FURNITURE SHOWROOM MANAGEMENT SYSTEM

Submitted By

K.SARANYA (19PCC011)

Under the guidance of

Mrs. V. YASODHA, M.Com(CA)., MBA., M.Phil., NET.,

In the partial fulfilment of the requirements for the award of the degree of

MASTER OF COMMERCE WITH COMPUTER APPLICATIONS

2020-2021

DEPARTMENT OF COMMERCE

AVINASHILINGAM INSTITUTE FOR HOME SCIENCE AND

HIGHER EDUCATION FOR WOMEN

COIMBATORE-641043

CERTIFICATE

CERTIFICATE

This is to certify that the project work entitled "**FURNITURE SHOWROOM MANAGEMENT SYSTEM",** submitted to Department of Commerce, Avinashilingam Institute for Home Science and Higher Education for Woman, Coimbatore, in partial fulfilment of MASTER OF COMMERCE WITH COMPUTER APPLICATIONS, is the record of the original project work done by **K. SARANYA (19PCC011)** during the period of her study, under my supervision and guidance.

Signature of the Supervisor

Signature of the Head of the Department

Submitted for the viva voice examination held on_____

Internal Examiner **External Examiner**

விவேக் பர்னிச்சர்
VIVEK FURNITURE
47-P, KANGEYAM ROAD, NEAR D.S.K. HOSPITAL, TIRUPUR-4.

TIN No. : 33152405803
CST No. : 1160641 / 11-02-2014
Phone : 0421 - 2421759, 4355759
Whatsapp : 99420 44876
Mobile : 98422 41876
E-mail : furniturevivek@gmail.com

TO WHOMSOEVER IT MAY CONCERN

This is to certify that **Ms. SARANYA.K.(REG.NO:19PCC011)** final year **M.com., (COMPUTER APPLICATIONS)** from **DEPARTMENT OF COMMERCE** student of **AVINASHILINGAM INSTITUTE FOR HOME SCIENCE AND HIGHER EDUCATION FOR WOMEN, COIMBATORE – 641043** has successfully completed her project work on the " **FURNITURE MANAGEMENT SYSTEM"** in our concern from December 2020 to April 2021.

During this period we found her to be sincere in work and regular in her attendance (100%). We are certifying that her conduct and character has been found to be Good.

For VIVEK FURNITURE
A. Vivekanandan.

DECLARATION

DECLARATION

I hereby declare that this project work entitled **"FURNITURE SHOWROOM MANAGEMENT SYSTEM"** submitted to Department of Commerce, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, in partial fulfilment of the requirement for the award of the **DEGREE OF MASTER OF COMMERCE WITH COMPUTER APPLICATIONS** is the bonafide record of original project work done by me during the period of my study under the supervision and guidance of **Mrs. V. YASODHA, M.Com(CA)., MBA., M.Phil., NET**., Assistant Professor (Temprorary)Department of Commerce.

Place: Coimbatore

Signature of the candidate

Date:

K.**SARANYA**

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

I would like to express my sincere thanks to **God Almighty,** for his constant love and grace that he has showed upon me, which kept me in good health, and sound mind without which my project would not have reached a successful end.

I would like to express my deep sense of reverential gratitude and sincere thanks to **Prof.S.P. Thyagarajan, D.Sc., Ph.D., M.D., Chancellor**, Avinashilingam Institute for Home Science and Higher Education for Women, for the opportunity given to me for undertaking this study and for providing the needed facility during the course of my study.

I owe my great deal of my gratitude to **Dr.Premavathy Vijayan, M.Sc., M.Ed., Dip.Spl.Edn., M.Phil., Ph.D., Vice Chancellor**, Avinashilingam Institute for Home Science and Higher Education for Women, for extending all resources that facilitated the conduct of the present study.

I express my gratitude to **Dr.S.Kowsalya, M.Sc., M.Phil., Ph.D., Registrar**, Avinashilingam Institute for Home Science and Higher Education for Women, for providing all facilities necessary for the study.

I would express my boundless thanks to **Dr.P.Chitramani, M.B.A., M.Phil., Ph.D., NET., SLET., Dean, School of Commerce and Management**, Avinashilingam Institute for Home Science and Higher Education for Women, for granting the facility required.

I am also thankful to **Dr. (Mrs).D.Geetha, M.Com., Dip.Ed., M.Phil., Ph.D., Professor and Head, Department of Commerce,** Avinashilingam Institute for Home Science and Higher Education for Women, for giving necessary help and support for completing the project successfully.

I express my heartfelt gratitude to my mentor **Mrs. V. YASODHA M.Com(CA)., MBA., M.Phil., NET., Assistant Professor (Temprorary) Department of Commerce**, Avinashilingam Institute for Home Science and Higher Education for Women, for the valuable guidance and for her timely support, suggestions and motivation throughout the project.

I have a great pleasure in expressing my deep sense of my gratitude to all other staff members and non-teaching staffs that helped us to complete the project.

I would extend our thanks to one and all helped us directly and indirectly for successful completion of our project. Last yet importantly, we would like to thank our parents, friends and all our well-wishers for their kind inspiration.

CONTENT

CONTENT

CHAPTER	TITLE	PAGE NO
I	**INTRODUCTION**	
II	**REQUIREMENT ELICITATION** Existing System Proposed System Feasibility of study	
III	**SYSTEM REQUIREMENT SPECIFICATION** Hardware Specification Software Specification Software prototyping Prototyping model	
IV	**SYSTEM DESIGN AND DEVELOPMENT** System design System flow diagram File design Database design Input Design Output Design Data flow Diagram ER Diagram Table Design Code Design Form Design	
V	**SYSTEM TESTING** Integrated testing Acceptance testing	
VI	**SYSTEM IMPLEMENTATION** Implementation procedure User traning	
VII	**CONCULUSION**	
	BIBLIOGRAPHY	

ABSTRACT

ABSTRACT

Vivek furniture Limited, established in the year 2008 and it is located in Tirupur. This well-known establishment acts as a one-stop destination serving customers both local and other parts of Tirupur. It provides various services to its customers ensuring valuable support according to their requirements. The employee ranges around 15 members. The furniture showroom deals with domestic furniture, school furniture, office furniture, etc. The belief that customer satisfaction is important they provide quality product and services .And also they maintain supplier purchase details, stock details, order details and sales details. It is a medium concern here they are maintaining all the details by using MS Access.

As the increasing customers, it is difficult to maintain the records in MS Access itself. So there is a need arises for new system. This newly proposed system is entitled on "FURNITURE SHOWROOM MANAGEMENT SYSTEM'', which is developed using Microsoft VISUAL STUDIO 2012 as front end and SQL SERVER 2008 as back end. Visual Basic.NET comes with enhanced visual designers, increased application performance, and a powerful integrated development environment (IDE). Repository component available in SQL Server Version 7.0 is now called Microsoft SQL Server 2008 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Service.

Through the feasibility study I analysed and found the technical capability, operational ability and economic capability. I used a document called System Requirement Specifications for describe the features and behaviour of a proposed system. In that I cited the details of hardware and software, builds a software prototype by using the incremental prototyping. After the successful completion of prototyping, I created the system design with necessary controls. In the back end application I created the required tables with necessary fields. Using data base connection I connected the front end and back end applications.

In system testing, I choose unit testing, integration testing, acceptance testing and validation testing is to discover the errors and execution. After all the test results are arrived, the proposed system is implemented. The users were trained to use this system.

Training helps the users to add, update and delete the records. To restore, adding and modifying the code to the system is helpful the maintenance. And it may also avoid future problems. The proposed system has been developed with good amount of flexibility without compromising on the response time.

The computerization of the entire system will enhance more accuracy and reduces major part of clerical works and it is fast, clear and legible reports can be generated without any ambiguity. Hence, by developing a system that is user friendly in nature, and many users are able to work on the system with little computer knowledge of training. The project has been developed and the objectives are achieved successfully. It can have an enhancement on proper in the future according to the user's requirements.

The project has been developed with front end as VB.Net and backend as SQL Server. And in future enhancements the frontend can also be changed. ASP.Net can replace the front-end tool such as HTML and CSS for more speed. To provide better facility regarding security, it uses security provider software and we access any place and any where use that.

CHAPTER-I

CHAPTER-I

INTRODUCTION

The project work is entitiled as "FURNITURE SHOWROOM " which sells different types of furniture for home, office, hospital etc., Earlier the showroom use to manufacture selected items but with the increasing response of the customer and their demand they have added items related to office, home, hospitals and others. So it becomes difficult to manage by using MS Access Software using all the order detail, price, employees' details etc. So the window application is created to automate the furniture based system. This includes the information about all products available, their price, orders and employees. This system gives generalize, concise and accurate information regarding billing, purchase details, stock etc.

This system provides any type of enquiry such as product details, stock, purchase details, supplier details, billing etc. This provides the interface of adding, updating, deleting the furniture name, total number of furniture, furniture sales details, customer name, address, contact number. In order details we can specify the customer name, address and which type of furniture.

In delivery details maintains the records when we delivering the product within the particular period. Finally admin maintaining the Monthly sales report it includes furniture sales and amount. The key portions of a Visual Basic project are the forms, controls, and code modules. A form or control has properties that you can alter. The program's code is contained primarily in events mentioned below in various forms in order to keep track of the system. VB.net is used as the front-end tool and for maintaining the database SQL is used as the backend.

OBJECTIVES OF THE PROJECT

- To maintain the highest quality of records.
- To make a new updation from MS Access to Visual basic.Net.
- To reduce the time consumption.
- To exploit the new implementation of technology.

ABOUT THE CONCERN

Vivek furniture Limited, established in the year 2008 and it is located in Tirupur. This well-known establishment acts as a one-stop destination serving customers both local and other parts of Tirupur. It provides various services to its customers ensuring valuable support according to their requirements.The furniture showroom management system deals with the domestic furniture, school furniture, office furniture, machineries, etc. The employee ranges around 15 members. The belief that customer satisfaction is as important as their products and services, have helped this establishment garner a vast base of customers, which continues to grow by the day.

The company is managed by the manager and is under the control of owner with proper surveyance. There are lots of security and has great vision over the customers.They assess the demand or description of need of furniture products in the market and place order to the dealers depending upon its sale.

They also undertake inspection of purchased products , the damaged or unfit product will be return-back to the suppliers. Only after proper inspection, the payment to the suppliers is done. They maintain purchase records ,stock records,order details and ,sales record .

PROFILE OF FURNITURE SHOWROOM :

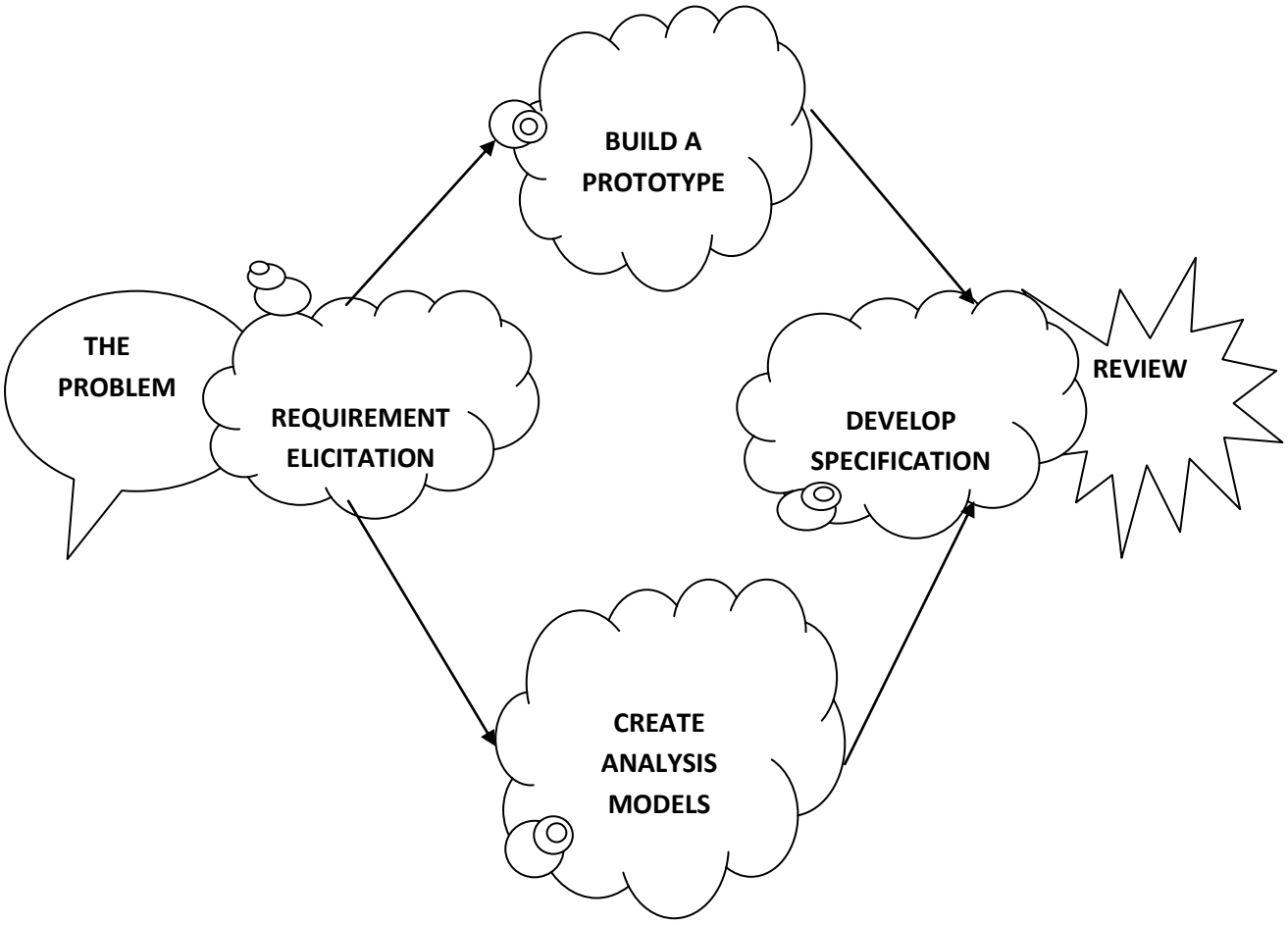
SHOP NAME	Vivek furniture
SHOP OWNER NAME	Mr. A.Vivekanandhan
YEAR OF ESTABLISHMENT	**2008**
ADDRESS	No,147-P,Kangeyam Road,Near DSK Hospital,Tirpur,Tamil Nadu,641604
PHONE NO	9842241576
EMAIL ID	furniturevivek@gmail.com

REQUIREMENT ELICITATION

CHAPTER-II
REQUIREMENT ELICITATION

Requirement elicitation it is the collection of the requirements of any given system or product from users, customers and other important stakeholders. While using the information from customers, you can find that an elicitation process is far more dedicated and widespread than usual information gathering.It is the practice of researching and discovering the requirements of a system from users,It involves studying a procedure or business in order to identify its goals and purposes and create system and procedures that will achieve them in an efficient way. Use cases are a widely used system modelling tool for identifying and expressing the functional requirements of a system.

REQUIREMENT ANALYSIS ELICITAION PROCESS



EXISTING SYSTEM

The Stock maintenance uses MS Access as a backend and maintains their records. There is lot of duplicate work, and chance of mistake. There is no option to find and print previous saved records. There is no security anybody can access any report and sensitive data, also no reports to summary report. The existing system has no security measure against logging in and no checks are made for authorized users. The end user has to remember a lot of command to make efficient use of the system. The system does not have any descriptive reports and thus did not help management in decision-making.

DRAW BACKS OF THE EXISTING SYSTEM

- Only Small volumes of data can be stored with case
- Security of information is low.
- Time Consumption.
- Maintanence of files is risky
- High Manpower.

PROPOSED SYSTEM

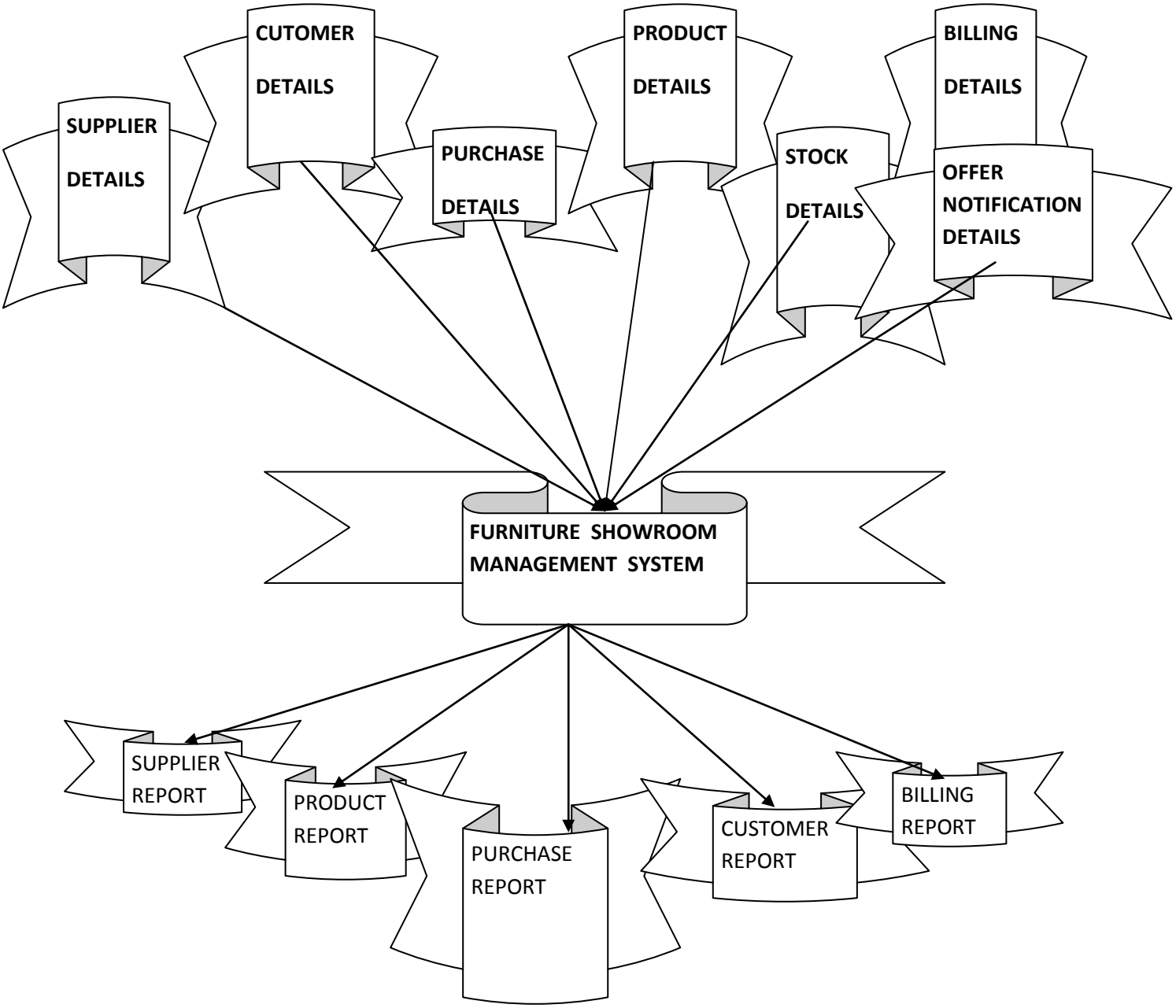
The drawbacks, which are faced during existing system, can be eradicated by using the proposed system. The main objective of the existing system is to provide a user-friendly interface. The system, which is proposed, now computerizes all the details that are maintained manually. Once the details are fed into the computer there is no need for various persons to deal with separate sections. Only a single person is enough to maintain all the reports. The security can also be given as per the requirement of the users.

BENEFITS OF PROPOSED SYSTEM

The following benefits accrue to the studio. It decides to implement the proposal system.

- Large volumes of data can be stored with case.
- Security of information is high
- Time saver
- Less manpower required
- Maintenance of file is flexible

DEVELOPMENT OF PROTOTYPE



FEASIBILITY STUDY

Feasibility study is carried out to select the best system that meets the performance requirements. Feasibility study is both necessary and prudent to evaluate the feasibility of the project at the earliest possible time.Feasibility is a practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software or not. Such information as resource availability, cost estimate for

software development, benefits of the software to organization, and cost to be incurred on its maintenance are considered. The objective of the feasibility study is to establish the reasons for developing a software that is acceptable to users, adaptable to change, and comfortable to standards.

The different types of feasibility are:

- Technical feasibility
- Economical feasibility
- Operational feasibility

TECHNICAL FEASIBILITY

In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyzes technical skills and capabilities of technical team, existing technology can be used or not, maintenance and up-gradation is easy or not for chosen technology etc.

- It analyses the technical and capabilities of the software development team members.
- It determines whether the relevant technology is stable and established.
- It ascertains that the technology chosen for software development has large number of user so that they can be consulted when problems arise, or when improvements are required.

OPERATIONAL FEASIBILITY

In Operational Feasibility degree of providing service to requirements is analyzed along with how much easy product will be to operate and maintenance after deployment. Along with this other operational scopes are determining usability of product, Determining suggested solution by software development team is acceptable or not etc.

ECONOMICAL FEASIBILITY

In Economic Feasibility study cost and benefit of the project is analyzed. It means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on.

After that it is analyzed whether project will be beneficial in terms of finance for organization or not.

- Cost incurred on software development to produce long-term gains for an organization.
- Cost required conducting full software investigation.
- Cost of hardware, software, development team and training.

FEASIBILITY STUDY PROCESS

- **Information assessment** Identifies information about whether the system helps in achieving the objectives of the organization. It also verifies that the system can be implemented using new technology and within the budget, and whether the system can be integrated with the existing system.
- **Information collection** Specifies the sources from where information about software can be obtained. Generally, these sources include users, and the software development team.
- **Report writing** Uses a feasibility report, which is the conclusion of the feasibility by the software development team. In includes the recommendation whether the software development should continue or not.

SYSTEM REQUIREMENT SPECIFICATION

CHAPTER-III

SYSTEM REQUIREMENT SPECIFICATION

A System Requirements Specifications is also known as a Software Requirements Specification is a document that describes the nature of a project, software or application. In simple words, SRS document is a manual of a project provided it is prepared before the user start a project/application. This document is also known by the names SRS report, software document. A software document is primarily prepared for a project, software or any kind of application.The system requirement specification fully describes what the software will do and how it will be expected to perfrom.

HARDWARE SPECIFICATION

The hardware must suit the application and development. It stores all the information and data permanently. So hardware must be reliable and cost effective. It is the fast enough to complete and it works effectively and efficiently.

PROCESSOR	:	INTEL i5 5th Generation
HARD DISK CAPACITY	:	500 GB
MONITOR	:	18.5 “ZEBION MONITOR
RAM	:	4 GB
KEYBOARD	:	LOGITECH
CPU CLOCK	:	2.4 GHz
MOUSE	:	LOGITECH

SOFTWARE SPECIFICATION

The software is an application to be used for the development of the proposed system.

OPERATION SYSTEM	:	WINDOWS 7 ULTIMATE
FRONT END	:	VISUAL STUDIO 2012
BACK END	:	SQL SERVER 2008

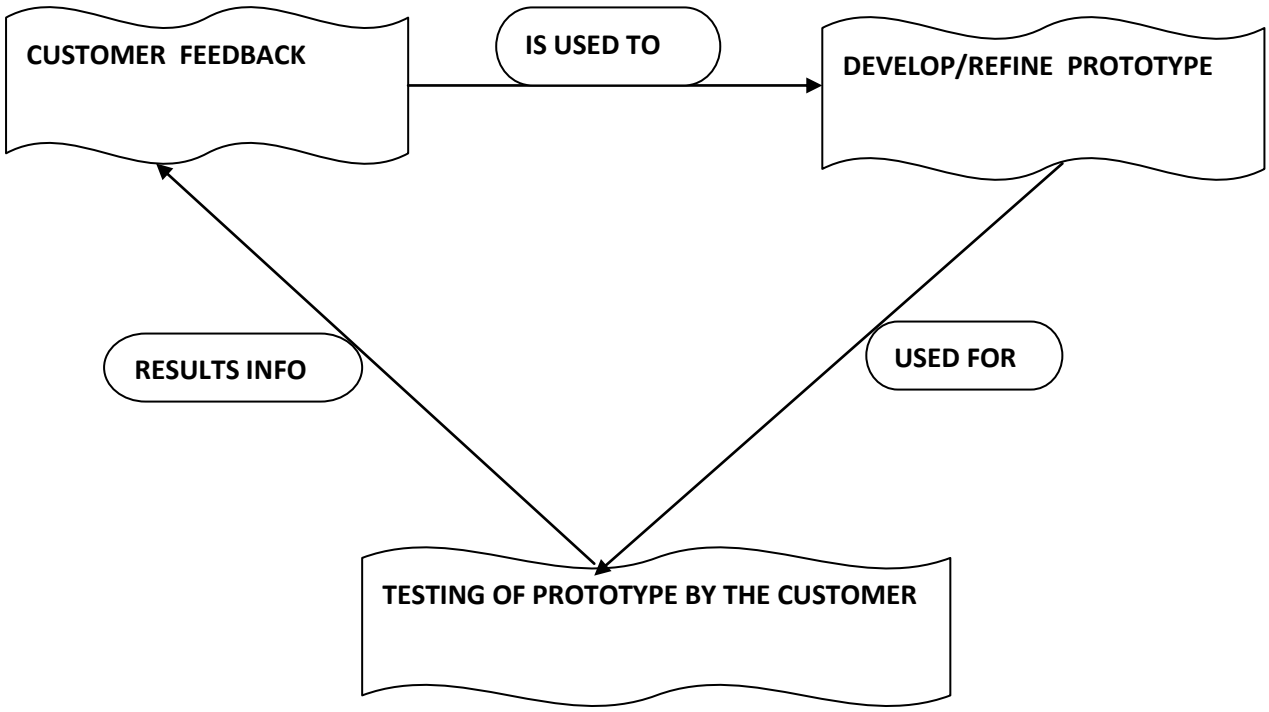
SOFTWARE PROTOTYPING:

The Software Prototyping refers to building software application prototypes which displays the functionality of the product under development, but may not actually hold the exact logic of the original software.Software prototyping is becoming very popular as a software development model, as it enables to understand customer requirements at an early stage of development. It helps get valuable feedback from the customer and helps software designers and developers understand about what exactly is expected from the product under development.

PROTOTYPING MODEL

The Prototyping Model is a Systems Development Methodology (SDM) within which a paradigm output (or an early approximation of a final system or product) is constructed, tested, and then reworked. It is done till an appropriate paradigm is achieved to help develop the entire system or product. This model works best in situations when all the details or requirements are not known well in advance. It is majorly a trial-and-error process which works in an iterative fashion. The Prototyping Model is one of the most popularly used Software Development Life Cycle Models (SDLC models).This model is used when the customers do not know the exact project requirements beforehand. In this model, a prototype of the end product is first developed, tested and refined as per customer feedback repeatedly till a final acceptable prototype is achieved which forms the basis for developing the final product. In this process model, the system is partially implemented before or during the analysis phase thereby giving the customers an opportunity to see the product early in the life cycle. The process starts by interviewing the customers and developing the incomplete high-

level paper model. This document is used to build the initial prototype supporting only the basic functionality as desired by the customer. Once the customer figures out the problems, the prototype is further refined to eliminate them. The process continues until the user approves the prototype and finds the working model to be satisfactory.



INCREMENTAL PROTOTYPING

Incremental prototyping refers to building multiple functional prototypes of the various sub-systems and then integrating all the available prototypes to form a complete system.In this type , the final expected product is broken into different small pieces of prototypes and being developed individually. In the end, when all individual pieces are properly developed, then the different prototypes are collectively merged into a single final product in their predefined order. It's a very efficient approach which reduces the complexity of the development process, where the goal is divided into sub-parts and each sub-part is developed individually. The time interval between the project begin and final delivery is substantially reduced because all parts of the system are prototyped and tested simultaneously. Of course, there might be the possibility that the pieces just not fit

together due to some lack ness in the development phase this can only be fixed by careful and complete plotting of the entire system before prototyping starts.

WINDOWS 7

Microsoft has launched another wonder called the “Microsoft Windows 7”. This is not a single form but it is a complete package of a number of editions which are designed for different users having specific features for the desired purpose.Microsoft announced it was developing a new OS that would become Windows 7. The company revealed the name Windows 7 in October 2008 , and made the OS generally available on Oct. 22, 2009. Windows 7 is an operating system that was produced by Microsoft and released as part of the WINDOWS NT family of operating systems.each edition of windows 7 includes all of the capabilities and features of the edition below it, and adds additional features oriented towards their market segments. Windows 7 edition is best known for its reliability,compatibility,and performance.It contains all the features and applications that are not available on other editions of windows 7.

It comes with tons of benefits and performance improvements which are very crucial for the system performance.Most of the windows users depend on this version of windows 7 because of its features and tools it has to offer.Windows 7 has improved security features and other enchanced features that were not present in the previous versions of windows like home basic,home premium and professional version.Windows 7 edition is aimed larger companies and other large institutes.

Windows 7 is worth a try.With Windows 7, users can pin applications to the taskbar. In addition, users can rearrange the applications on the taskbar in any order they see fit. Other additions include libraries for storing files.. In addition, Windows 7 was the first version of Windows to support multi touch capabilities. It also features more accurate handwriting recognition.Windows 7 introduced the Snap and Shake capabilities. Snap enables a user to drag an open window to the left or right side of the screen and have it automatically resize to take up half the screen.

If a user pulls the window off the side, it reverts to the size and shape it was before he snapped it to the side of the screen. A user can automatically maximize a window by dragging it to the top of the screen.With Shake, users can hide all inactive

windows to reveal the desktop by clicking the top of an open window and quickly dragging it back and forth. Users can also easily reach the desktop with the Show Desktop button on the bottom right of the screen, which minimizes all open windows.

FEATURES

- Fast user switching
- Network assistant
- Remote control for the diagnosis(Remote assistant)
- Simplified user interface
- Windows Media Player
- Internet Explorer 6.0
- Windows
- Movie Maker

VISUAL STUDIO 2012

VB.NET provides the easiest, most productive language and tool for rapidly building Windows and Web applications. Visual Basic .NET comes with enhanced visual designers, increased application performance and a powerful integrated development environment (IDE). It also supports creation of applications for wireless, Internet-enabled hand-held devices.

The system is developed using Visual Basic. NET, which is a very popular Microsoft Product developed by Microsoft Corporation. This is one of the improved languages from basic language. Visual basic .NET includes a variety of open active controls for user interfaces to design application forms.

Like all other .NET languages, VB.NET has complete support for object-oriented concepts. Everything in VB.NET is an object, including all of the primitive types (Short, Integer , Long , String, Boolean etc) and user-defined types, events, and even assemblies. All objects inherits from the based class object.VB.NET provides the easiest, most productive language and tool for rapidly building Windows and Web applications. Visual Basic .NET comes with enhanced visual designers, increased application performance and a powerful integrated development environment (IDE). It also supports creation

of applications for wireless, Internet-enabled hand-held devices. The following reasons make VB.Net a widely used professional language- Modern, general purpose.

- ✓ Object oriented
- ✓ Component oriented
- ✓ Easy to learn
- ✓ Structured language
- ✓ It produces efficient programs
- ✓ It can be compiled on a variety of computer platforms
- ✓ Part of .Net framework

Features Vb.Net

- VB.NET is an ideal programming language for developing sophisticated professional application for Microsoft windows
- It makes use of the graphical user interface for creating powerful applications, which enables the user to interact easily within an application.
- VB.NET provides many aspects such as easier comprehension, user friendliness and faster application development, which help the developer to design the application more effectively.
- VB.NET provides the facilities such as log in dialog form, browser form, query form, option dialog form and wizard from which enable the developer design the application more effectively

SQL SERVER 2008

SQL Server is a software applications first launched with Microsoft SQL Server 2005 that is used for configuring, managing and administering all components within Microsoft SQL Server. The tool includes both script editors and graphical tools which work with objects and features of the server.

A central feature of SSMS is the object explorer, which allows the user to browse, select, and act upon any of the objects within the server. It also shipped a separate express edition that could be freely downloaded; however recent versions of SSMS are fully capable of connecting to and manage any SQL Server Express instance. Microsoft also incorporated backwards compatibility for older versions of SQL Server instances.

Features of SQL Server

The OLAP Services feature available in SQL Server Version 7.0 is now called SQL Server 2008 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server Version 7.0 is now called Microsoft SQL Server 2008 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Service.

SYSTEM DESIGN AND DEVELOPMENT

CHAPTER-IV

SYSTEM DESIGN AND DEVELOPMENT

SYSTEM DESIGN

Software design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

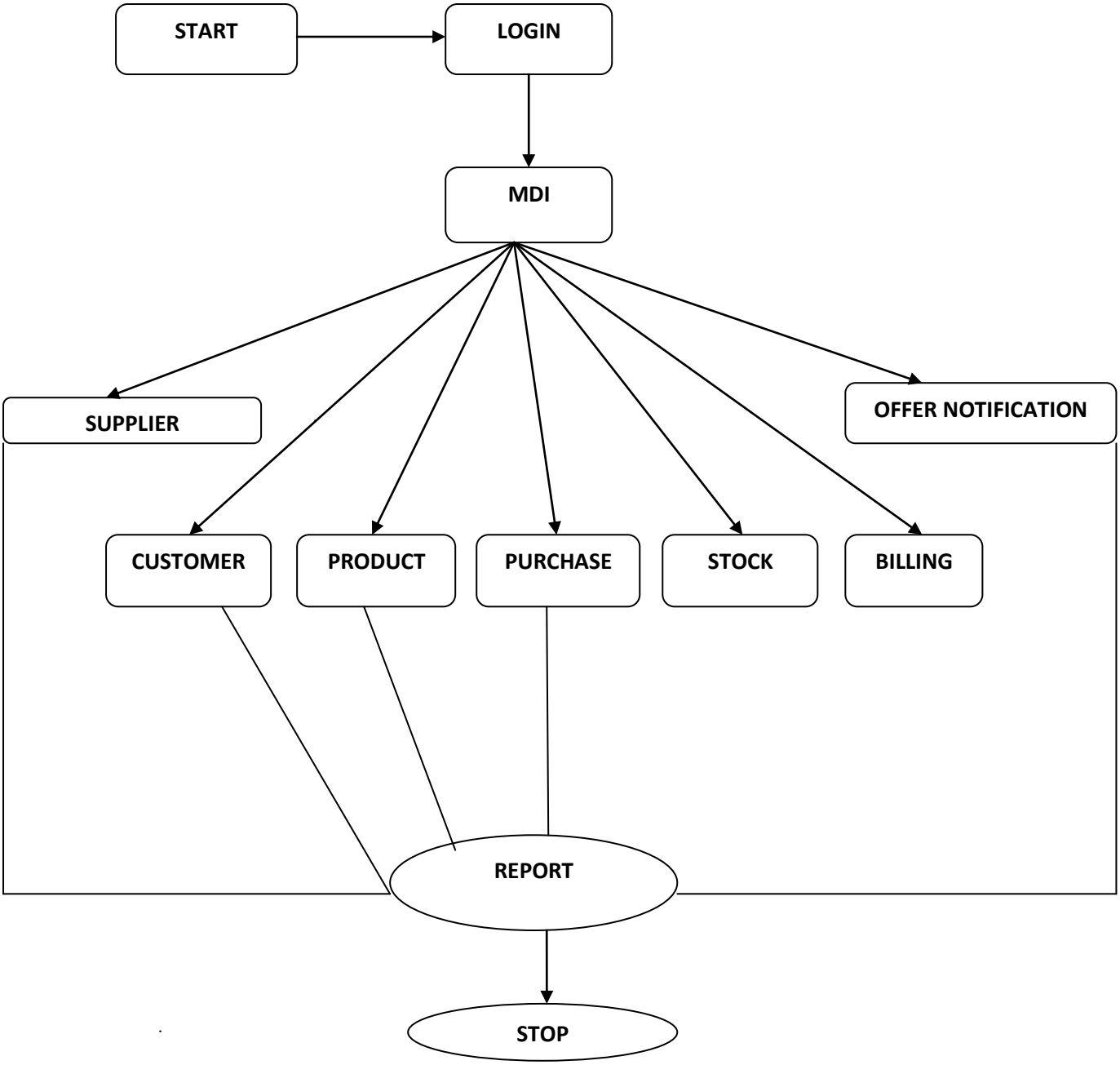
The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective.rom the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.

System design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirement. System design could be seen as the application of system theory to product development.

SYSTEM FLOW DIAGRAM

FURNITURE SHOWROOM MANAGEMENT SYSTEM



MODULES

- Admin login
- Customer Details
- Supplier Details
- Stock Details
- Purchase Details
- Product Details
- Billing Details
- Offer notification
- Report

ADMIN LOGIN

This is the initial module in which admin can login**.** Admin has all right to view and modify the details in the system. Admin logs into the system by specifying unique username and password. Admin has to enter all the details subjected to the studio and has the full authority to add, delete and update information.

CUSTOMER DETAILS

This module is used for customer Information registration for Billing Software. Once the customer is registered we can use the same Account for all consequent items can purchased we add the sub module of the form like customer id, customer name and address.

SUPPLIER DETAILS

This module is used for Supplier registration for delivery in the Furniture. If the stock is not available, the Furniture shop orders and buys from a prescribed vendor. The amount will be paid by deducting the total amount acquired in the sales activity Once the Supplier is registered we can use the same logon for all consequent items can purchased we add the sub module of the form like Supplier id, Supplier name and address.

STOCK DETAILS

This module is used to Stock Furniture items. Stock for sales or else he/she needs to advertise he/she uses this module the items bought from the vendor will be entered here and this will be added to the stock.

PURCHASE DETAILS

This is purchase module from where all procedures take place. Only through this module purchase can start/stop Furniture and also to view details of users, products going to present in sales, Stock level are updated by purchase through this module. This provides the report of the items sold for a particular month/ year and also gives the total amount acquired.

BILLING DETAILS

As the customer buys the products and comes to the billing counter, the user is supposed to enter the item name it purchased and the quantity of the item wanted to purchase. This is not a huge a task. The system will display all the items whose name starts with the letter selected by the user. It can select out of those displayed. Finally a separate will be generated for each customer.

OFFER NOTIFICATION

In offer notification customer can enter their name,mobile number,email id,address and description and click on send ., so that the customer will receive the message whether the item is on sale.

REPORT GENERATION

The report is the final stage in this project. The report can be generated for various details. The reports for the stock of the produced goods can be generated for verification that the delivery can be done or not.

FILE DESIGN

A code generator is a suite of programs that matches the input to an appropriate code template and from these produces modules of code. The code is made simple in such a way that another programmer can easily understand and work on that in future. The crucial phase in the system development life cycle is the successful implementation of the new system design. The process of converting as new or revised system into an operational one is known as system implementation. This includes all those activities that take place to convert from an old system to a new system. The system can be implemented only after a through testing is done and if it is found to work according to the specifications.

INPUT DESIGN

Input design is the process of connecting the user-originated inputs into a computer to used formats. The goal of the input design is to make data entry Logical and free from errors. Errors in the input database controlled by input design this application is being developed in a user-friendly manner. The forms are being designed in such a way that during the processing the cursor is placed in the position where the data must be entered. An option of selecting an appropriate input from the values of validation is made for each of the data entered. Concerning clients comfort the project is designed with perfect validation on each field and to display error messages with appropriate suggestions. Help managers are also provided whenever user entry to a new field he/she can understand what is to be entered. Whenever user enter a error data error manager displayed user can move to next field only after entering a correct data

OUTPUT DESIGN

Computer output is the most important and direct source of information to the user. Efficient intelligible output design should improve the system's relationship with the user and admin in decision-making. Output design generally refers to the results generated by the system. For many end users on the basis of the output the evaluate the usefulness of the application. Efficient software must be able to produce and efficient effective reports.

DATABASE DESIGN

The database design involves creation of tables that are represented in physical database as stored files. They have their own existence. Each table constitute of rows and columns where each row can be viewed as record that consists of related information and column can be viewed as field of data of same type. The table is also designed with some position can have a null value. The database design of project is designed in such a way values are kept without redundancy and with normalized format. Refer the appendix for screen shots of Database Design.

DATA FLOW DIAGRAM

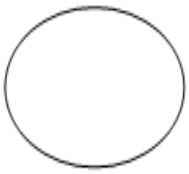
A data Flow diagram (DFD) is a process-orientation representation of an application system. It is a picture of the movement of data between external entities and the process and data stores within a system. It is a graphical representation of the "flow" of data through an

information system. DFDs can also be used for the visualization of data processing. It provides no information about the timing of processes, or about whether processes will operate in sequence or in parallel. It is an element of structured analysis. It is graphical representation of a system or a system or a portion of a system.

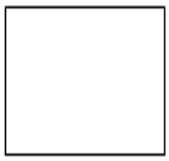
The components of a typical dataflow diagram are as follows,

- The entity process.
- The flow.
- The data store.

DFD SYMBOLS:

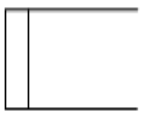


Process that transforms data flow



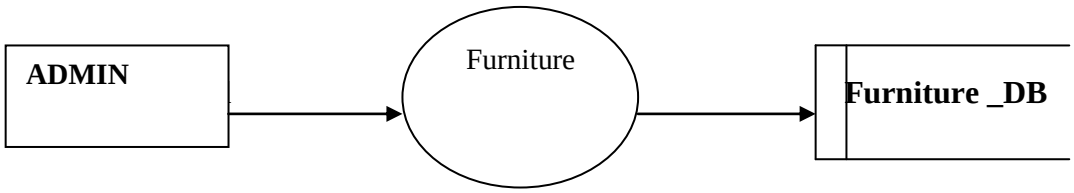
Entity

Data flow

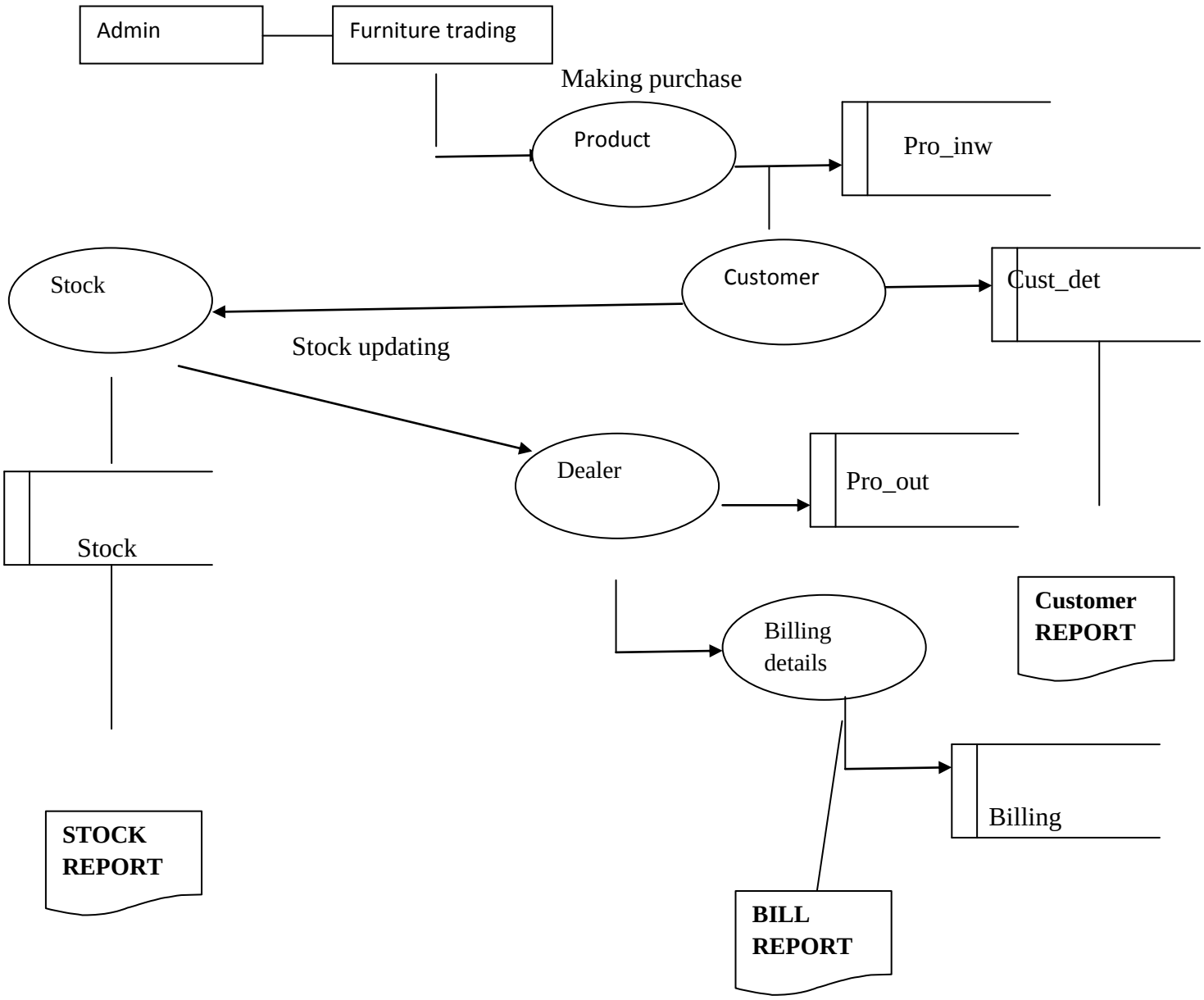


Data Base or Table

LEVEL-0



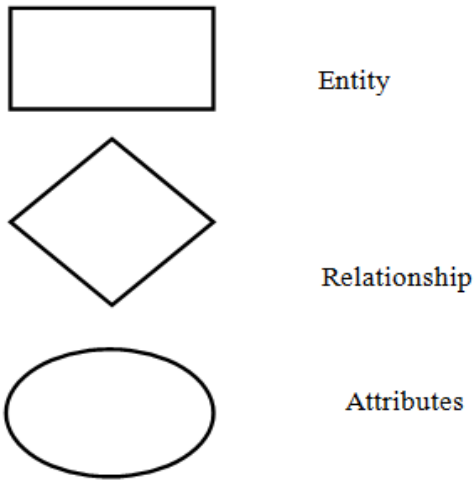
LEVEL-1



ENTITY RELATIONSHIP DIAGRAM

The Entity-Relationship (ER) model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram, which is used to visually represent data objects. The model has been extended and today it is commonly used for database design. The entity relationship diagram is based on a perception of real world that consists of a collection of basic objects, called entities and of relationship among the objects. Entities are described in a database by a set of attributes. The set of all entities of the same type, and the set of all relationships of the same type, are termed as an entity set, and relationship set respectively. The overall logical structure of a database can be expressed graphically by an entity relationship diagram, which is built up using the notations the features of ER model ,it maps well to relational model .It is simple and easy to understand with minimum of training. Therefore, database designer to communicate with the end user can use the model.In addition, the model can be used as a design plan by the database to implement a data model in specific database management software.

ER-DIAGRM SYMBOL



ENITITY REALTIONSHIP DIAGRAM

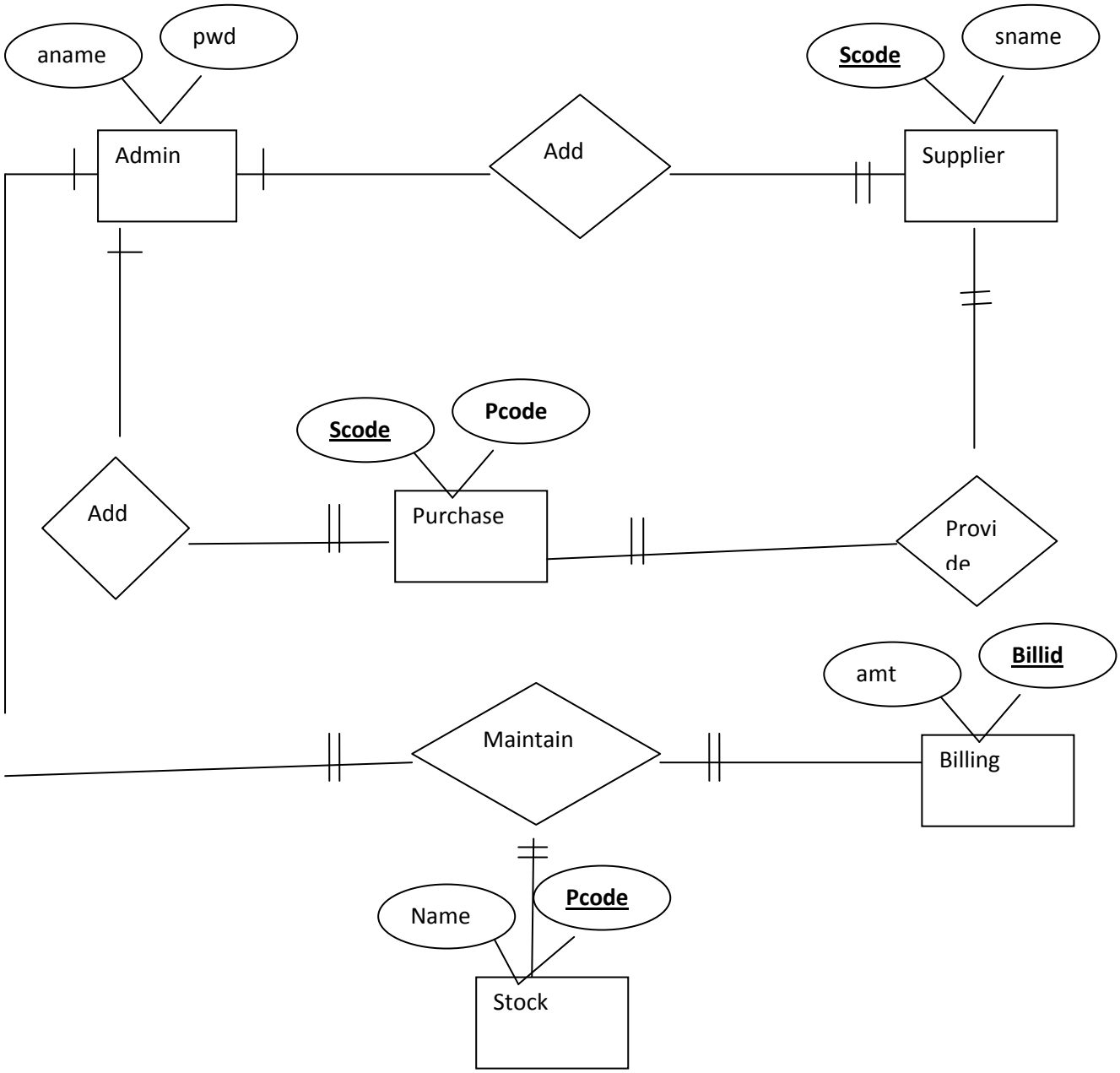


TABLE NAME:SUPPLIER DETAILS

PRIMARY KEY:SCODE

Field name	Data type	Description
Curdate	Varchar,(50)	This field contains current date
Scode	Varchar(50)	This field contains supplier code
Sname	Varchar,(20)	This field contains supplier name
Compname	Varchar,(25)	This field contains company name
Compaddr	Varchar,(50)	This field contains company address
Mno	Varchar,(50)	This field contains mobile number
Fno	Varchar,(50)	This field contains fax number

TABLE NAME:PRODUCT DETAILS

PRIMARY KEY:PCODE

Field Name	Data type	Description
Curdate	Varchar,(50)	This field contains current date
Pcode	Vaarchar,(50)	This field contains product code
Iname	Varchar,(20)	This field contains Item name
Itype	Varchar,(20)	This field contains Item type

TABLE NAME:PURCHASE DETAILS

PRIMARY KEY:PURID

Field name	Data type	Description
Curdate	Varchar,(50)	This field contains current date
Purid	Varchar,(50)	This field contains product code
Scode	Varchar,(20)	This field contains supplier code
Sname	Varchar(50)	This field contains supplier name
Pcode	Varchar,(50)	This field contains product code
Iname	Varchar,(50)	This field contains item name
Itype	Varchar,(50)	This field contains Item type
Compname	Varchar,(50)	This field contains company name
Noofqty	Bignit	This field contains number of quantity
Amtqty	Varchar,(50)	This field contains amount per quantity
Totamt	Varchar,(50)	This field contains Total amount

TABLE NAME:CUSTOMER DETAILS

PRIMARY KEY:CUSDET

Field name	Data type	Description
Curdate	Varchar,(50)	This field contains current date
Ccode	Varchar(50)	This field contains customer code
Cname	Varchar,(50)	This field contains customer name
Caddr	Varchar,(50)	This field contains customer address
Cmno	Varchar,(50)	This field contains customer mobile number

TABLE NAME:BILLING DETAILS

PRIMARY KEY:BILLID

Field Name	Data type	Description
Curdate	Varchar,(50)	This field contains cuurentdate
Billid	Varchar,(50)	This field contains bill Id
Ccode	Varchar,(20)	This field contains customer code
Cname	Varchar,(20)	This field contains customer number
Purid	Varchar,(20)	This field contains purchase id
Iname	Varchar,(50)	This field contains item name
Itype	Varchar.(50)	This field contains item type
Noofqty	Varchar,(50)	This field contains no.of.quantity
Amtqty	Varchar,(50)	This field contains amount per quantity
Totamt	Bignit	This field contains total amount

TABLE NAME:OFFER NOTIFICATION

Field Name	Data type	Description
Custmobile	Varchar,(50)	This field contains the customer mobile number
Custname	Varchar,(50)	This field contains the customer name
Email	Varchar,(50)	This field contains the email
Address	Varchar,(50)	This field contains the address
Description	Varchar,(MAX)	This field contains the description

NORMALIZATION

Normalization is the process of organizing data into a related table; it also eliminates redundancy and increases the integrity which improves performance of the query. To normalize a database, we divide the database into tables and establish relationships between the tables.It is the process of efficiently organizing data in a database. There are two goals of the normalization process: eliminating redundant data (for example, storing the same data in more than one table) and ensuring data dependencies make sense (only storing related data in a table). Both of these are worthy goals as they reduce the amount of space a database consumes and ensure that data is logically stored. There are several benefits for using Normalization in Database.

BENEFITS

- Reduction of redundant data
- Data consistency within the database
- A much more flexible database design
- Faster update due to less number of columns in one table
- A better handle on database security

FIRST NORMAL FORM (1NF)

First normal form (1NF) sets the very basic rules for an organized database: Eliminate duplicative columns from the same table. Create separate tables for each group of related data and identify each row with a unique column or set of columns (the primary key).

- Create Primary KeY

SECOND NORMAL FORM

- Second normal form (2NF) further addresses the concept of removing duplicative data:
- Meet all the requirements of the first normal form.

THIRD NORMAL FORM (3NF)

- Third normal form (3NF) goes one large step further:
- Meet all the requirements of the second normal form.

- Remove columns that are not dependent upon the primary key.

LOGIN FORM

```
Public Class Form1
    Dim a As String
 Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
        If uname.Text = "" Or pwd.Text = "" Then
            MsgBox("Please Enter Username and Password Then Login",
MsgBoxStyle.Information)
        ElseIf uname.Text = "admin" And pwd.Text = "admin" Then
            Dim f As New Main
            f.Show()
            Me.Hide()
        Else
            MsgBox("Invalid Username And Password", MsgBoxStyle.Information)
            uname.Text = ""
            pwd.Text = ""
            uname.Focus()
        End If
    End Sub
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
        End
End Sub
Private Sub uname_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
uname.KeyPress
        If Not Char.IsLetter(e.KeyChar) AndAlso Not Char.IsControl(e.KeyChar) Then
e.KeyChar = ""
    End Sub
End Class
```

HOMEPAGE:

```
Public Class Main
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        Dim f As New Supplier
        f.Show()
    End Sub
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
        Dim f As New Purchase
        f.Show()
    End Sub
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
        Dim f As New Stock
        f.Show()
    End Sub
    Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button7.Click
        Dim f As New Product
        f.Show()
    End Sub
    Private Sub Button9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button9.Click
        Dim f As New Customer
        f.Show()
    End Sub
    Private Sub Button8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button8.Click
        Dim f As New Billing
        f.Show()
    End Sub
```

```
    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
        Dim f As New Supreport
        f.Show()
    End Sub
    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
        Dim f As New purchasereport
        f.Show()
    End Sub
    Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
        Dim f As New Productreport
        f.Show()
    End Sub
    Private Sub Button12_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button12.Click
        Dim f As New cusreport
        f.Show()
    End Sub
    Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button11.Click
        Dim f As New BillReport1
        f.Show()
    End Sub
    Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
        End
    End Sub
    Private Sub Button13_Click(ByVal sender As System.Object, ByVal e
AsSystem.EventArgs) Handles Button13.Click
        Dim f As New offernot
        f.Show()
End Sub
```

```
Private Sub Button14_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        Dim f As New servicenot
        f.Show()
    End Sub
  Private Sub Main_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    End Sub
End Class
```

SUPPLIER CODING

```
Imports System.Data
Imports System.Data.SqlClient
Public Class Supplier
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As SqlCommand
    Dim qry As String
    Dim adp As SqlDataAdapter
    Dim ds As DataSet
    Dim i As Integer
 Private Sub bclear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bclear.Click
        curdate.Text = Date.Today
        sname.Text = ""
        compname.Text = ""
        compaddr.Text = ""
        mno.Text = ""
        phno.Text = ""
        fno.Text = ""
        Try
            Dim p As String
            qry = "select scode from supdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
```

```
        ds = New DataSet
        adp.Fill(ds, "supdet")
        i = ds.Tables("supdet").Rows.Count
        p = ds.Tables("supdet").Rows(i - 1)(0)
        p = Mid(p, 2)
        p = Val(p) + 1
        If Len(p) = 1 Then
            p = "000" + p
        ElseIf Len(p) = 2 Then
            p = "00" + p
        ElseIf Len(p) = 3 Then
            p = "0" + p
        End If
        scode.Text = "S" + p
    Catch ex As Exception
        scode.Text = "S0001"
    End Try
    Try
        qry = "select * from supdet"
        com = New SqlCommand(qry, con)
        adp = New SqlDataAdapter(com)
        ds = New DataSet
        adp.Fill(ds, "supdet")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember = "supdet"
    Catch ex As Exception
    End Try
  End Sub
Private Sub bsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bsave.Click
    If curdate.Text = "" Or sname.Text = "" Or compname.Text = "" Or compaddr.Text = "" Or mno.Text = "" Or phno.Text = "" Or fno.Text = "" Then
        MsgBox("Please Fill All Details Then Save", MsgBoxStyle.Information)
    Else
```

```
        qry = "insert into supdet values('" + curdate.Text + "','" + scode.Text + "','" + sname.Text + "','" + compname.Text + "','" + compaddr.Text + "','" + mno.Text + "','" + phno.Text + "','" + fno.Text + "')"
        com = New SqlCommand(qry, con)
        con.Open()
        com.ExecuteScalar()
        con.Close()
        MsgBox("Successfully Saved", MsgBoxStyle.Information)
    End If
    curdate.Text = Date.Today
    sname.Text = ""
    compname.Text = ""
    compaddr.Text = ""
    mno.Text = ""
    phno.Text = ""
    fno.Text = ""
    Try
        Dim p As String
        qry = "select scode from supdet"
        com = New SqlCommand(qry, con)
        adp = New SqlDataAdapter(com)
        ds = New DataSet
        adp.Fill(ds, "supdet")
        i = ds.Tables("supdet").Rows.Count
        p = ds.Tables("supdet").Rows(i - 1)(0)
        p = Mid(p, 2)
        p = Val(p) + 1
        If Len(p) = 1 Then
            p = "000" + p
        ElseIf Len(p) = 2 Then
            p = "00" + p
        ElseIf Len(p) = 3 Then
            p = "0" + p
        End If
```

```
            scode.Text = "S" + p
        Catch ex As Exception
            scode.Text = "S0001"
        End Try
        Try
            qry = "select * from supdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "supdet")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "supdet"
        Catch ex As Exception
        End Try
    End Sub
    Private Sub bsearch_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles bsearch.Click
        Try
            Dim s As String
            s = InputBox("Enter Supplier Code:")
            qry = "select * from supdet where scode='" + s + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "supdet")
            curdate.Text = ds.Tables("supdet").Rows(0)(0)
            scode.Text = ds.Tables("supdet").Rows(0)(1)
            sname.Text = ds.Tables("supdet").Rows(0)(2)
            compname.Text = ds.Tables("supdet").Rows(0)(3)
            compaddr.Text = ds.Tables("supdet").Rows(0)(4)
            mno.Text = ds.Tables("supdet").Rows(0)(5)
            phno.Text = ds.Tables("supdet").Rows(0)(6)
            fno.Text = ds.Tables("supdet").Rows(0)(7)
        Catch ex As Exception
```

```
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub bedit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bedit.Click
        If curdate.Text = "" Or sname.Text = "" Or compname.Text = "" Or compaddr.Text = "" Or mno.Text = "" Or phno.Text = "" Or fno.Text = "" Then
            MsgBox("Please Fill Search Any One Detail Then Edit", MsgBoxStyle.Information)
        Else
            qry = "update supdet set curdate='" + curdate.Text + "',sname='" + sname.Text + "',compname='" + compname.Text + "',compaddr='" + compaddr.Text + "',mno='" + mno.Text + "',email='" + phno.Text + "',website='" + fno.Text + "' where scode='" + scode.Text + "'"
            com = New SqlCommand(qry, con)
            con.Open()
            com.ExecuteScalar()
            con.Close()
            MsgBox("Successfully Updated", MsgBoxStyle.Information)
        End If
        curdate.Text = Date.Today
        sname.Text = ""
        compname.Text = ""
        compaddr.Text = ""
        mno.Text = ""
        phno.Text = ""
        fno.Text = ""
        Try
            Dim p As String
            qry = "select scode from supdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "supdet")
            i = ds.Tables("supdet").Rows.Count
```

```
            p = ds.Tables("supdet").Rows(i - 1)(0)
            p = Mid(p, 2)
            p = Val(p) + 1
            If Len(p) = 1 Then
                p = "000" + p
            ElseIf Len(p) = 2 Then
                p = "00" + p
            ElseIf Len(p) = 3 Then
                p = "0" + p
            End If
            scode.Text = "S" + p
        Catch ex As Exception
            scode.Text = "S0001"
        End Try
        Try
            qry = "select * from supdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "supdet")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "supdet"
        Catch ex As Exception
        End Try
    End Sub
    Private Sub bexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bexit.Click
        Me.Close()
    End Sub
    Private Sub Supplier_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        curdate.Text = Date.Today
        sname.Text = ""
        compname.Text = ""
```

```
compaddr.Text = ""
mno.Text = ""
phno.Text = ""
fno.Text = ""
Try
    Dim p As String
    qry = "select scode from supdet"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "supdet")
    i = ds.Tables("supdet").Rows.Count
    p = ds.Tables("supdet").Rows(i - 1)(0)
    p = Mid(p, 2)
    p = Val(p) + 1
    If Len(p) = 1 Then
        p = "000" + p
    ElseIf Len(p) = 2 Then
        p = "00" + p
    ElseIf Len(p) = 3 Then
        p = "0" + p
    End If
    scode.Text = "S" + p
Catch ex As Exception
    scode.Text = "S0001"
End Try
Try
    qry = "select * from supdet"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "supdet")
    DataGridView1.DataSource = ds
    DataGridView1.DataMember = "supdet"
```

```
        Catch ex As Exception
        End Try
    End Sub
    Private Sub curdate_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles curdate.TextChanged
    End Sub
    Private Sub sname_KeyPress(sender As Object, e As KeyPressEventArgs) Handles sname.KeyPress
        If Not Char.IsLetter(e.KeyChar) AndAlso Not Char.IsControl(e.KeyChar) Then e.KeyChar = ""
    End Sub
    Private Sub sname_TextChanged(sender As Object, e As EventArgs) Handles sname.TextChanged
    End Sub
    Private Sub mno_KeyPress(sender As Object, e As KeyPressEventArgs) Handles mno.KeyPress
        If Not Char.IsNumber(e.KeyChar) AndAlso Not Char.IsControl(e.KeyChar) Then e.KeyChar = ""
    End Sub
End Class
```

PRODUCT FORM CODING

```
Imports System.Data
Imports System.Data.SqlClient
Public Class Product
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As SqlCommand
    Dim qry As String
    Dim adp As SqlDataAdapter
    Dim ds As DataSet
    Dim i As Integer
```

```
    Private Sub Product_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        curdate.Text = Date.Today
        iname.Text = ""
        itype.Text = ""
        Try
            Dim p As String
            qry = "select pcode from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            i = ds.Tables("product").Rows.Count
            p = ds.Tables("product").Rows(i - 1)(0)
            p = Mid(p, 2)
            p = Val(p) + 1
            If Len(p) = 1 Then
                p = "000" + p
            ElseIf Len(p) = 2 Then
                p = "00" + p
            ElseIf Len(p) = 3 Then
                p = "0" + p
            End If
            pcode.Text = "P" + p
        Catch ex As Exception
            pcode.Text = "P0001"
        End Try
        Try
            qry = "select * from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
```

```
            DataGridView1.DataMember = "product"
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub bclear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bclear.Click
        curdate.Text = Date.Today
        iname.Text = ""
        itype.Text = ""
        Try
            Dim p As String
            qry = "select pcode from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            i = ds.Tables("product").Rows.Count
            p = ds.Tables("product").Rows(i - 1)(0)
            p = Mid(p, 2)
            p = Val(p) + 1
            If Len(p) = 1 Then
                p = "000" + p
            ElseIf Len(p) = 2 Then
                p = "00" + p
            ElseIf Len(p) = 3 Then
                p = "0" + p
            End If
            pcode.Text = "P" + p
        Catch ex As Exception
            pcode.Text = "P0001"
        End Try
        Try
            qry = "select * from product"
```

```
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "product"
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub bsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bsave.Click
        If curdate.Text = "" Or pcode.Text = "" Or iname.Text = "" Or itype.Text = "" Then
            MsgBox("Please Fill All Details Then Save", MsgBoxStyle.Information)
        Else
            Dim q As String
            q = "0"
            qry = "insert into product values('" + curdate.Text + "','" + pcode.Text + "','" +
iname.Text + "','" + itype.Text + "'," + q + ")"
            com = New SqlCommand(qry, con)
            con.Open()
            com.ExecuteScalar()
            con.Close()
            MsgBox("Successfully Saved", MsgBoxStyle.Information)
        End If
        curdate.Text = Date.Today
        iname.Text = ""
        itype.Text = ""
        Try
            Dim p As String
            qry = "select pcode from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
```

```
            adp.Fill(ds, "product")
            i = ds.Tables("product").Rows.Count
            p = ds.Tables("product").Rows(i - 1)(0)
            p = Mid(p, 2)
            p = Val(p) + 1
            If Len(p) = 1 Then
                p = "000" + p
            ElseIf Len(p) = 2 Then
                p = "00" + p
            ElseIf Len(p) = 3 Then
                p = "0" + p
            End If
            pcode.Text = "P" + p
        Catch ex As Exception
            pcode.Text = "P0001"
        End Try
        Try
            qry = "select * from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "product"
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub bsearch_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles bsearch.Click
        Try
            Dim s As String
            s = InputBox("Enter Product Code:")
            qry = "select * from product where pcode='" + s + "'"
```

```
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            curdate.Text = ds.Tables("product").Rows(0)(0)
            pcode.Text = ds.Tables("product").Rows(0)(1)
            iname.Text = ds.Tables("product").Rows(0)(2)
            itype.Text = ds.Tables("product").Rows(0)(3)
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub bedit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bedit.Click
        If curdate.Text = "" Or pcode.Text = "" Or iname.Text = "" Or itype.Text = "" Then
            MsgBox("Please Search Any One Details Then Edit", MsgBoxStyle.Information)
        Else
            qry = "update product set curdate='" + curdate.Text + "',iname='" + iname.Text +
"',itype='" + itype.Text + "' where pcode='" + pcode.Text + "'"
            com = New SqlCommand(qry, con)
            con.Open()
            com.ExecuteScalar()
            con.Close()
            MsgBox("Successfully Updated", MsgBoxStyle.Information)
        End If
        curdate.Text = Date.Today
        iname.Text = ""
        itype.Text = ""
        Try
            Dim p As String
            qry = "select pcode from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
```

```
            adp.Fill(ds, "product")
            i = ds.Tables("product").Rows.Count
            p = ds.Tables("product").Rows(i - 1)(0)
            p = Mid(p, 2)
            p = Val(p) + 1
            If Len(p) = 1 Then
                p = "000" + p
            ElseIf Len(p) = 2 Then
                p = "00" + p
            ElseIf Len(p) = 3 Then
                p = "0" + p
            End If
            pcode.Text = "P" + p
        Catch ex As Exception
            pcode.Text = "P0001"
        End Try
        Try
            qry = "select * from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "product"
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub bexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bexit.Click
        Me.Close()
    End Sub
End Class
```

PURCHASE FORM CODING

```
Imports System.Data
Imports System.Data.SqlClient
Public Class Purchase
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As SqlCommand
    Dim qry As String
    Dim adp As SqlDataAdapter
    Dim ds As DataSet
    Dim i, j As Integer
    Private Sub Purchase_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        curdate.Text = Date.Today
        scode.Text = ""
        sname.Text = ""
        pcode.Text = ""
        iname.Text = ""
        itype.Text = ""
        compname.Text = ""
        noofqty.Text = ""
        amtqty.Text = ""
        totamt.Text = ""
        scode.Items.Clear()
        Try
            qry = "select scode from supdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "supdet")
            i = ds.Tables("supdet").Rows.Count
            For j = 0 To i - 1
                scode.Items.Add(ds.Tables("supdet").Rows(j)(0))
```

```
    Next
Catch ex As Exception
End Try
pcode.Items.Clear()
Try
    qry = "select pcode from product"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "product")
    i = ds.Tables("product").Rows.Count
    For j = 0 To i - 1
        pcode.Items.Add(ds.Tables("product").Rows(j)(0))
    Next
Catch ex As Exception
End Try
Try
    Dim p As String
    qry = "select purid from purchase"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "purchase")
    i = ds.Tables("purchase").Rows.Count
    p = ds.Tables("purchase").Rows(i - 1)(0)
    p = Mid(p, 2)
    p = Val(p) + 1
    If Len(p) = 1 Then
        p = "000" + p
    ElseIf Len(p) = 2 Then
        p = "00" + p
    ElseIf Len(p) = 3 Then
        p = "0" + p
    End If
```

```
            purid.Text = "H" + p
        Catch ex As Exception
            purid.Text = "H0001"
        End Try
        Try
            qry = "select * from purchase"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "purchase")
            DataGridView2.DataSource = ds
            DataGridView2.DataMember = "purchase"
        Catch ex As Exception
        End Try
        Try
            qry = "select * from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "product"
        Catch ex As Exception
        End Try
    End Sub
    Private Sub bclear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bclear.Click
        curdate.Text = Date.Today
        scode.Text = ""
        sname.Text = ""
        pcode.Text = ""
        iname.Text = ""
        itype.Text = ""
        compname.Text = ""
```

```
noofqty.Text = ""
amtqty.Text = ""
totamt.Text = ""
Try
    Dim p As String
    qry = "select purid from purchase"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "purchase")
    i = ds.Tables("purchase").Rows.Count
    p = ds.Tables("purchase").Rows(i - 1)(0)
    p = Mid(p, 2)
    p = Val(p) + 1
    If Len(p) = 1 Then
        p = "000" + p
    ElseIf Len(p) = 2 Then
        p = "00" + p
    ElseIf Len(p) = 3 Then
        p = "0" + p
    End If
    purid.Text = "H" + p
Catch ex As Exception
    purid.Text = "H0001"
End Try
Try
    qry = "select * from purchase"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "purchase")
    DataGridView2.DataSource = ds
    DataGridView2.DataMember = "purchase"
```

```
        Catch ex As Exception
        End Try
        Try
            qry = "select * from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "product"
        Catch ex As Exception
        End Try
    End Sub
    Private Sub bsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bsave.Click
        If curdate.Text = "" Or purid.Text = "" Or scode.Text = "" Or sname.Text = "" Or
pcode.Text = "" Or iname.Text = "" Or itype.Text = "" Or compname.Text = "" Or
noofqty.Text = "" Or amtqty.Text = "" Or totamt.Text = "" Then
            MsgBox("Please Fill All Details Then Save", MsgBoxStyle.Information)
        Else
            qry = "insert into purchase values('" + curdate.Text + "','" + purid.Text + "','" +
scode.Text + "','" + sname.Text + "','" + pcode.Text + "','" + iname.Text + "','" + itype.Text +
"','" + compname.Text + "'," + noofqty.Text + ",'" + amtqty.Text + "','" + totamt.Text + "')"
            com = New SqlCommand(qry, con)
            con.Open()
            com.ExecuteScalar()
            con.Close()
            qry = "select qty from product where pcode='" + pcode.Text + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            Dim s1 As String
            Dim s2 As String
```

```
    s1 = ds.Tables("product").Rows(0)(0)
    s2 = Val(s1) + Val(noofqty.Text)
    qry = "update product set qty=" + s2 + " where pcode='" + pcode.Text + "'"
    com = New SqlCommand(qry, con)
    con.Open()
    com.ExecuteScalar()
    con.Close()
    MsgBox("Successfully Saved", MsgBoxStyle.Information)
End If
curdate.Text = Date.Today
scode.Text = ""
sname.Text = ""
pcode.Text = ""
iname.Text = ""
itype.Text = ""
compname.Text = ""
noofqty.Text = ""
amtqty.Text = ""
totamt.Text = ""
Try
    Dim p As String
    qry = "select purid from purchase"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "purchase")
    i = ds.Tables("purchase").Rows.Count
    p = ds.Tables("purchase").Rows(i - 1)(0)
    p = Mid(p, 2)
    p = Val(p) + 1
    If Len(p) = 1 Then
        p = "000" + p
    ElseIf Len(p) = 2 Then
        p = "00" + p
```

```
            ElseIf Len(p) = 3 Then
                p = "0" + p
            End If
            purid.Text = "H" + p
        Catch ex As Exception
            purid.Text = "H0001"
        End Try
        Try
            qry = "select * from purchase"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "purchase")
            DataGridView2.DataSource = ds
            DataGridView2.DataMember = "purchase"
        Catch ex As Exception
        End Try
        Try
            qry = "select * from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "product"
        Catch ex As Exception
        End Try
    End Sub
    Private Sub bexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bexit.Click
        Me.Close()
    End Sub
```

```
    Private Sub scode_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles scode.SelectedIndexChanged
        Try
            qry = "select sname,compname from supdet where scode='" + scode.SelectedItem + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "supdet")
            sname.Text = ds.Tables("supdet").Rows(0)(0)
            compname.Text = ds.Tables("supdet").Rows(0)(1)
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub pcode_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles pcode.SelectedIndexChanged
        Try
            qry = "select iname,itype from product where pcode='" + pcode.SelectedItem + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            iname.Text = ds.Tables("product").Rows(0)(0)
            itype.Text = ds.Tables("product").Rows(0)(1)
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub totamt_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles totamt.Click
        totamt.Text = Val(noofqty.Text) * Val(amtqty.Text)
    End Sub
 End Class
```

STOCK DETAILS CODING :

```
Imports System.Data
Imports System.Data.SqlClient
Public Class Stock
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As SqlCommand
    Dim qry As String
    Dim adp As SqlDataAdapter
    Dim ds As DataSet
    Dim i, j As Integer
    Private Sub Stock_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        pname.Items.Clear()
        Try
            qry = "select iname from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            i = ds.Tables("product").Rows.Count
            For j = 0 To i - 1
                pname.Items.Remove(ds.Tables("product").Rows(j)(0))
                pname.Items.Add(ds.Tables("product").Rows(j)(0))
            Next
        Catch ex As Exception
        End Try
    End Sub
Private Sub bsearch_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bsearch.Click
        Try
            qry = "select * from purchase where iname='" + pname.Text + "'"
            com = New SqlCommand(qry, con)
```

```
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "purchase")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "purchase"
        Catch ex As Exception
        End Try
        Try
            qry = "select sum(noofqty) from purchase where iname='" + pname.Text + "'"
            com = New SqlCommand(qry, con)
            con.Open()
            qty.Text = com.ExecuteScalar
            con.Close()
        Catch ex As Exception
        End Try
        Try
            qry = "select * from product where iname='" + pname.Text + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView2.DataSource = ds
            DataGridView2.DataMember = "product"
        Catch ex As Exception
        End Try
      End Sub
    Private Sub bexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bexit.Click
        Me.Close()
    End Sub
    Private Sub LinkLabel1_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles LinkLabel1.LinkClicked
        offernot.Show()
    End Sub
```

End Class

CUSTOMER DETAILS CODING

```
Imports System.Data
Imports System.Data.SqlClient
Imports System.Text.RegularExpressions
Public Class Customer
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As SqlCommand
    Dim qry As String
    Dim adp As SqlDataAdapter
    Dim ds As DataSet
    Dim i As Integer
Private Sub bclear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bclear.Click
        curdate.Text = Date.Today
        cname.Text = ""
        caddr.Text = ""
        cmno.Text = ""
            Try
        Dim p As String
        qry = "select ccode from cusdet"
        com = New SqlCommand(qry, con)
        adp = New SqlDataAdapter(com)
        ds = New DataSet
        adp.Fill(ds, "cusdet")
        i = ds.Tables("cusdet").Rows.Count
        p = ds.Tables("cusdet").Rows(i - 1)(0)
        p = Mid(p, 2)
        p = Val(p) + 1
        If Len(p) = 1 Then
            p = "000" + p
        ElseIf Len(p) = 2 Then
```

```
            p = "00" + p
        ElseIf Len(p) = 3 Then
            p = "0" + p
        End If
        ccode.Text = "C" + p
    Catch ex As Exception
        ccode.Text = "C0001"
    End Try
    Try
        qry = "select * from cusdet"
        com = New SqlCommand(qry, con)
        adp = New SqlDataAdapter(com)
        ds = New DataSet
        adp.Fill(ds, "cusdet")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember = "cusdet"
    Catch ex As Exception
    End Try
End Sub
Private Sub bsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bsave.Click
    If curdate.Text = "" Or ccode.Text = "" Or cname.Text = "" Or caddr.Text = "" Or
cmno.Text = "" Or TextBox1.Text = "" Then
        MsgBox("Please Fill All Details", MsgBoxStyle.Information)
    Else
        qry = "insert into cusdet values('" + curdate.Text + "','" + ccode.Text + "','" +
cname.Text + "','" + caddr.Text + "','" + cmno.Text + "','" + TextBox1.Text + "')"
        com = New SqlCommand(qry, con)
        con.Open()
        com.ExecuteScalar()
        con.Close()
        MsgBox("Successfully Saved", MsgBoxStyle.Information)
    End If
    curdate.Text = Date.Today
```

```
cname.Text = ""
caddr.Text = ""
cmno.Text = ""
Try
    Dim p As String
    qry = "select ccode from cusdet"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "cusdet")
    i = ds.Tables("cusdet").Rows.Count
    p = ds.Tables("cusdet").Rows(i - 1)(0)
    p = Mid(p, 2)
    p = Val(p) + 1
    If Len(p) = 1 Then
        p = "000" + p
    ElseIf Len(p) = 2 Then
        p = "00" + p
    ElseIf Len(p) = 3 Then
        p = "0" + p
    End If
    ccode.Text = "C" + p
Catch ex As Exception
    ccode.Text = "C0001"
End Try
Try
    qry = "select * from cusdet"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "cusdet")
    DataGridView1.DataSource = ds
    DataGridView1.DataMember = "cusdet"
```

```
        Catch ex As Exception
        End Try
    End Sub
    Private Sub bsearch_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bsearch.Click
        Try
            Dim s As String
            s = InputBox("Enter Customer Code")
            qry = "select * from cusdet where ccode='" + s + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "cusdet")
            curdate.Text = ds.Tables("cusdet").Rows(0)(0)
            ccode.Text = ds.Tables("cusdet").Rows(0)(1)
            cname.Text = ds.Tables("cusdet").Rows(0)(2)
            caddr.Text = ds.Tables("cusdet").Rows(0)(3)
            cmno.Text = ds.Tables("cusdet").Rows(0)(4)
            TextBox1.Text = ds.Tables("cusdet").Rows(0)(5)
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub
    Private Sub bedit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bedit.Click
        If curdate.Text = "" Or ccode.Text = "" Or cname.Text = "" Or caddr.Text = "" Or cmno.Text = "" Then
            MsgBox("Please Search Any One Detail Then Edit", MsgBoxStyle.Information)
        Else
            qry = "update cusdet set curdate='" + curdate.Text + "',cname='" + cname.Text + "',caddr='" + caddr.Text + "',cmno='" + cmno.Text + "',emailid='" + TextBox1.Text + "' where ccode='" + ccode.Text + "'"
            com = New SqlCommand(qry, con)
            con.Open()
```

```
    com.ExecuteScalar()
    con.Close()
    MsgBox("Successfully Updated", MsgBoxStyle.Information)
End If
curdate.Text = Date.Today
cname.Text = ""
caddr.Text = ""
cmno.Text = ""
Try
    Dim p As String
    qry = "select ccode from cusdet"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "cusdet")
    i = ds.Tables("cusdet").Rows.Count
    p = ds.Tables("cusdet").Rows(i - 1)(0)
    p = Mid(p, 2)
    p = Val(p) + 1
    If Len(p) = 1 Then
        p = "000" + p
    ElseIf Len(p) = 2 Then
        p = "00" + p
    ElseIf Len(p) = 3 Then
        p = "0" + p
    End If
    ccode.Text = "C" + p
Catch ex As Exception
    ccode.Text = "C0001"
End Try
Try
    qry = "select * from cusdet"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
```

```
        ds = New DataSet
        adp.Fill(ds, "cusdet")
        DataGridView1.DataSource = ds
        DataGridView1.DataMember = "cusdet"
      Catch ex As Exception
      End Try
   End Sub
   Private Sub bexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bexit.Click
      Me.Close()
   End Sub
   Private Sub Customer_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
      curdate.Text = Date.Today
      cname.Text = ""
      caddr.Text = ""
      cmno.Text = ""
      Try
         Dim p As String
         qry = "select ccode from cusdet"
         com = New SqlCommand(qry, con)
         adp = New SqlDataAdapter(com)
         ds = New DataSet
         adp.Fill(ds, "cusdet")
         i = ds.Tables("cusdet").Rows.Count
         p = ds.Tables("cusdet").Rows(i - 1)(0)
         p = Mid(p, 2)
         p = Val(p) + 1
         If Len(p) = 1 Then
            p = "000" + p
         ElseIf Len(p) = 2 Then
            p = "00" + p
         ElseIf Len(p) = 3 Then
            p = "0" + p
```

```
            End If
            ccode.Text = "C" + p
        Catch ex As Exception
            ccode.Text = "C0001"
        End Try
        Try
            qry = "select * from cusdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "cusdet")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "cusdet"
        Catch ex As Exception
        End Try
    End Sub
    Private Sub cmno_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
cmno.KeyPress
        If Not Char.IsNumber(e.KeyChar) AndAlso Not Char.IsControl(e.KeyChar) Then
e.KeyChar = ""
    End Sub
    Function EmailAddressCheck(ByVal emailAddress As String) As Boolean
        Dim pattern As String = "^[a-zA-Z][\w\.-]*[a-zA-Z0-9]@[a-zA-Z0-9][\w\.-]*[a-zA-Z0-
9]\.[a-zA-Z][a-zA-Z\.]*[a-zA-Z]$"
        Dim emailAddressMatch As Match = Regex.Match(emailAddress, pattern)
        If emailAddressMatch.Success Then
            EmailAddressCheck = True
        Else
            MsgBox("Invalid Email-id")
            EmailAddressCheck = False
            TextBox1.Text = ""
        End If
    End Function
```

```
    Private Sub TextBox1_TextChanged(sender As Object, e As EventArgs) Handles
TextBox1.TextChanged
    End Sub
    Private Sub TextBox1_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles TextBox1.Validating
        EmailAddressCheck(TextBox1.Text)
    End Sub
End Class
```

BILLING DETAILS CODING

```
Imports System.Data
Imports System.Data.SqlClient
Public Class Billing
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial
Catalog=flower;User ID=sa;Password=sql")
    Dim com As SqlCommand
    Dim qry, qty As String
    Dim adp As SqlDataAdapter
    Dim ds As DataSet
    Dim i, j As Integer
    Dim k, s As String
    Private Sub Billing_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        Timer1.Start()
        curdate.Text = Date.Today
        ccode.Text = ""
        cname.Text = ""
        purid.Text = ""
        iname.Text = ""
        itype.Text = ""
        noofqty.Text = ""
        amtqty.Text = ""
        totamt.Text = ""
```

```
Try
    Dim p As String
    qry = "select billid from billing"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "billing")
    i = ds.Tables("billing").Rows.Count
    p = ds.Tables("billing").Rows(i - 1)(0)
    p = Mid(p, 2)
    p = Val(p) + 1
    If Len(p) = 1 Then
        p = "000" + p
    ElseIf Len(p) = 2 Then
        p = "00" + p
    ElseIf Len(p) = 3 Then
        p = "0" + p
    End If
    billid.Text = "B" + p
Catch ex As Exception
    billid.Text = "B0001"
End Try
purid.Items.Clear()
Try
    qry = "select purid from purchase"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "purchase")
    i = ds.Tables("purchase").Rows.Count
    For j = 0 To i - 1
        purid.Items.Add(ds.Tables("purchase").Rows(j)(0))
    Next
Catch ex As Exception
```

```
End Try
ccode.Items.Clear()
Try
    qry = "select ccode from cusdet"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "cusdet")
    i = ds.Tables("cusdet").Rows.Count
    For j = 0 To i - 1
        ccode.Items.Add(ds.Tables("cusdet").Rows(j)(0))
    Next
Catch ex As Exception
End Try
Try
    qry = "select * from purchase"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "purchase")
    DataGridView1.DataSource = ds
    DataGridView1.DataMember = "purchase"
Catch ex As Exception
End Try
Try
    qry = "select * from billing"
    com = New SqlCommand(qry, con)
    adp = New SqlDataAdapter(com)
    ds = New DataSet
    adp.Fill(ds, "billing")
    DataGridView2.DataSource = ds
    DataGridView2.DataMember = "billing"
Catch ex As Exception
End Try
```

```
    End Sub
    Private Sub purid_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles purid.SelectedIndexChanged
        Try
            qry = "select iname,itype,amtqty from purchase where purid='" + purid.SelectedItem + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "purchase")
            iname.Text = ds.Tables("purchase").Rows(0)(0)
            itype.Text = ds.Tables("purchase").Rows(0)(1)
            amtqty.Text = ds.Tables("purchase").Rows(0)(2)
Dim s As String
            qry = "select qty from product where iname='" + iname.Text + "' and itype='" + itype.Text + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            s = ds.Tables("product").Rows(0)(0)
            MsgBox("Total Stock Is:" + s)
        Catch ex As Exception
End Try
    End Sub
    Private Sub totamt_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles totamt.Click
        'totamt.Text = Val(amtqty.Text) * Val(noofqty.Text)
    End Sub
    Private Sub addpro_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
        'cart.Text = Val(cart.Text) + Val(totamt.Text)
    End Sub
```

```
    Private Sub bsave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bsave.Click
        If curdate.Text = "" Or billid.Text = "" Or ccode.Text = "" Or cname.Text = "" Or purid.Text = "" Or iname.Text = "" Or itype.Text = "" Or noofqty.Text = "" Or amtqty.Text = "" Or totamt.Text = "" Then
            MsgBox("Please Fill All Details Then Save", MsgBoxStyle.Information)
        Else
            qry = "insert into billing values('" + curdate.Text + "','" + billid.Text + "','" + ccode.Text + "','" + cname.Text + "','" + purid.Text + "','" + iname.Text + "','" + itype.Text + "','" + noofqty.Text + "','" + amtqty.Text + "'," + totamt.Text + ")"
            com = New SqlCommand(qry, con)
            con.Open()
            com.ExecuteScalar()
            con.Close()
 qry = "select qty from product where iname='" + iname.Text + "' and itype='" + itype.Text + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            adp.Fill(ds, "product")
            Dim s As String
            Dim s1 As String
            s = ds.Tables("product").Rows(0)(0)
            s1 = Val(s) - Val(noofqty.Text)
            qry = "update product set qty=" + s1 + " where iname='" + iname.Text + "' and itype='" + itype.Text + "'"
            com = New SqlCommand(qry, con)
            con.Open()
            com.ExecuteScalar()
            con.Close()
            MsgBox("Successfully Saved", MsgBoxStyle.Information)
        End If
        curdate.Text = Date.Today
        'ccode.Text = ""
        'cname.Text = ""
```

```
        purid.Text = ""
        iname.Text = ""
        itype.Text = ""
        noofqty.Text = ""
        amtqty.Text = ""
        totamt.Text = ""
        TextBox1.Text = ""
        Try
            qry = "select * from purchase"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "purchase")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "purchase"
        Catch ex As Exception
 End Try
        Try
            qry = "select * from billing"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "billing")
            DataGridView2.DataSource = ds
            DataGridView2.DataMember = "billing
 Catch ex As Exception
End Try
End Sub
  Private Sub bexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bexit.Click
        Me.Close()
    End Sub
Private Sub bclear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles bclear.Click
```

```
        curdate.Text = Date.Today
        ccode.Text = ""
        cname.Text = ""
        purid.Text = ""
        iname.Text = ""
        itype.Text = ""
        noofqty.Text = ""
        amtqty.Text = ""
        totamt.Text = ""
        TextBox1.Text = ""
Try
            Dim p As String
            qry = "select billid from billing"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "billing")
            i = ds.Tables("billing").Rows.Count
            p = ds.Tables("billing").Rows(i - 1)(0)
            p = Mid(p, 2)
            p = Val(p) + 1
            If Len(p) = 1 Then
                p = "000" + p
            ElseIf Len(p) = 2 Then
                p = "00" + p
            ElseIf Len(p) = 3 Then
                p = "0" + p
            End If
            billid.Text = "B" + p
        Catch ex As Exception
            billid.Text = "B0001"
        End Try
        Try
            qry = "select * from purchase"
```

```
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "purchase")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "purchase"
        Catch ex As Exception
        End Try
        Try
            qry = "select * from billing"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "billing")
            DataGridView2.DataSource = ds
            DataGridView2.DataMember = "billing"
        Catch ex As Exception
        End Try
    End Sub
    Private Sub ccode_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ccode.SelectedIndexChanged
        Try
            qry = "select cname from cusdet where ccode='" + ccode.SelectedItem + "'"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "cusdet")
            cname.Text = ds.Tables("cusdet").Rows(0)(0)
        Catch ex As Exception
        End Try
    End Sub
    Private Sub amtqty_GotFocus(sender As Object, e As EventArgs) Handles amtqty.GotFocus
        'amtqty.Text = Val(noofqty.Text) * Val(amtqty.Text)
```

```
    End Sub
    Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
        Label15.Text = TimeOfDay
    End Sub
    Private Sub TextBox1_LostFocus(sender As Object, e As EventArgs) Handles TextBox1.LostFocus
        s = Val(k) + Val(k) * Val(TextBox1.Text) / 100
        totamt.Text = s
    End Sub
    Private Sub noofqty_LostFocus(sender As Object, e As EventArgs) Handles noofqty.LostFocus
        k = Val(noofqty.Text) * Val(amtqty.Text)
    End Sub
End Class
```

SUPPLIER REPORT

```
Imports System.Data.SqlClient
Public Class supreport
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As New SqlCommand
    Dim adp As New SqlDataAdapter
    Dim ds As New DataSet
    Dim qry As String
    Dim i As Integer
    Private Sub supreport_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        Try
            qry = "select * from supdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "supdet")
```

```
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "supdet"
        Catch ex As Exception
            MsgBox("Invalid Id")
        End Try
    End Sub
End Class
```

CUSTOMER REPORT

```
Imports System.Data.SqlClient
Public Class cusreport
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As New SqlCommand
    Dim adp As New SqlDataAdapter
    Dim ds As New DataSet
    Dim qry As String
    Dim i As Integer
    Private Sub cusreport_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        Try
            qry = "select * from cusdet"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "cusdet")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "cusdet"
        Catch ex As Exception
            MsgBox("Invalid Id")
        End Try
    End Sub
```

```
End Class
```

PRODUCT REPORT

```
Imports System.Data.SqlClient
Public Class Productreport
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As New SqlCommand
    Dim adp As New SqlDataAdapter
    Dim ds As New DataSet
    Dim qry As String
    Dim i As Integer
    Private Sub Productreport_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        Try
            qry = "select * from product"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "product")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "product"
        Catch ex As Exception
            MsgBox("Invalid Id")
        End Try
    End Sub
End Class
```

PURCHASE REPORT

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

```
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As New SqlCommand
    Dim adp As New SqlDataAdapter
    Dim ds As New DataSet
    Dim qry As String
    Dim i As Integer
    Private Sub purchasereport_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        Try
            qry = "select * from purchase"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "purchase")
            DataGridView1.DataSource = ds
            DataGridView1.DataMember = "purchase"
        Catch ex As Exception
            MsgBox("Invalid Id")
        End Try
    End Sub
End Class
```

BILLING REPORT

```
Imports System.Data
Imports System.Data.SqlClient

Public Class BillReport1
    Dim con As New SqlConnection("Data Source=ADMIN-PC\SQLEXPRESS;Initial Catalog=flower;User ID=sa;Password=sql")
    Dim com As SqlCommand
    Dim qry As String
```

```
    Dim adp As SqlDataAdapter
    Dim ds As DataSet
    Dim i, j As Integer

    Private Sub BillReport1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
     billid.Items.Clear()
        Try
            qry = "select billid from billing"
            com = New SqlCommand(qry, con)
            adp = New SqlDataAdapter(com)
            ds = New DataSet
            adp.Fill(ds, "billing")
            i = ds.Tables("billing").Rows.Count
            For j = 0 To i - 1
                billid.Items.Remove(ds.Tables("billing").Rows(j)(0))
                billid.Items.Add(ds.Tables("billing").Rows(j)(0))
            Next
        Catch ex As Exception
        End Try
    End Sub
    Private Sub bsearch_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bsearch.Click
        Try
            If billid.Text = "" Then
                MsgBox("Please Select Any One BillID Then Search", MsgBoxStyle.Information)
            Else
                qry = "select * from billing where billid='" + billid.Text + "'"
                com = New SqlCommand(qry, con)
                adp = New SqlDataAdapter(com)
                ds = New DataSet
                adp.Fill(ds, "billing")
                DataGridView1.DataSource = ds
                DataGridView1.DataMember = "billing"
```

```
            qry = "select sum(totamt) from billing where billid='" + billid.Text + "'"
            com = New SqlCommand(qry, con)
            con.Open()
            tot.Text = com.ExecuteScalar
            con.Close()
        End If
    Catch ex As Exception
    End Try
  End Sub
  Private Sub bexit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles bexit.Click
    Me.Close()
  End Sub
  Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Try
      qry = "select * from billing"
      com = New SqlCommand(qry, con)
      adp = New SqlDataAdapter(com)
      ds = New DataSet
      adp.Fill(ds, "billing")
      DataGridView1.DataSource = ds
      DataGridView1.DataMember = "billing"
      qry = "select sum(totamt) from billing"
      com = New SqlCommand(qry, con)
      con.Open()
      tot.Text = com.ExecuteScalar
      con.Close()
    Catch ex As Exception
    End Try
  End Sub
End Class
```

FORM DESIGN OUTPUT

LOGIN FORM

HOME PAGE

FURNITURE SHOWROOM INFORMATION

SUPPLIER DETAILS | PRODUCT DETAILS | PURCHASE DETAILS

STOCK DETAILS | CUSTOMER DETAILS | BILLING DETAILS

OFFER NOTIFICATION

SUPPLIER REPORT | PRODUCT REPORT | PURCHASE REPORT

CUSTOMER REPORT | SALES REPORT | EXIT

SUPPLIER DETAILS

SUPPLIER DETAILS

Date: 21.03.2020

Supplier Code: S0001

Name: janani

Company Name: vv traders

Address: 12 madukari market road

Mobile No: 9685748574

Email Id: janani@gmail.com

Company Website: www.vvtraders.com

CLEAR | SAVE | SEARCH | UPDATE | EXIT

curdate	scode	sname	compname	compaddr	mno	email	we
21.03.2020	S0001	janani	vv traders	12 madukari mar...	9685748574	janani@gmail.com	ww
05-04-2021	S0002	saran	olivya furniture	82,north streettiru...	8978976899	olivyafumiture@g...	oliv
03-05-2021	S0003	rajesh	selvi furniture	33,south street,th...	8987676789	rajesh@gmail.com	ww
04-05-2021	S0004	saranya	vivek furniture	11,raja street,	7558147817	kasaranya1999...	ww

PRODUCT DETAILS

FURNITURE SHOWROOM

PRODUCT DETAILS

Date: 21.03.2020

Product Code: P0001

Item Name: sandol wood

Type of Item: door, window

CLEAR | SAVE | SEARCH | UPDATE | EXIT

curdate	pcode	iname	itype	qty
21.03.2020	P0001	sandol wood	doorwindow	9
25/03/2021	P0002	Computer Chair	Rolling Type	49

PURCHASE DETAILS

PURCHASE DETAILS

Date: 08-04-2021

Purchase ID	H0003	Item Type	doorwindow
Supplier Code	S0001	Company Name	vv traders
Name	janani	No Of Quantity	6
Product Code	P0001	Amount Per Quantity	500
Item Name	sandol wood	Total Amount	3000

CLEAR SAVE EXIT

	curdate	pcode	iname	itype
▶	21.03.2020	P0001	sandol wood	doorwindow
	25/03/2021	P0002	Computer Chair	Rolling Type
*				

	curdate	purid	scode	sname	pcode
▶	21.03.2020	H0001	S0001	janani	P0001
	25/03/2021	H0002	S0001	janani	P0002
*					

STOCK DETAILS

STOCK DETAILS

Product Name: sandol wood SEARCH EXIT

	curdate	purid	scode	sname	pcode	iname	itype	compname	noofqty
▶	21.03.2020	H0001	S0001	janani	P0001	sandol wood	doorwindow	vv traders	10
*									

Purchase Stock Qty: 10

Available Stock

	curdate	pcode	iname	itype	qty
▶	21.03.2020	P0001	sandol wood	doorwindow	4
*					

CUSTOMER DETAILS

FURNITURE SHOWROOM

CUSTOMER DETAILS

Date: 21.03.2020 | Address: collector office

Customer ID: C0002 | Mobile No: 7485967485

Name: kalpana | Email Id: kalpana@gmail.com

CLEAR | SAVE | SEARCH | UPDATE | EXIT

curdate	ccode	cname	caddr	cmno	emailid
11.03.2020	C0001	rajkumar	malumachampatty	9942964888	rajkumar@gmail....
21.03.2020	C0002	kalpana	collector office	7485967485	kalpana@gmail.c...
25/03/2021	C0003	nagarjun	12 kk colony sulur	9685848596	nagarjun@gmail....

BILLING DETAILS

BILLING DETAILS Time: 19:17:59

Date: 22-04-2021 | Item Name: Computer Chair

Bill ID: B0004 | ItemType: Rolling Type

Customer Code: C0001 | No Of Quantity: 20

Customer Name: rajkumar | Amount Per Quantity: 2000

Purchase ID: H0002 | GST: 10

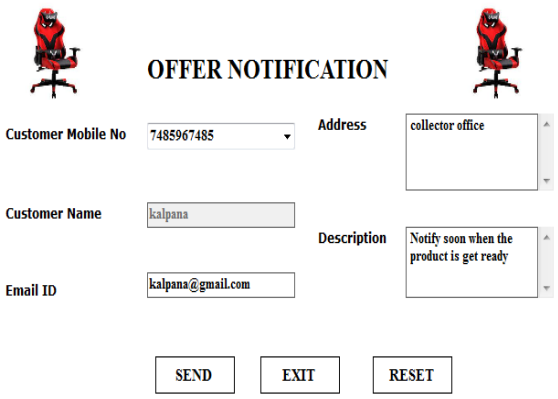
Total Amount: 44000

CLEAR | SAVE | EXIT

curdate	purid	scode	sname	pcode
21.03.2020	H0001	S0001	janani	P0001
25/03/2021	H0002	S0001	janani	P0002

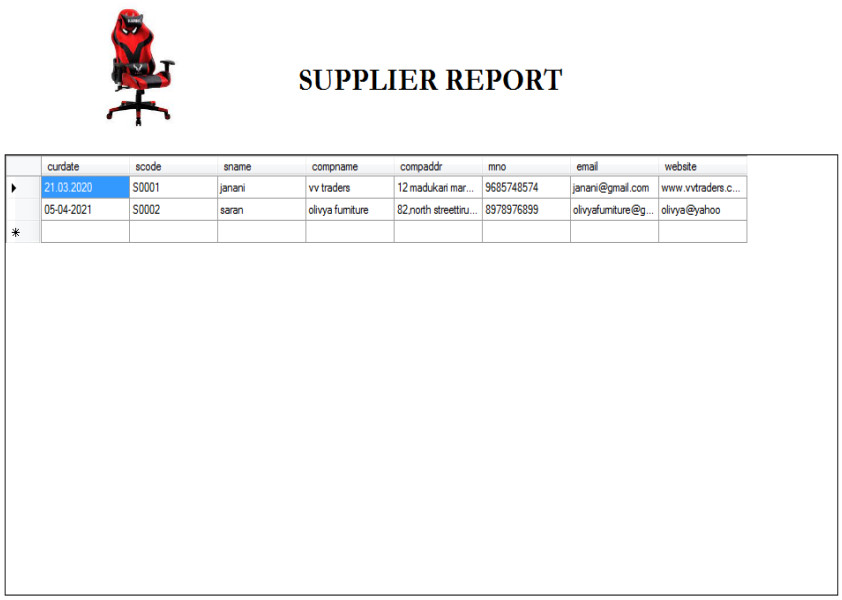
curdate	billid	ccode	cname	purid
21.03.2020	B0001	C0001	rajkumar	H0001
25/03/2021	B0002	C0003	nagarjun	H0002
05-04-2021	B0003	C0003	nagarjun	H0001

OFFER NOTIFICATION



REPORTS

SUPPLIER REPORT



PRODUCT REPORT

Product Report

	curdate	pcode	iname	itype	qty
▶	21.03.2020	P0001	sandol wood	doorwindow	4
	25/03/2021	P0002	Computer Chair	Rolling Type	49
*					

PURCHASE REPORT

Purchase Report

	curdate	purid	scode	sname	pcode	iname	itype	compname	noofqty
▶	21.03.2020	H0001	S0001	janani	P0001	sandol wood	doorwindow	vv traders	10
	25/03/2021	H0002	S0001	janani	P0002	Computer Chair	Rolling Type	vv traders	50
*									

CUSTOMER REPORT

Customer Report

	curdate	ccode	cname	caddr	cmno	emailid
▶	11.03.2020	C0001	rajkumar	malumachampatty	9942964888	rajkumar@g
	21.03.2020	C0002	kalpana	collector office	7485967485	kalpana@gr
	25/03/2021	C0003	nagarjun	12 kk colony sulur	9685848596	nagarjun@g
*						

BILLING REPORT

BILLING REPORT

Bill ID B0003 SEARCH VIEW ALL EXIT

	curdate	billid	ccode	cname	purid	iname	itype	noofqty	amtqty
▶	21.03.2020	B0001	C0001	rajkumar	H0001	sandol wood	doorwindow	1	250
	25/03/2021	B0002	C0003	nagarjun	H0002	Computer Chair	Rolling Type	1	2000
	05-04-2021	B0003	C0003	nagarjun	H0001	sandol wood	doorwindow	5	250
*									

Total Amount 3645

SYSTEM TESTING

CHAPTER-V

SYSTEM TESTING

SYSTEM TESTING

System testing is the state of implementation, which is aimed at ensuring that the system works accurately and efficiently. It certifies that the whole set of programs hang together. System testing requires a test plans, that consists of several key activities and steps for run program, string, system and user acceptance testing.. Testing phase is the development phase that validates the code against the functional specifications. The objective of testing is to discover errors. To fulfil this objective a series of test step such as the unit test, integration test, validation and system test where planned and executed.

UNIT TESTING

The unit testing was conducted during the development phase. we test their specific functionality individually or with other units. However, unit testing is designed to test small pieces of functionality rather than the system as a whole. This allows us to conduct the first round of testing to eliminate bugs before the other major tests**.**Unit testing comprises the set of tests performed by an individual prior to integration of the unit into large system. The situation is illustrated in as follows Coding-> Debugging ->Unit Testing -> Integration testing->Output Testing

The four categories of test that a programmer will typically perform on a program unit

- Functional test
- Performance test
- Structure test

Functional test involve exercising the code with nominal input values for which the expected results are known as well as boundary values and special values. Performance testing determines the amount of execution time spent in various parts of unit program through put and response time and device utilization by the program. Structured testing is concerned with a exercising the internal logic of a program and traversing paths. Functional testing, ,performance testing are referred as “black box” testing and structure testing is referred as “white box” testing

OUTPUT TESTING

Asking the user about the format required by them tests the output generated by the system under consideration .It can be done in two ways, One on screen and other on printer format. The output format on the screen is found to be correct as the format designed In system test.

INTEGRATED TESTING

Integrated testing is a systematic technique for constructing tests to uncover errors associated with interface .Objective is to take unit tested modules and build a program structure that has been dictated by design There is a often attempt non – incremental integration that is to construct the program using a "Big bang "approach. All components are combined in advance. The entire program is tested as a whole incremental integration is the antithesis of the bang approach.

- The main control module is used to test drive and stubs are substituted for all components directly subroutine to the main control module.
- Depending on the integration approach selected subroutine stubs are replaced on at a time with actual components.
- Tests are conducted as each component is integrated.
- On completion of each set of tests another stubs is replaced with the real components.

ACCEPTANCE TESTING

Acceptance testing involves planning an execution of a functional test, performance test and to verify that the implemented system satisfies the requirement.The acceptance testing is the final stage of the user the various possibilities of the data are entered and the results are tested.

VALIDATION TESTING

At the end of Integration Testing, software is completely assembled as a package, interfacing errors have been uncovered and correction testing begins. Software Testing and Validation is achieved through serried of black box tests that demonstrate conformity with the requirements. A test plan outlines the classes of tests to be conducted and a test procedure defines specific test cases that will be used to demonstrate conformity with requirements.

SYSTEM IMPLEMENTATION

CHAPTER-VI
SYSTEM IMPLEMENTATION

Implementation is a process of ensuring that the information system is operational. In the project where the theoretical design is turned into a working system. The most crucial stage is achieving a successful new system and giving a user confidence in that the new system will work efficiently and effectively in the implementation stage. The stage consist of Testing a developed program with sample data,Detection and correction of errorCreating whether the system meets a user requirements,Making necessary changes as desired by users.Training user personal.

IMPLEMENTATION PROCEDURES

The implementation phase is less creative than system design. A system design may be dropped at any time prior to implementation, although it becomes more difficult when it goes to the design phase. The final report of the implementation phase includes procedural flowcharts, record layouts, and a workable plan for implementing the candidate system design into a operational design.

USER TRAINING

- End-user training is an important part of the computer-based information system development, which must be provided to employees to enable them to do their own problem solving.
- User training involves how to operate the equipment, troubleshooting the system problem, determining whether a problem that arose is caused by the equipment or software.
- Most user training deals with the operation of the system itself. The training courses must be designed to help the user with fast mobilization for the organization.

SYSTEM MAINTENANCE

Maintenance is expensive .One way to reduce the maintenance costs are through maintenance management and software modification audits .Maintenance means restoring something to its original conditions. Enhancement means adding, modifying the code to support the changes in the user specification. System maintenance conforms the system to its original requirements and enhancement adds to system capability by incorporating new requirements.

Types of maintenance are

- Perfective maintenance
- Preventive maintenance

PERFECTIVE MAINTENANCE

It mainly deals with implementing new or changed user requirements. It involves making functional enhancements to the systm in addition to the activities to increase the systems performance even when the changes have not been suggested by faults.

PREVENTIVE MAINTENANCE

Changes made to the system to avoid future problems. Any changes can be made in the future and our project can adopt the changes.

CONCLUSION

CHAPTER-VI
CONCLUSION

The main objective of the project is to bring a full-fledged computerized organization, and to enable the transaction details to maintain records, which makes easier. Thus, the proposed system has been developed with good amount of flexibility without compromising on the response time. Computerization of the entire system will enhance more accuracy and reduces major part of clerical works. Fast, clear and legible reports can be generated without any ambiguity. Integrated database design and ease of maintenance is a major advantage of the system. User friendliness is a unique feather of the system. Hence by developing a system that is user-friendly in nature, many users are able to work on the system with little of computer knowledge and training.

SCOPE OF FUTURE ENHANCEMENTS:

The project has been developed and the objectives are achieved successfully. The project has been developed with front end as VB.Net and backend as SQL Server. The frontend can also be changed. ASP.Net can replace the front-end tool such as HTML and CSS for more speed. The system is currently developed and ready for implementation to include the system is highly feasible and user friendly. To provide better facility regarding security, it uses security provider software and we access any place and any where use that. It can have an enhancement on proper in the future according to the user's requirements.

BIBLIOGRAPHY

BIBLIOGRAPHY

- .Elias Awath, "**SYSTEM ANALYSIS AND DESIGN**",Tata Mc Graw Hill Publication, Sixth Edition,2003
- .S.Ramachandran,"**COMPUTER AIDED DESIGN**", Air Walk Publication, Third Edition,2003
- .Richard Fairley,"**SOFTWARE ENGINEERING CONCEPTS**",Tata Mc Graw Hill Publication, Second Edition,1997
- .Distributed .NET Programming in VB .NET by Tom Barnaby
- Professional VB.NET, 2nd Edition by Fred Barwell, et al
- The .NET Languages: A Quick Translation Guide by Brian Bischof
- Programming VB.NET: A Guide for Experienced Programmers by Gary Cornell, Jonathan Morrison
- Learning Visual Basic.NET Through Applications by Clayton Crooks II
- Visual Basic .NET How to Program (2nd Edition) by Harvey M. Deitel, Paul J. Deitel, Tem R. Nieto

WEBSITES

- https://www.tutorialspoint.com/vb.net/
- https://en.m.wikipedia.org/wiki/visualbasic.net
- https://www.techopedia.com
- https://www.javatpoint.com
- https://www.techonthenet.com
- https://www.smartdraw.com