text))

    End Sub


    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button5.Click

        Me.Close()

    End Sub


    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button3.Click

        qry = "delete from  tbl_bill where billid='" & txtbillid.Text & "'"

        cmd = New SqlCommand(qry, con)

        Dim i As Integer = cmd.ExecuteNonQuery()
        If i > 0 Then
            MsgBox("Sucessfully Deleted")
            clear()
            auto()
        Else
            MsgBox("Cannot Delete ")
        End If
    End Sub
End Class
Imports System.Data.SqlClient
Public Class frmBuildingOrder

    Private Sub frmBuildingOrder_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        If con.State = ConnectionState.Closed Then
            con.ConnectionString = constr
            con.Open()
        End If
        qry = "Select * from tblcustomer"
        cmd = New SqlCommand(qry, con)
        Dim dr As SqlDataReader
        dr = cmd.ExecuteReader
        While dr.Read
            txtcid.Items.Add(dr(0))
        End While
        dr.Close()
        Call autono()
        txtoid.Text = autono()
    End Sub
    Private Function autono() As Integer
        qry = "select max(oid) from tbl_border"
        cmd = New SqlCommand(qry, con)
        Dim dr As SqlDataReader
        dr = cmd.ExecuteReader
        If dr.Read Then
            autono = dr(0) + 1
        Else
            autono = 1000
        End If
        dr.Close()
    End Function


    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
        qry = "insert into tbl_border values(" & txtoid.Text & ",'" & odate.Text & "'," & txtcid.Text & ",'" & txtcname.Text & "','" & txtaddr.Text & "','" & txtsname.Text & "','" & txtpno.Text & "'," & txttsqft.Text & "," & txtbsqft.Text & "," & txtroom.Text & ",'" & txtrdetails.Text & "','" & txtfdetails.Text & "','" & txtadetails.Text & "','" & txtkdetails.Text & "','" & txttdetails.Text & "')"
        cmd = New SqlCommand(qry, con)
        cmd.ExecuteNonQuery()
        MsgBox("Record Inserted..")
    End Sub


    Private Sub txtcid_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtcid.SelectedIndexChanged
        qry = "Select * from tblcustomer where customerid=" & CInt(txtcid.Text)
        cmd = New SqlCommand(qry, con)
        Dim dr As SqlDataReader
        dr = cmd.ExecuteReader
        If dr.Read Then
            txtcname.Text = dr(1)
            txtaddr.Text = dr(2)
        End If
        dr.Close()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
        Dim oid As Integer
        oid = InputBox("Enter Your Oider Id")
        qry = "delete from tbl_border where oid=" & oid
        cmd = New SqlCommand(qry, con)
        cmd.ExecuteNonQuery()
        MsgBox("Record Deleted..")
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button5.Click
        Me.Close()
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        Call fclear()
    End Sub
    Private Sub fclear()
        txtoid.Text = ""
        txtcname.Text = ""
        txtaddr.Text = ""
        txtsname.Text = ""

        txtpno.Text = ""

        txttsqft.Text = ""

        txtbsqft.Text = ""

        txtroom.Text = ""

        txtrdetails.Text = ""

        txtfdetails.Text = ""

        txtadetails.Text = ""

        txtkdetails.Text = ""

        txttdetails.Text = ""

        autono()

    End Sub

End Class


Imports System.Data.SqlClient

Public Class frmPlanning

    Sub auto()

        qry = "select max(planid) from tbl_planning"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        If dr.Read Then

            If IsDBNull(dr(0)) Then

                txtMaterialID.Text = 2000

            Else

                txtMaterialID.Text = dr(0) + 1

            End If

            dr.Close()

        End If
        dr.Close()

    End Sub

    Sub clear()

        Txtbuildetails.Clear()

        Txtbuildsqft.Clear()

        txtCustomerName.Clear()

        Txtday.Clear()

        Txtlandft.Clear()

        txtMaterialID.Clear()

        Txtmatreq.Clear()

        cmbCategory.Text = ""

    End Sub


    Sub order()

        qry = "Select * from tbl_border"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        cmbCategory.Items.Clear()

        While dr.Read

            cmbCategory.Items.Add(dr("oid"))

        End While

        dr.Close()

    End Sub

    Private Sub frmPlanning_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load

        If con.State = ConnectionState.Closed Then

            con.ConnectionString = constr

            con.Open()
        End If

        auto()

        order()

    End Sub


    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button5.Click

        Me.Close()

    End Sub


    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click

        clear()

        cmbCategory.Focus()

        auto()

    End Sub


    Private Sub cmbCategory_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmbCategory.SelectedIndexChanged

        qry = "Select * from tbl_border where oid='" & cmbCategory.Text & "'"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        If dr.Read = True Then

            txtCustomerName.Text = dr("cname")

            Txtlandft.Text = dr("tlsqft")

            Txtbuildsqft.Text = dr("bsqft")

        End If

        dr.Close()

    End Sub
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
        Try
            If txtMaterialID.Text = "" Or cmbCategory.Text = "" Or txtCustomerName.Text = "" Or Txtbuildsqft.Text = "" Or Txtlandft.Text = "" Then
                MsgBox("Please Fill All The Details")
            Else
                qry = "insert into tbl_planning values('" & txtMaterialID.Text & "','" & DateTimePicker1.Text & "','" & cmbCategory.Text & "','" & txtCustomerName.Text & "','" & Txtlandft.Text & "','" & Txtbuildsqft.Text & "','" & Txtbuildetails.Text & "','" & Txtmatreq.Text & "','" & Txtday.Text & "')"
                cmd = New SqlCommand(qry, con)
                Dim i As Integer = cmd.ExecuteNonQuery()
                If i > 0 Then
                    MsgBox("Sucessfully Inserted")
                    clear()
                    auto()
                Else
                    MsgBox("Cannot Insert ")
                End If
            End If
        Catch ex As Exception

        End Try
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
        qry = "delete  from tbl_planning where planid='" & txtMaterialID.Text & "' "
        cmd = New SqlCommand(qry, con)
        Dim i As Integer = cmd.ExecuteNonQuery()
        If i > 0 Then
            MsgBox("Sucessfully Deleted")
            clear()
            auto()
        Else
            MsgBox("Please Give Proper Planning ID")
        End If
    End Sub
End Class
Imports System.Data.SqlClient
Public Class frmPurchaseMaterial
    Sub auto()
        qry = "select max(purid) from tbl_purchase"
        cmd = New SqlCommand(qry, con)
        Dim dr As SqlDataReader
        dr = cmd.ExecuteReader
        If dr.Read Then
            If IsDBNull(dr(0)) Then
                txtpurlID.Text = 3000
            Else
                txtpurlID.Text = dr(0) + 1
            End If
            dr.Close()
        End If
        dr.Close()
    End Sub
    Sub clear()
        Txtmatname.Clear()
        Txtnetamt.Clear()
        txtpurlID.Clear()

        Txtqty.Clear()

        Txtrate.Clear()

        txtsuppname.Clear()

        Txttax.Clear()

        Txttotamount.Clear()

        Txtuom.Clear()

        Cmbmatid.Text = ""

        cmbsupp.Text = ""

    End Sub

    Sub supplier()

        qry = "Select * from tblSupplier"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        cmbsupp.Items.Clear()

        While dr.Read

            cmbsupp.Items.Add(dr("Supplierid"))

        End While

        dr.Close()

    End Sub

    Sub material()

        qry = "Select * from tblMaterial"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        Cmbmatid.Items.Clear()

        While dr.Read

            Cmbmatid.Items.Add(dr("MaterialID"))
        End While

        dr.Close()

    End Sub

    Private Sub frmPurchaseMaterial_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

        If con.State = ConnectionState.Closed Then

            con.ConnectionString = constr

            con.Open()

        End If

        auto()

        supplier()

        material()

    End Sub


    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

        clear()

        auto()

    End Sub


    Private Sub cmbsupp_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmbsupp.SelectedIndexChanged

        qry = "Select * from tblSupplier where Supplierid='" & cmbsupp.Text & "'"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        If dr.Read = True Then

            txtsuppname.Text = dr("SupplierName")

        End If

        dr.Close()
    End Sub


    Private Sub Cmbmatid_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Cmbmatid.SelectedIndexChanged

        qry = "Select * from tblMaterial where MaterialID='" & Cmbmatid.Text & "'"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        If dr.Read = True Then

            Txtmatname.Text = dr("MaterialName")

            Txtuom.Text = dr("UOM")

        End If

        dr.Close()

    End Sub


    Private Sub Txtqty_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Txtqty.TextChanged

        Txttotamount.Text = Val(Val(Txtrate.Text) * Val(Txtqty.Text))

        Txttax.Text = Val(Val(Txttotamount.Text) * (0.04))

        Txtnetamt.Text = Val(Val(Txttotamount.Text) + Val(Txttax.Text))

    End Sub


    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click

        qry = "insert into tbl_purchase values('" & txtpurlID.Text & "','" & DateTimePicker1.Text & "','" &
cmbsupp.Text & "','" & txtsuppname.Text & "','" & Cmbmatid.Text & "','" & Txtmatname.Text & "','" &
Txtuom.Text & "','" & Txtrate.Text & "','" & Txtqty.Text & "','" & Txttotamount.Text & "','" & Txttax.Text
& "','" & Txtnetamt.Text & "') "

        cmd = New SqlCommand(qry, con)

        Dim i As Integer = cmd.ExecuteNonQuery

        If i > 0 Then
```

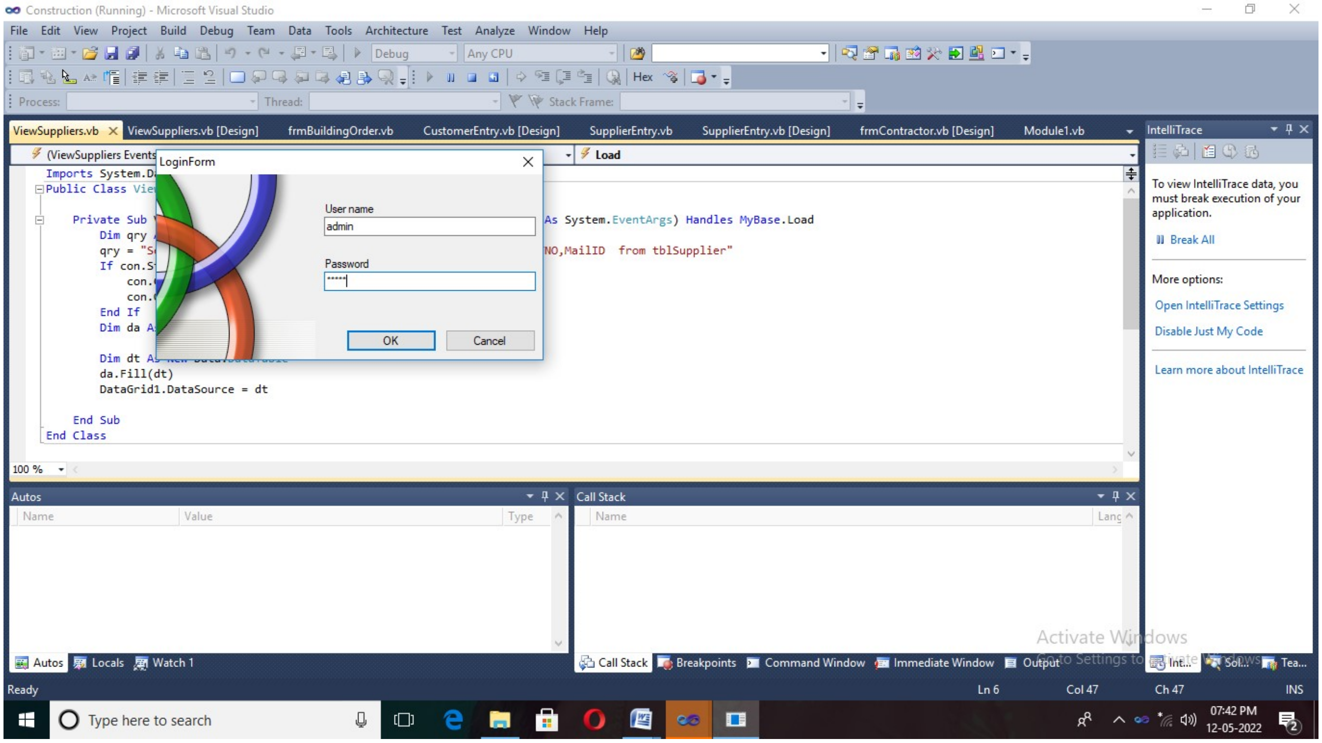
```
            MsgBox("Sucessfully Inserted")
            clear()
            auto()
        Else
            MsgBox("Cannot Insert")
        End If
    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button5.Click
        Me.Close()
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
        qry = "delete from tbl_purchase where purid='" & txtpurlID.Text & "'"
        cmd = New SqlCommand(qry, con)
        Dim i As Integer = cmd.ExecuteNonQuery
        If i > 0 Then
            MsgBox("Sucessfully Deleted")
            clear()
            auto()
        Else
            MsgBox("Cannot Delete")
        End If
    End Sub
End Class
Imports System.Data.SqlClient
Public Class MaterialEntry
```

```
    Private Sub MaterialEntry_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load

        If con.State = ConnectionState.Closed Then

            con.ConnectionString = constr

            con.Open()

        End If

        txtMaterialID.Text = autono()

    End Sub

    Private Function autono() As Integer

        qry = "select max(MaterialID) from tblMaterial"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader

        If dr.Read Then

            autono = dr(0) + 1


        Else

            autono = 1000

        End If

        dr.Close()

    End Function


    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click

        qry = "insert into tblMaterial values(" & txtMaterialID.Text & ",'" & txtMaterialName.Text & "','" &
cmbCategory.SelectedItem & "','" & cmbUOM.SelectedItem & "')"

        cmd = New SqlCommand(qry, con)

        cmd.ExecuteNonQuery()

        MsgBox("Record Inserted..")
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
        txtMaterialName.Text = ""
        cmbCategory.Text = ""
        cmbUOM.Text = ""
        autono()
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

    End Sub

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button5.Click
        Me.Hide()
    End Sub
End Class
Imports System.Data.SqlClient
Public Class SupplierEntry

    Private Sub SupplierEntry_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
        If con.State = ConnectionState.Closed Then
            con.ConnectionString = constr
            con.Open()
        End If
        txtSupplierID.Text = autono()
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click

        txtSupplierID.Text = ""

        txtSupplierName.Text = ""

        txtaddress.Text = ""

        txtmailid.Text = ""

        txtvatno.Text = 0

        txtphone.Text = ""

        txtSupplierID.Text = autono()

        txtSupplierName.Focus()

        autono()


    End Sub


    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click

        qry = "insert into tblSupplier values(" & txtSupplierID.Text & ",'" & txtSupplierName.Text & "','" &
txtaddress.Text & "','" & cmbCountry.SelectedItem & "','" & txtmailid.Text & "','" & txtvatno.Text & "','" &
txtphone.Text & "')"

        cmd = New SqlCommand(qry, con)

        cmd.ExecuteNonQuery()

        MsgBox("Record Inserted..")




    End Sub

    Private Function autono() As Integer

        qry = "select max(SupplierID) from tblSupplier"

        cmd = New SqlCommand(qry, con)

        Dim dr As SqlDataReader

        dr = cmd.ExecuteReader
        If dr.Read Then
            autono = dr(0) + 1

        Else
            autono = 1000
        End If
        dr.Close()
    End Function

    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button5.Click
        Me.Close()
    End Sub
    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button4.Click
        Dim cusid As Integer
        cusid = InputBox("Enter a Supplier ID")
        qry = "Select * from tblSupplier where SupplierID=" & cusid
        cmd = New SqlCommand(qry, con)
        Dim dr As SqlDataReader
        dr = cmd.ExecuteReader
        If dr.Read Then
            txtSupplierID.Text = dr(0)
            txtSupplierName.Text = dr(1)
            txtaddress.Text = dr(2)
            cmbCountry.Text = dr(3)
            txtmailid.Text = dr(4)
            txtvatno.Text = dr(5)
            txtphone.Text = dr(6)
        Else
```

```
            MsgBox("Record Not Found..")
        End If
        dr.Close()
    End Sub
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button3.Click
        If txtSupplierID.Text = "" Then
            MsgBox("Plese Find the record to delete..")
            Exit Sub
        End If
        qry = "delete from tblSupplier where SupplierID=" & txtSupplierID.Text
        cmd = New SqlCommand(qry, con)
        cmd.ExecuteNonQuery()
        MsgBox("Record Deleted..")
    End Sub
    Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button6.Click
        qry = "update tblSupplier set Suppliername='" & txtSupplierName.Text & "',address='" &
txtaddress.Text & "',Suppliertype='" & cmbCountry.SelectedItem & "',MailID='" & txtmailid.Text & "',
VatNo='" & txtvatno.Text & "', PhoneNo='" & txtphone.Text & "' where SupplierID=" & txtSupplierID.Text
        cmd = New SqlCommand(qry, con)
        cmd.ExecuteNonQuery()
        MsgBox("Record Modified..")
    End Sub
    Private Sub GroupBox1_Enter(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
GroupBox1.Enter

    End Sub
End Class
```

D. Sample Input

Login Screen



LoginForm

User name: admin

Password: *****

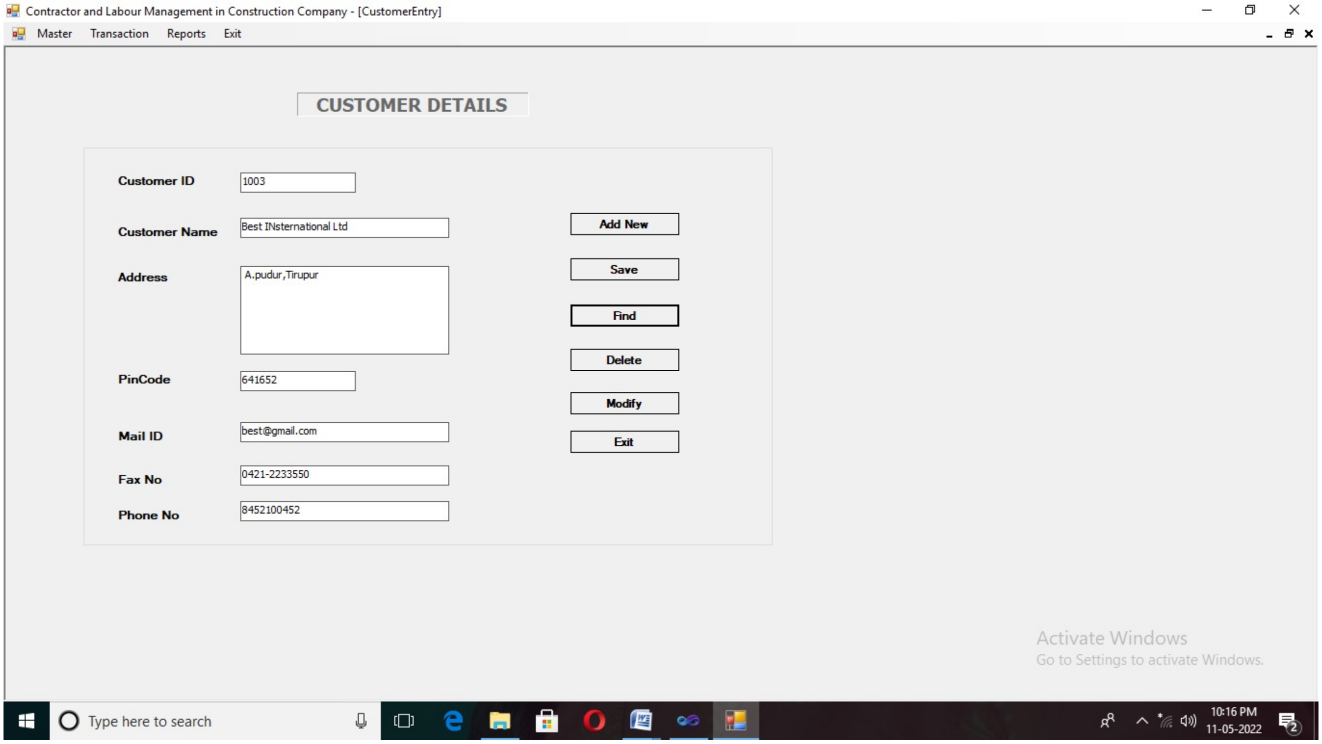
```
Imports System.D
Public Class Vie

    Private Sub                                    As System.EventArgs) Handles MyBase.Load
        Dim qry
        qry = "S                                   NO,MailID  from tblSupplier"
        If con.S
            con.
            con.
        End If
        Dim da A

        Dim dt A
        da.Fill(dt)
        DataGrid1.DataSource = dt

    End Sub
End Class
```

To view IntelliTrace data, you must break execution of your application.

Screen Attributes :

- This form shows the login details.

Customer Details



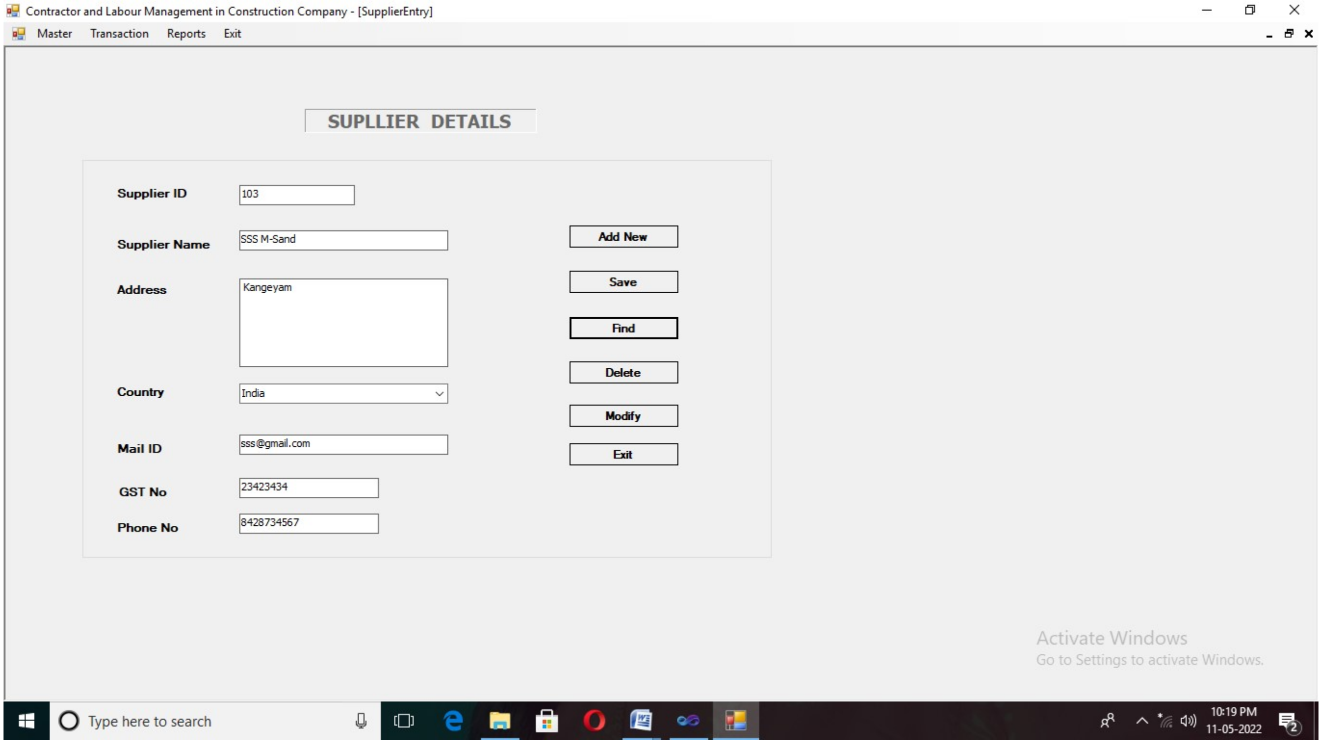
CUSTOMER DETAILS

Customer ID	1003
Customer Name	Best INsternational Ltd
Address	A.pudur,Tirupur
PinCode	641652
Mail ID	best@gmail.com
Fax No	0421-2233550
Phone No	8452100452

Screen Attributes :

- This form shows the coustomer details.

Supplier Details


Contractor and Labour Management in Construction Company - [SupplierEntry]

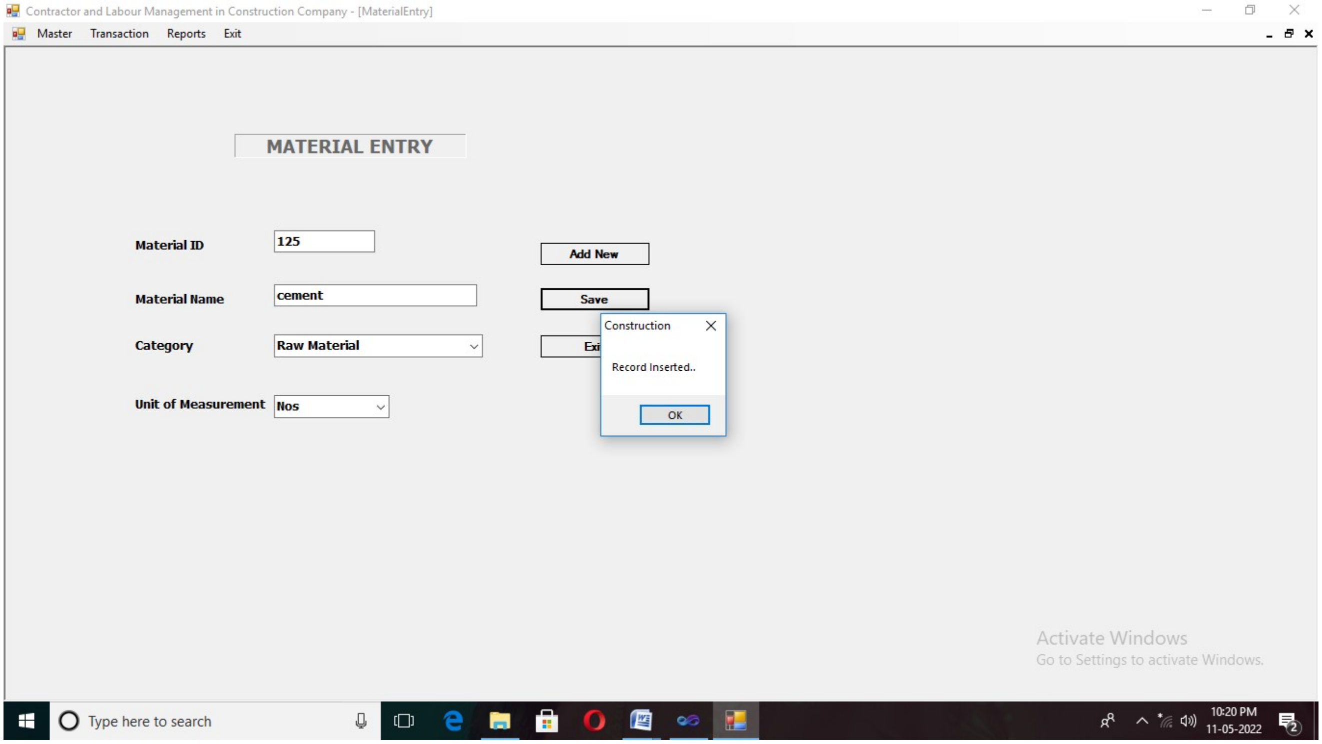
SUPLLIER DETAILS

Supplier ID	103
Supplier Name	SSS M-Sand
Address	Kangeyam
Country	India
Mail ID	sss@gmail.com
GST No	23423434
Phone No	8428734567

Screen Attributes :

- This form shows the supplier details.

Material Entry



MATERIAL ENTRY

Material ID: 125

Material Name: cement

Category: Raw Material

Unit of Measurement: Nos

Construction

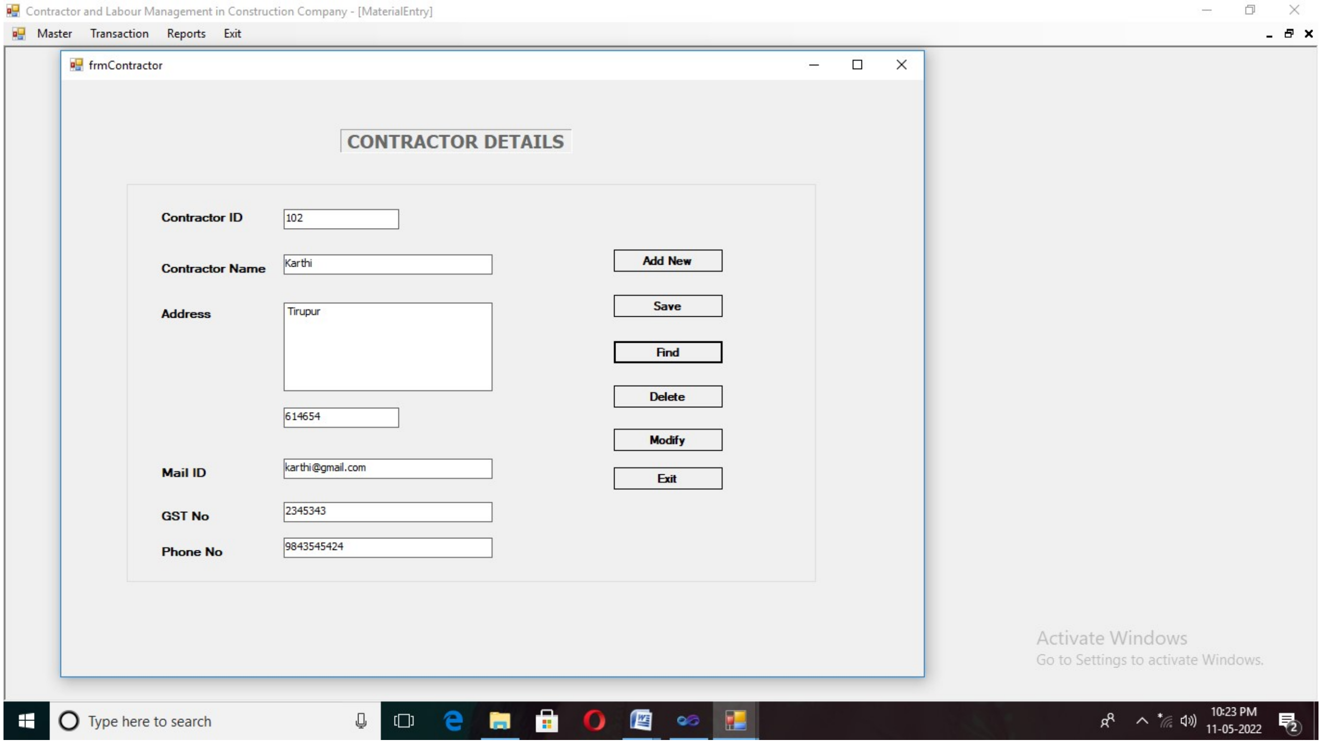
Record Inserted..

Contractor and Labour Management in Construction Company - [MaterialEntry]
Master
Transaction
Reports
Exit
Add New
Save
Exi
OK
Activate Windows
Go to Settings to activate Windows.
Type here to search
10:20 PM
11-05-2022

Screen Attributes :

- This form show the material entry.

Contractor Details

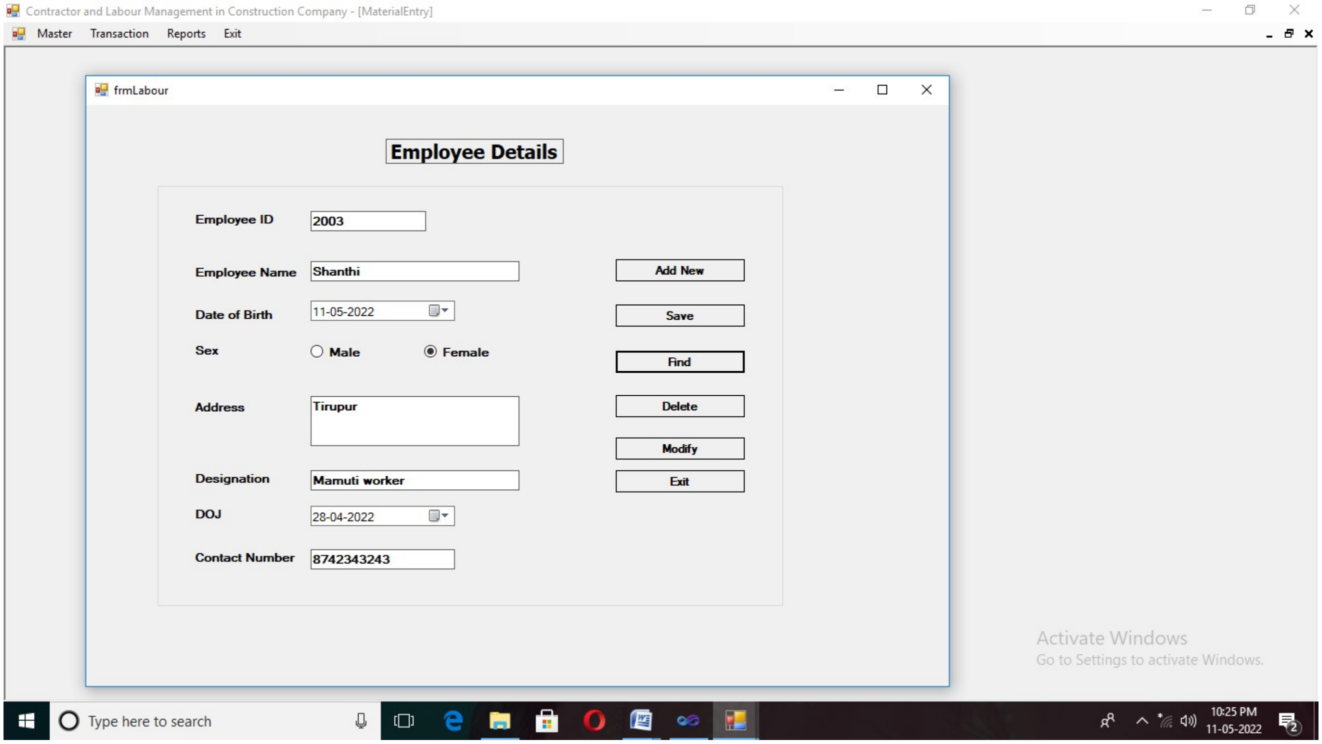


Contractor ID	102
Contractor Name	Karthi
Address	Tirupur
	614654
Mail ID	karthi@gmail.com
GST No	2345343
Phone No	9843545424

Screen Attributes :

- This form shows the contractor details.

Employee Details



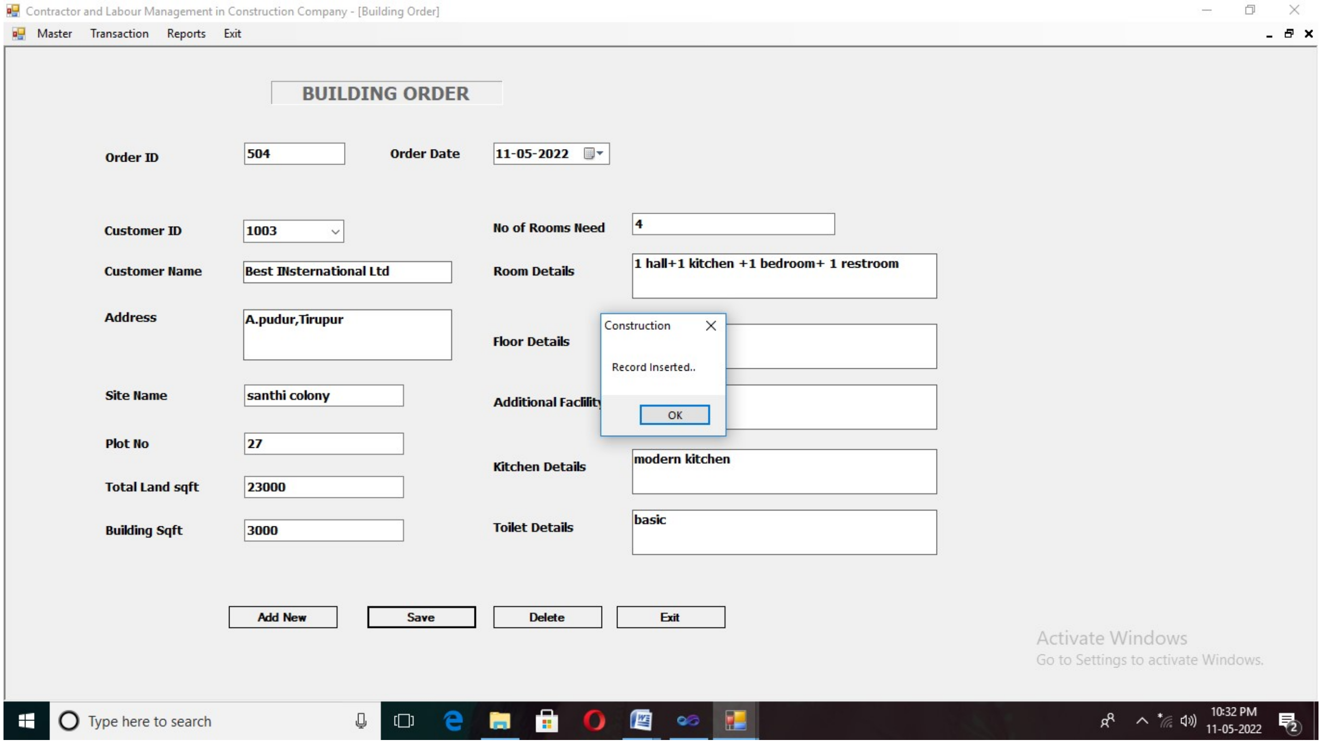
Employee Details

Employee ID	2003
Employee Name	Shanthi
Date of Birth	11-05-2022
Sex	○ Male ● Female
Address	Tirupur
Designation	Mamuti worker
DOJ	28-04-2022
Contact Number	8742343243

Screen Attributes :

- This form show the Employee details.

Building Order


Contractor and Labour Management in Construction Company - [Building Order]
Master
Transaction
Reports
Exit
Add New
Save
Delete
Exit
Activate Windows
Go to Settings to activate Windows.
Type here to search
10:32 PM
11-05-2022

BUILDING ORDER

Order ID	504	Order Date	11-05-2022
Customer ID	1003	No of Rooms Need	4
Customer Name	Best INsternational Ltd	Room Details	1 hall+1 kitchen +1 bedroom+ 1 restroom
Address	A.pudur,Tirupur	Floor Details	
Site Name	santhi colony	Additional Facility	
Plot No	27	Kitchen Details	modern kitchen
Total Land sqft	23000	Toilet Details	basic
Building Sqft	3000		

Construction

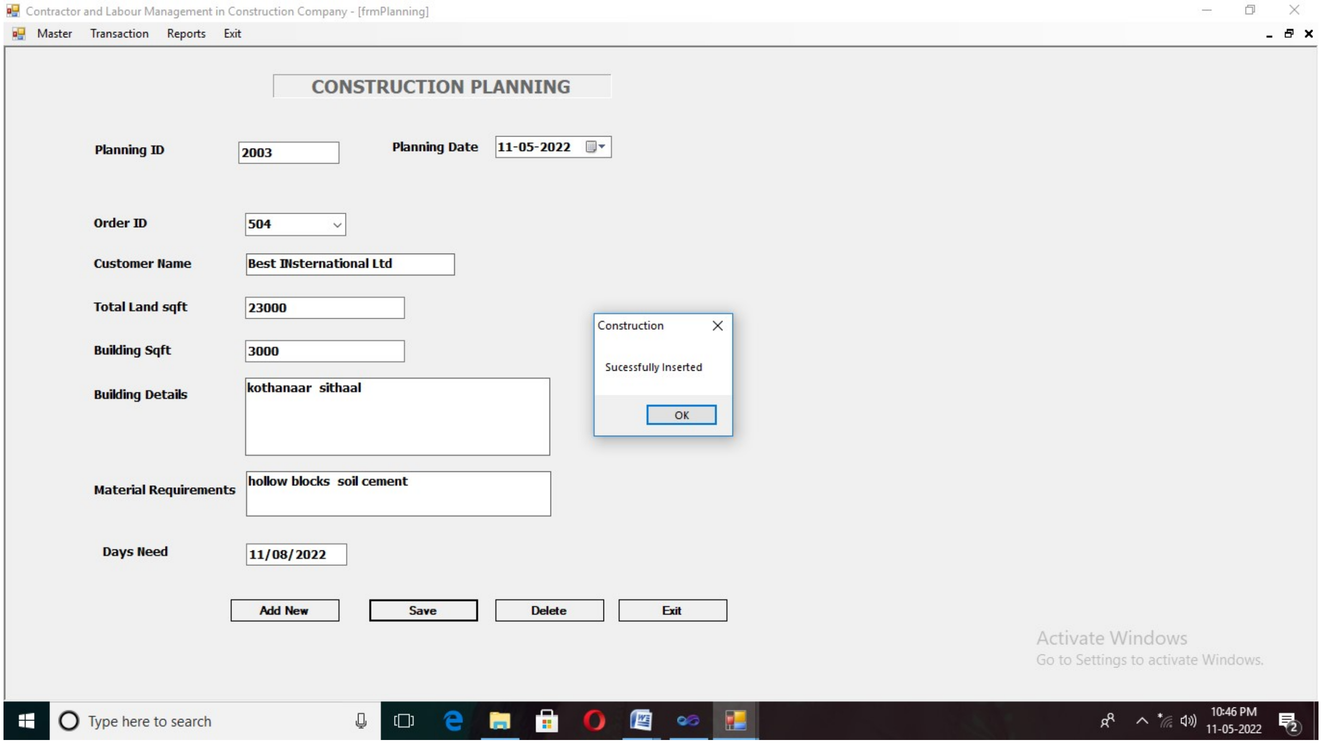
Record Inserted..

OK

Screen Attributes :

- This form show the building orders.

Construction Planning



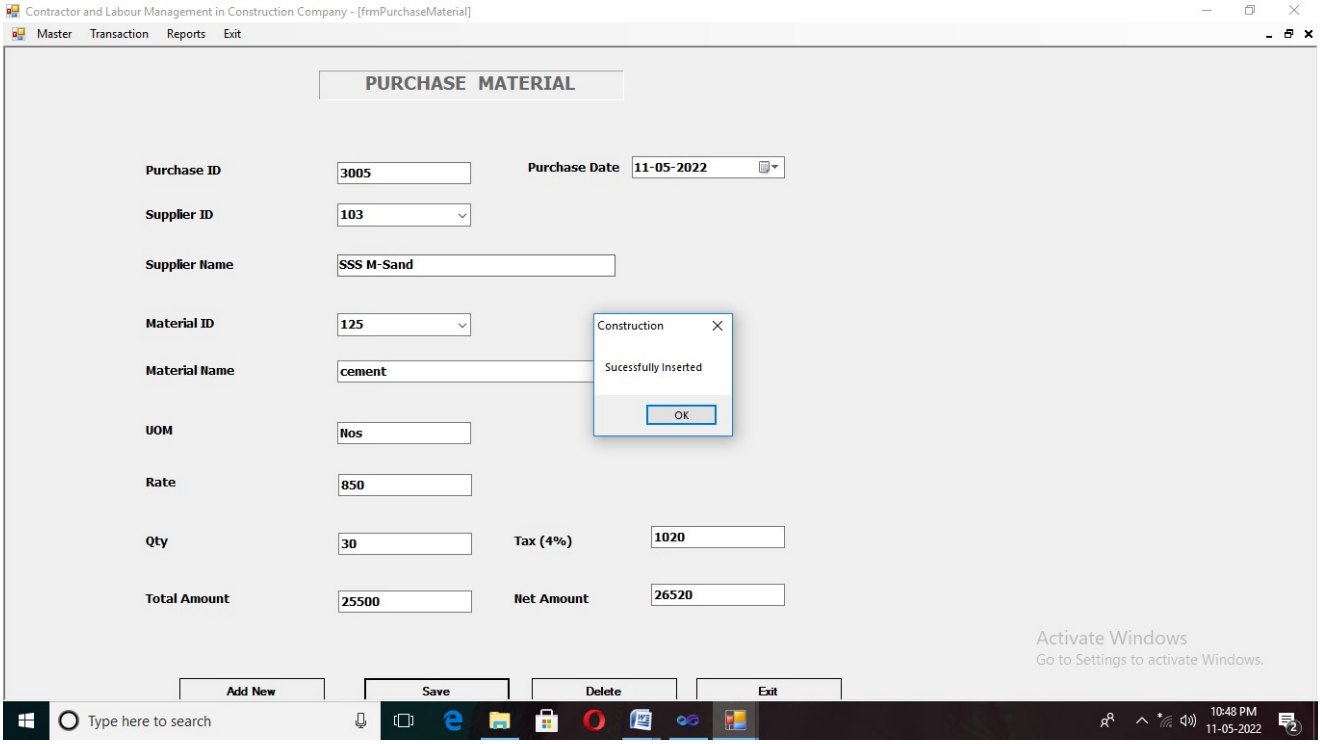
Planning ID	2003
Planning Date	11-05-2022
Order ID	504
Customer Name	Best INsternational Ltd
Total Land sqft	23000
Building Sqft	3000
Building Details	kothanaar sithaal
Material Requirements	hollow blocks soil cement
Days Need	11/08/2022

Sucessfully Inserted

Screen Attributes :

- This form show the construction planning.

Purchase Material



Contractor and Labour Management in Construction Company - [frmPurchaseMaterial]

PURCHASE MATERIAL

Purchase ID	3005	Purchase Date	11-05-2022
Supplier ID	103		
Supplier Name	SSS M-Sand		
Material ID	125		
Material Name	cement		
UOM	Nos		
Rate	850		
Qty	30	Tax (4%)	1020
Total Amount	25500	Net Amount	26520

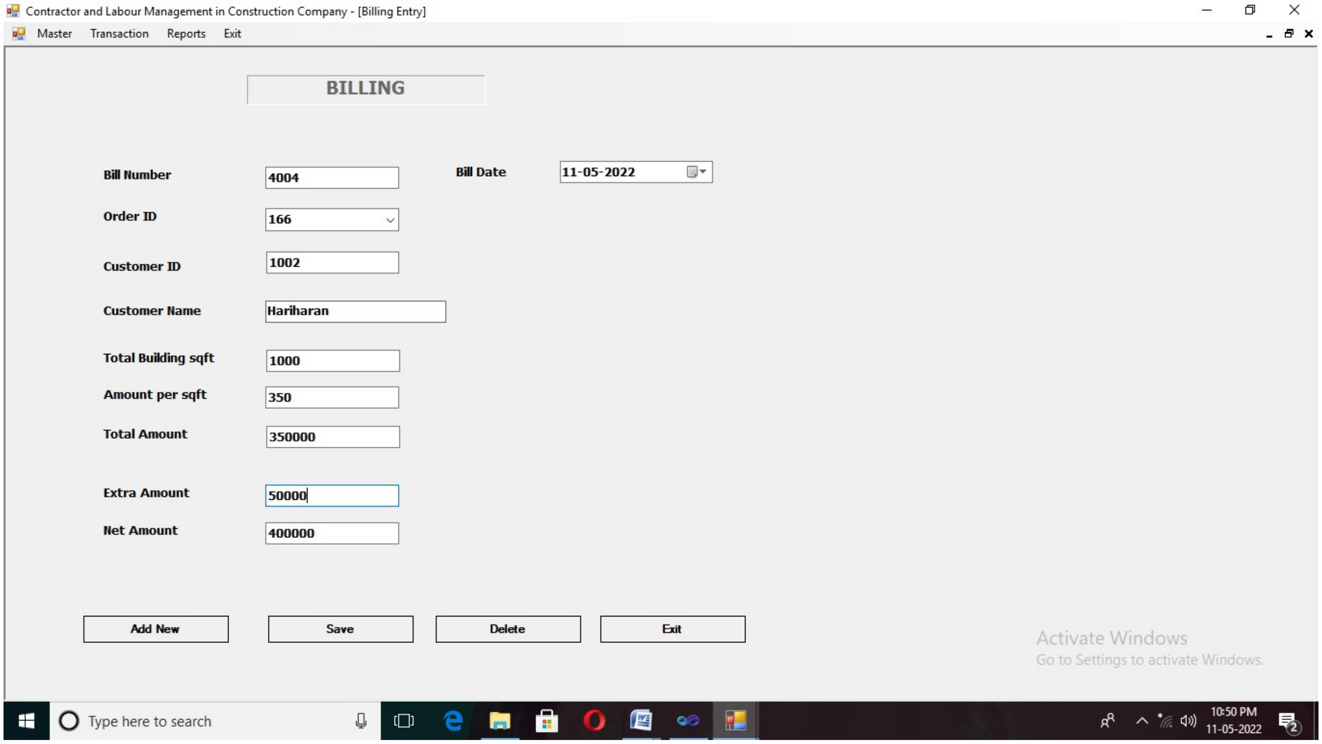
Construction

Sucessfully Inserted

Screen Attributes :

- This form shows the purchase material.

Billing Details


Contractor and Labour Management in Construction Company - [Billing Entry]
Master
Transaction
Reports
Exit
Add New
Save
Delete
Exit
Activate Windows
Go to Settings to activate Windows.
Type here to search
10:50 PM
11-05-2022

BILLING

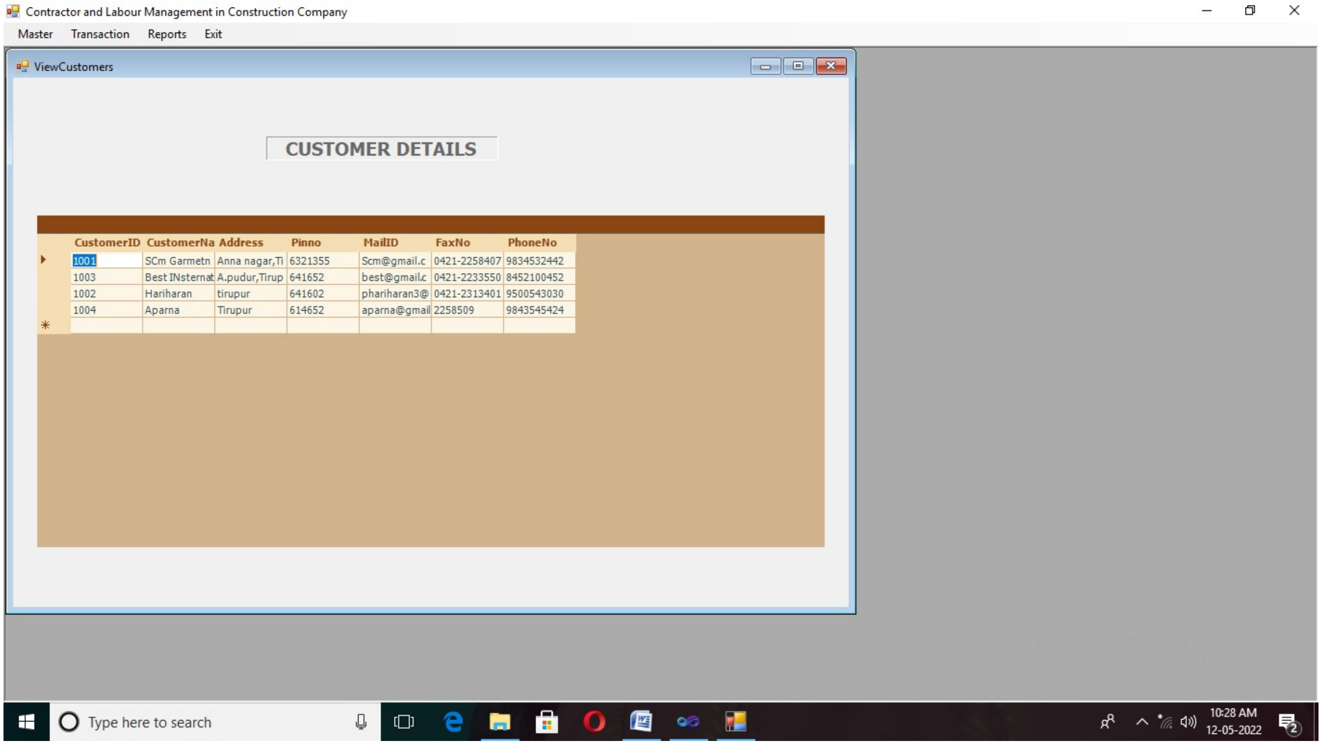
Bill Number	4004	Bill Date	11-05-2022
Order ID	166		
Customer ID	1002		
Customer Name	Hariharan		
Total Building sqft	1000		
Amount per sqft	350		
Total Amount	350000		
Extra Amount	50000		
Net Amount	400000		

Screen attributes :

- This form shows the billing details.

E.SAMPLE OUTPUT

Customer Details Report



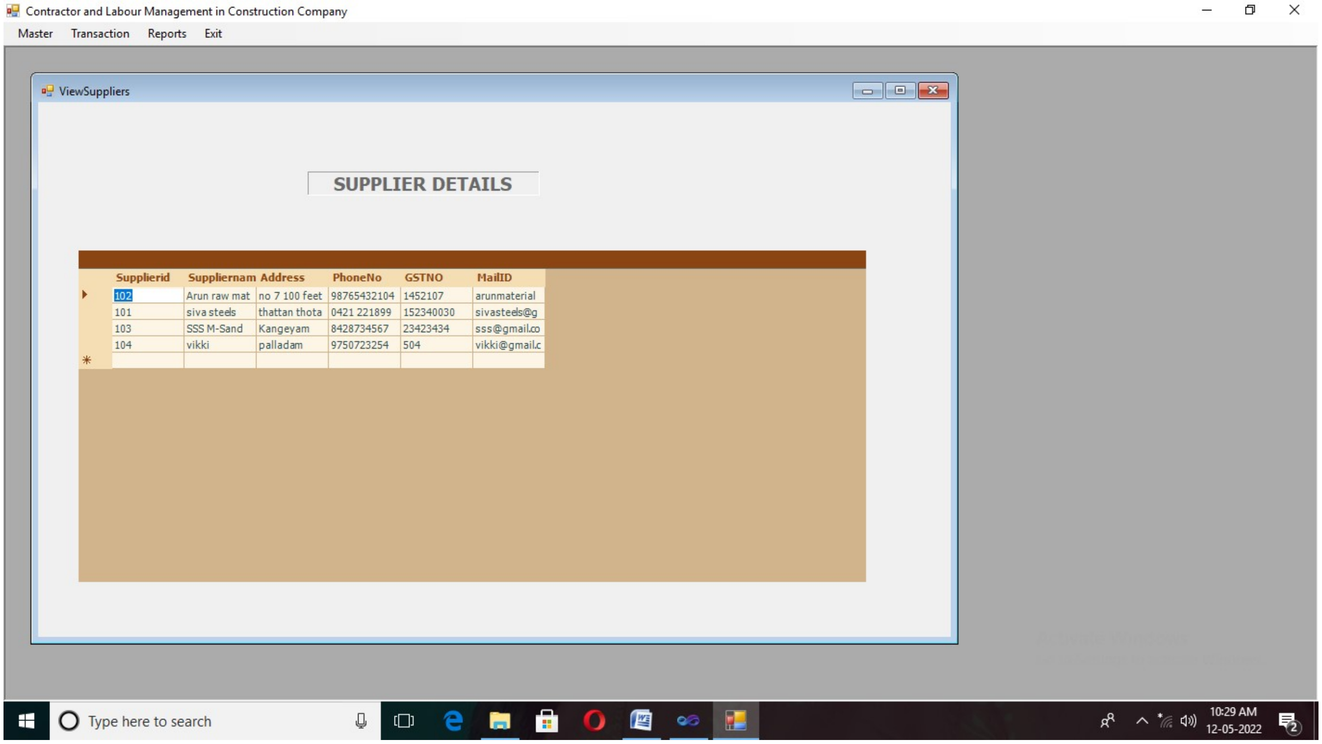
CUSTOMER DETAILS

CustomerID	CustomerNa	Address	Pinno	MailID	FaxNo	PhoneNo
1001	SCm Garmetn	Anna nagar,Ti	6321355	Scm@gmail.c	0421-2258407	9834532442
1003	Best INsternat	A.pudur,Tirup	641652	best@gmail.c	0421-2233550	8452100452
1002	Hariharan	tirupur	641602	phariharan3@	0421-2313401	9500543030
1004	Aparna	Tirupur	614652	aparna@gmail	2258509	9843545424

Screen Attributes :

- This form shows customer details report.

Supplier Details Report

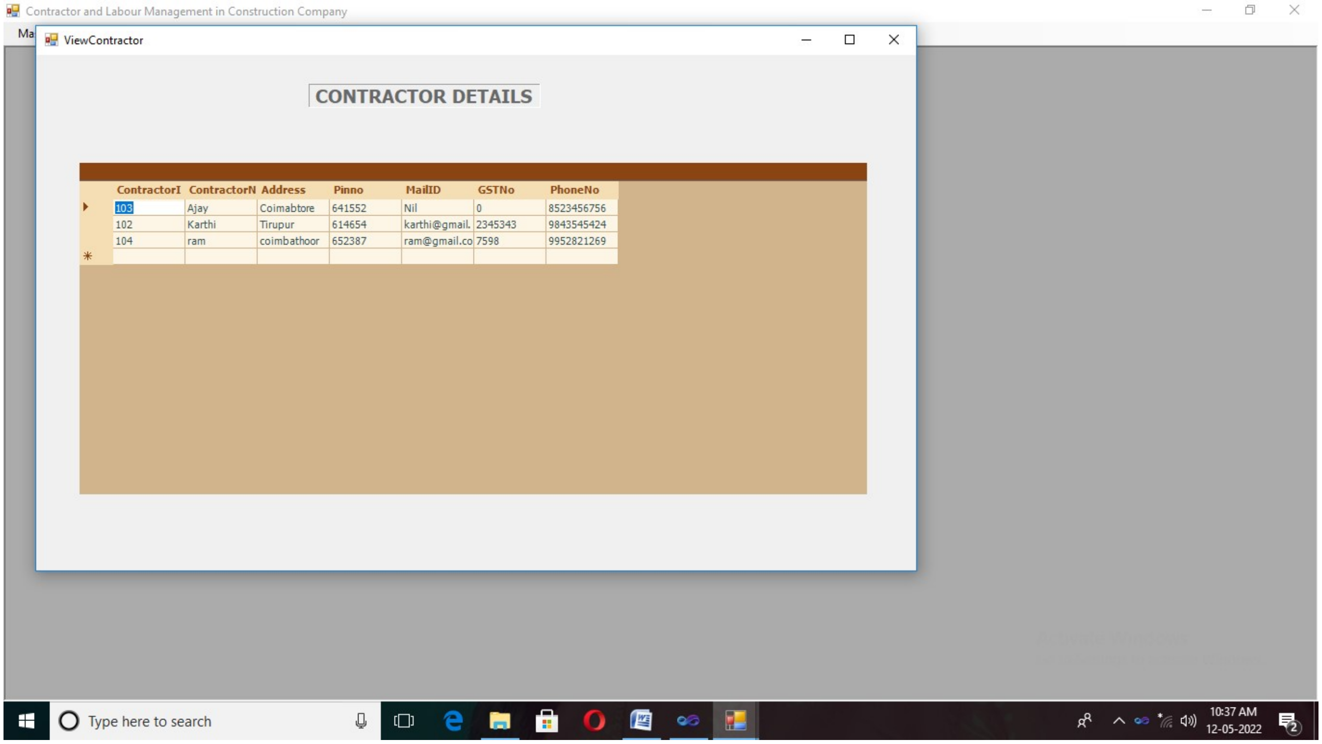


Supplierid	Suppliernam	Address	PhoneNo	GSTNO	MailID
102	Arun raw mat	no 7 100 feet	98765432104	1452107	arunmaterial
101	siva steels	thattan thota	0421 221899	152340030	sivasteels@g
103	SSS M-Sand	Kangeyam	8428734567	23423434	sss@gmail.co
104	vikki	palladam	9750723254	504	vikki@gmail.c

Screen Attributes :

- This form shows supplier details report.

Contractor Details Report



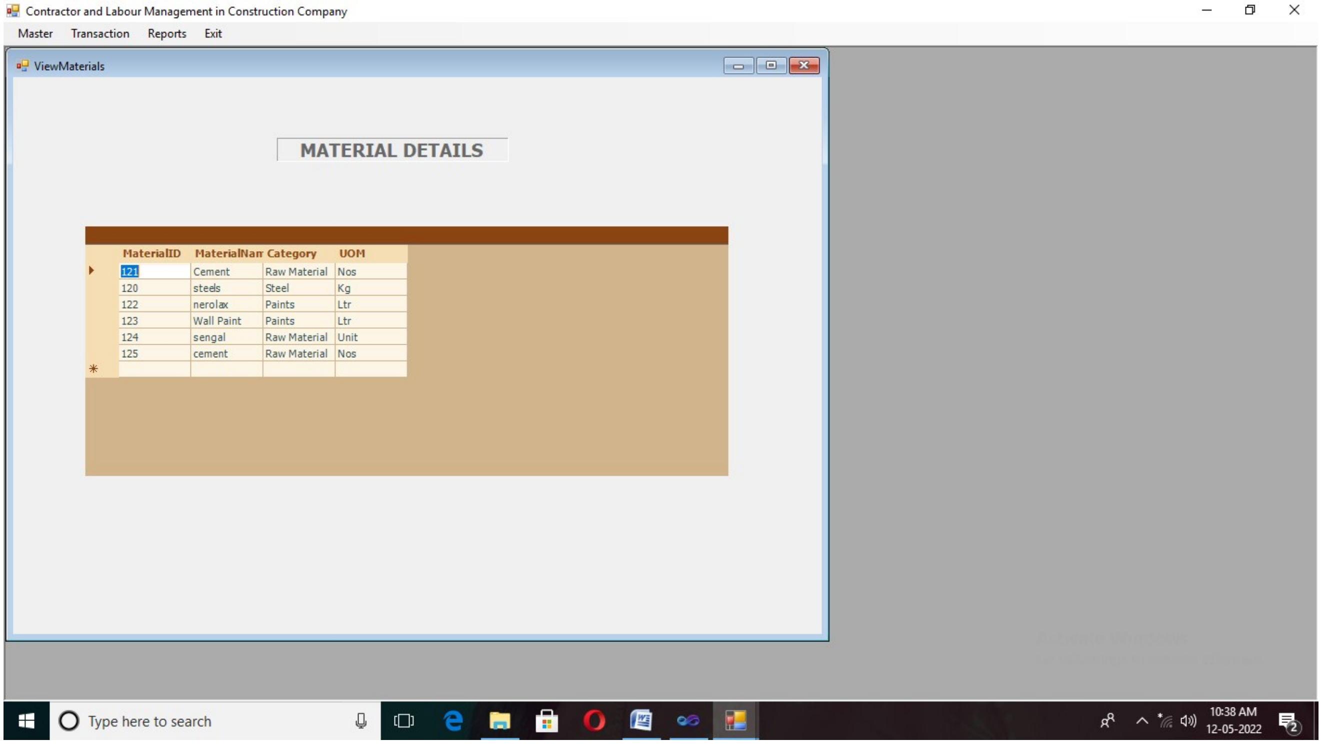
CONTRACTOR DETAILS

ContractorI	ContractorN	Address	Pinno	MailID	GSTNo	PhoneNo
103	Ajay	Coimabtore	641552	Nil	0	8523456756
102	Karthi	Tirupur	614654	karthi@gmail.	2345343	9843545424
104	ram	coimbathoor	652387	ram@gmail.co	7598	9952821269

Screen Attributes :

- This form shows contractor details report.

Material Details Report



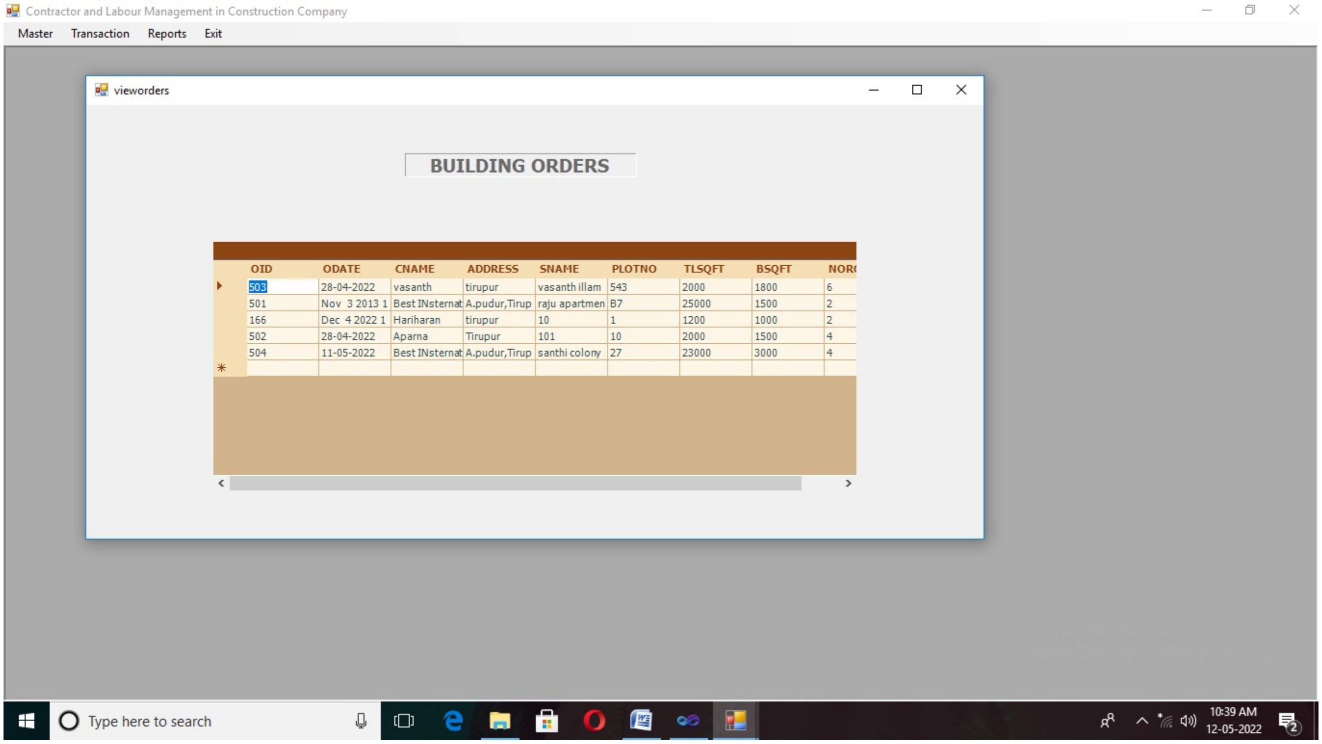
MATERIAL DETAILS

MaterialID	MaterialNam	Category	UOM
121	Cement	Raw Material	Nos
120	steels	Steel	Kg
122	nerolax	Paints	Ltr
123	Wall Paint	Paints	Ltr
124	sengal	Raw Material	Unit
125	cement	Raw Material	Nos

Screen Attributes :

- This form shows material detail report.

Building Orders Report



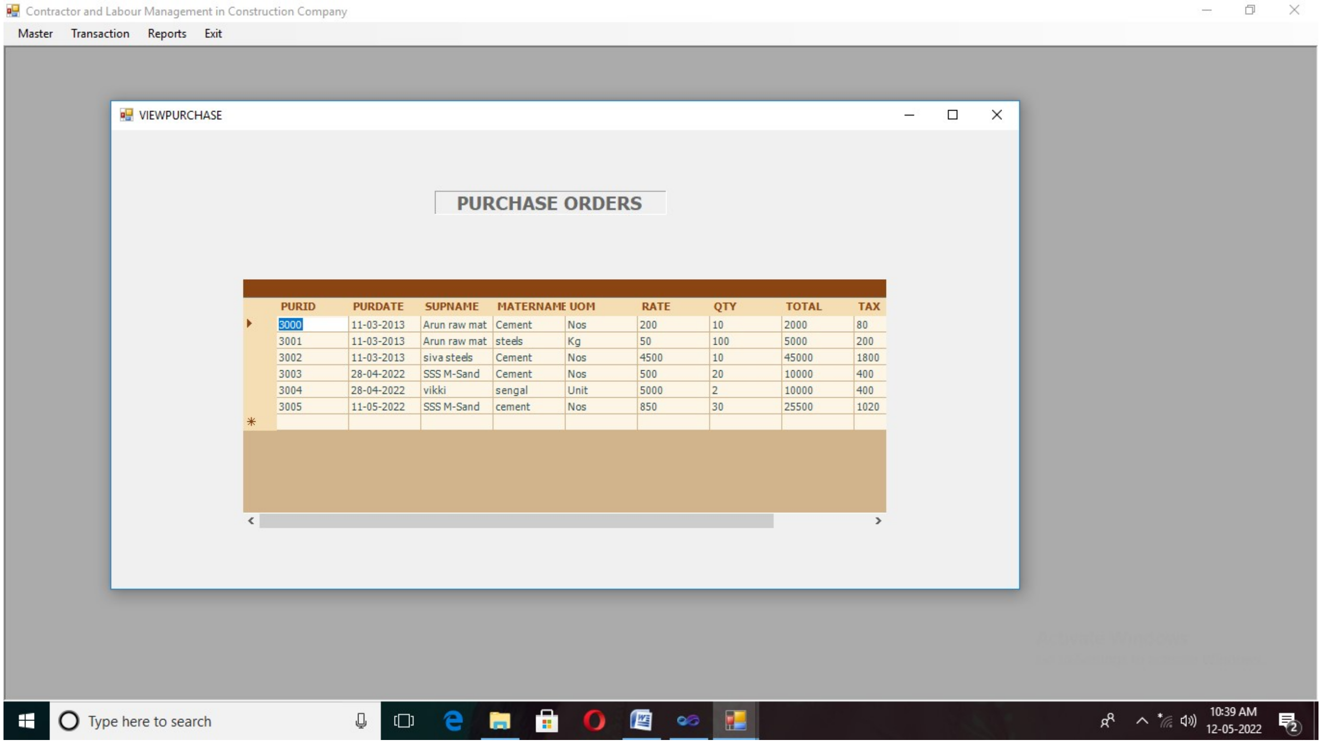
BUILDING ORDERS

OID	ODATE	CNAME	ADDRESS	SNAME	PLOTNO	TLSQFT	BSQFT	NORO
503	28-04-2022	vasanth	tirupur	vasanth illam	543	2000	1800	6
501	Nov 3 2013 1	Best INsternat	A.pudur,Tirup	raju apartmen	B7	25000	1500	2
166	Dec 4 2022 1	Hariharan	tirupur	10	1	1200	1000	2
502	28-04-2022	Aparna	Tirupur	101	10	2000	1500	4
504	11-05-2022	Best INsternat	A.pudur,Tirup	santhi colony	27	23000	3000	4

Screen Attributes :

- This form shows building orders report.

Purchase Orders Report



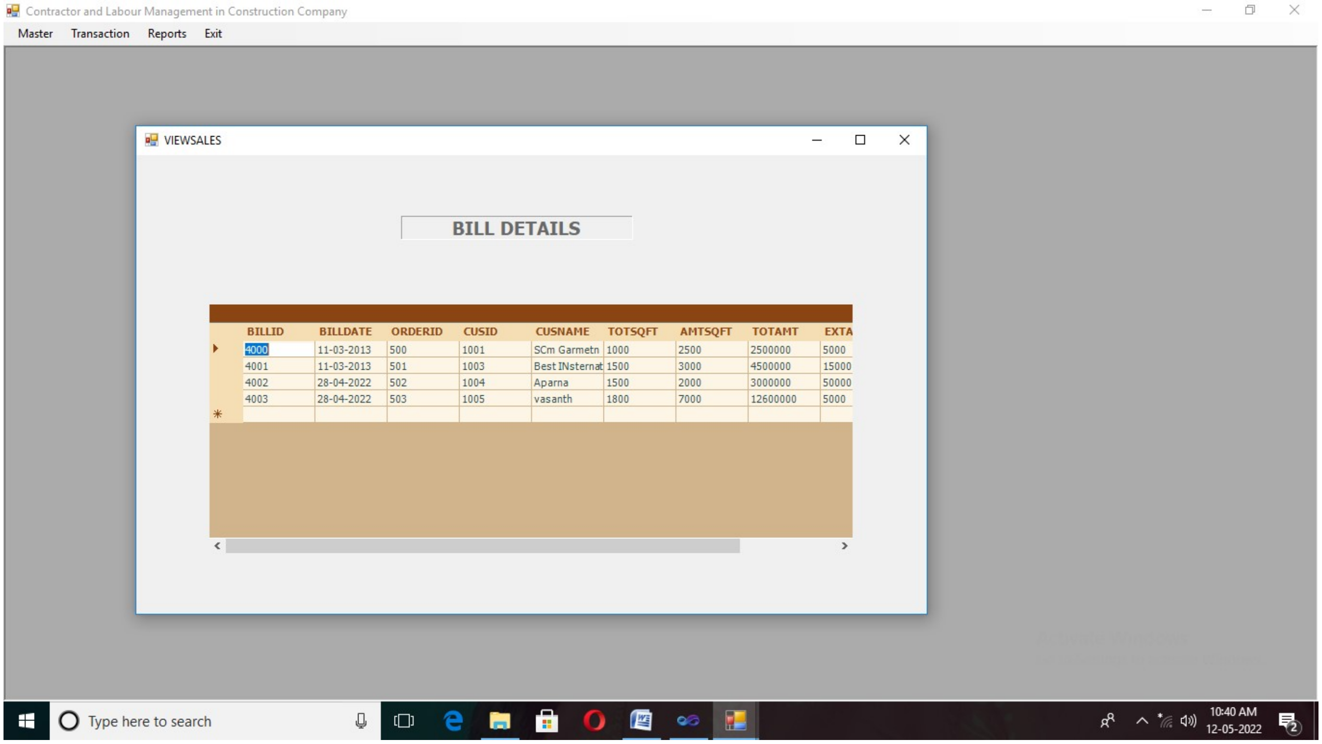
PURCHASE ORDERS

PURID	PURDATE	SUPNAME	MATERNAME	UOM	RATE	QTY	TOTAL	TAX
3000	11-03-2013	Arun raw mat	Cement	Nos	200	10	2000	80
3001	11-03-2013	Arun raw mat	steels	Kg	50	100	5000	200
3002	11-03-2013	siva steels	Cement	Nos	4500	10	45000	1800
3003	28-04-2022	SSS M-Sand	Cement	Nos	500	20	10000	400
3004	28-04-2022	vikki	sengal	Unit	5000	2	10000	400
3005	11-05-2022	SSS M-Sand	cement	Nos	850	30	25500	1020

Screen Attributes :

- This form show purchase order report.

Bill Details Report



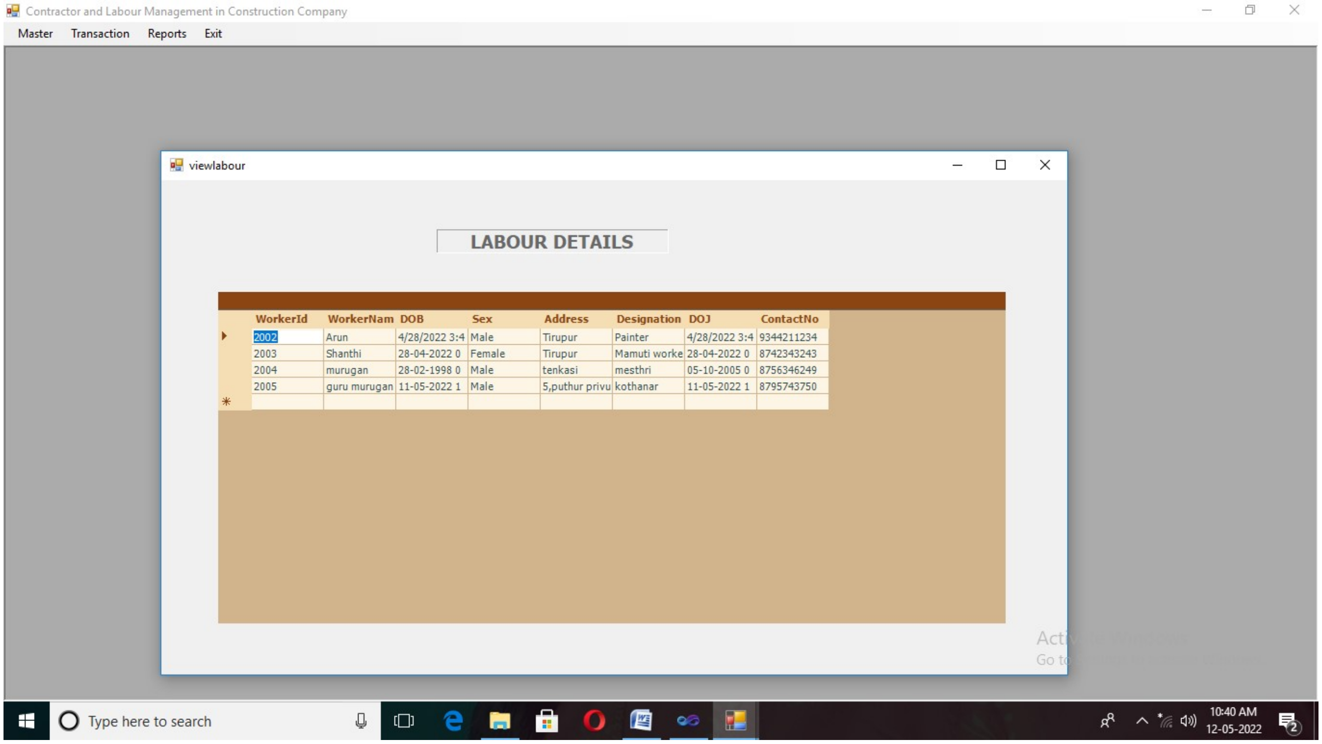
BILL DETAILS

BILLID	BILLDATE	ORDERID	CUSID	CUSNAME	TOTSQFT	AMTSQFT	TOTAMT	EXTA
4000	11-03-2013	500	1001	SCm Garmetn	1000	2500	2500000	5000
4001	11-03-2013	501	1003	Best INsternat	1500	3000	4500000	15000
4002	28-04-2022	502	1004	Aparna	1500	2000	3000000	50000
4003	28-04-2022	503	1005	vasanth	1800	7000	12600000	5000

Screen Attributes :

- This form shows bill details report.

Labour Details Report



LABOUR DETAILS

WorkerId	WorkerNam	DOB	Sex	Address	Designation	DOJ	ContactNo
2002	Arun	4/28/2022 3:4	Male	Tirupur	Painter	4/28/2022 3:4	9344211234
2003	Shanthi	28-04-2022 0	Female	Tirupur	Mamuti worke	28-04-2022 0	8742343243
2004	murugan	28-02-1998 0	Male	tenkasi	mesthri	05-10-2005 0	8756346249
2005	guru murugan	11-05-2022 1	Male	5,puthur privu	kothanar	11-05-2022 1	8795743750

Screen Attributes :

- This form shows labour detail report.