

Just shop shopping is an online destination for shopper to search and select their favorite products, add them to their cart and get them home delivered after the confirmation of the payment. This website advertises the products for shopping. To buy products, customer has to create an account with the website. Those who do not have an account, they can only view the available product. They can't buy it. Once the customer has created account, not only he can view the products, he can also add the product to the cart and can place an order to buy those products. This application then generates bill for that particular customer. After the confirmation, the customer has to enter his credit card details to buy those products. This project is a business to consumer, e-commerce web site. Users can search among the varieties of products and choose among their favorite brands. The consumer can place orders as required and can specify the delivery address if other than the address registered earlier with the website. These include laptop, hard disk, clothing, books, foot wear, gaming devices, kitchen appliances and many more.



Mehtab Alam

JUST SHOP-Shopping

The shopping destination



Pursuing PhD Scholar in Computer Science and Engineering, with Master of Technology in Information Security and Cyber Forensics and Bachelor of Technology in Information Technology. Residing in New Delhi, India



Mehtab Alam

JUST SHOP-Shopping

FOR AUTHOR USE ONLY

Mehtab Alam

JUST SHOP-Shopping

The shopping destination

FOR AUTHOR USE ONLY

LAP LAMBERT Academic Publishing

Imprint

Any brand names and product names mentioned in this book are subject to trademark, brand or patent protection and are trademarks or registered trademarks of their respective holders. The use of brand names, product names, common names, trade names, product descriptions etc. even without a particular marking in this work is in no way to be construed to mean that such names may be regarded as unrestricted in respect of trademark and brand protection legislation and could thus be used by anyone.

Cover image: www.ingimage.com

Publisher:

LAP LAMBERT Academic Publishing

is a trademark of

International Book Market Service Ltd., member of OmniScriptum Publishing Group

17 Meldrum Street, Beau Bassin 71504, Mauritius

Printed at: see last page

ISBN: 978-620-3-58124-9

Copyright © Mehtab Alam

Copyright © 2021 International Book Market Service Ltd., member of OmniScriptum Publishing Group

FOR AUTHOR USE ONLY

JUST SHOP

The project entitled “Just Shop” enables customer to buy products and accessories from the comfort of their homes through online website. This website advertises the products for shopping. To buy products, customer has to create an account with the website. Those who do not have an account, they can only view the available product. They can't buy it. Once the customer has created account, not only he can view the products, he can also add the product to the cart and can place an order to buy those products. This application then generates bill for that particular customer. After the confirmation, the customer has to enter his credit card details to buy those products.

This project is a business to consumer, e-commerce web site. Users can search among the varieties of products and choose among their favorite brands. The consumer can place orders as required and can specify the delivery address if other than the address registered earlier with the website.

Online Shopping is a well-known company serving India and other countries, with a variety in all the available products. These include laptop, hard disk, clothing, books, foot wear, gaming devices, kitchen appliances and many more.

This project is a B2C (business to costumer) e- commerce website, which means that online money transactions is taking place between consumer and business house.



TABLE OF CONTENT

| S.No. | Content | Page No. |
|--------------|---|-----------------|
| I | List of Figures | 9 |
| II | List of Tables | 10 |
| 1 | OVERVIEW OF THE PROJECT | 11-16 |
| 1.1 | Objective | 12 |
| 1.2 | Introduction <ul style="list-style-type: none"> • User Account Creation • User Login • Product Categories • Accessories • Search Module • Cart • Payment Module • Stock | 13 |
| 1.3 | Problem Statement | 16 |
| 2 | SYSTEM STUDY AND ANALYSIS | 17-25 |
| 2.1 | Introduction | 18 |
| 2.2 | Feasibility Study <ul style="list-style-type: none"> • Technical Feasibility • Operational Feasibility • Economical Feasibility • Legal Feasibility • Schedule Feasibility | 19 |
| 2.3 | Other Feasibility Factors <ul style="list-style-type: none"> • Market and Real Estate Feasibility • Resource Feasibility • Cultural Feasibility • Financial Feasibility | 21 |
| 2.4 | Output | 22 |
| 2.5 | Hardware And Software Requirements <ul style="list-style-type: none"> • Hardware Requirements • Software Requirements | 23 |
| 2.6 | Existing System <ul style="list-style-type: none"> • Drawback Of Existing System | 24 |
| 2.7 | Proposed System <ul style="list-style-type: none"> • Advantages Of Proposed Systems | 25 |

| | | |
|------|---|-------|
| 3 | SYSTEM DESIGN | 26-56 |
| 3.1 | Introduction <ul style="list-style-type: none"> • Logical Design • Physical Design • Design/Specification Activities | 27 |
| 3.2 | Input Design | 29 |
| 3.3 | Output Design | 29 |
| 3.4 | Use Case Diagram <ul style="list-style-type: none"> • Manage Account • Manage Products • Search Products • Place Order • Checkout • Courier Company | 30 |
| 3.5 | Cost Effort Estimation | 36 |
| 3.6 | Class Diagram <ul style="list-style-type: none"> • Class Diagram for Just Shop | 40 |
| 3.7 | Entity Relationship Diagram | 42 |
| 3.8 | Data Flow Diagram <ul style="list-style-type: none"> • Symbols Used In DFD • Context Level DFD | 44 |
| 3.9 | Context Level DFD | 46 |
| 3.10 | Level 1 DFD Administrator <ul style="list-style-type: none"> • New User Registration • Login • User Activity | 47 |
| 3.11 | Database Design | 50 |
| 3.12 | Database Table Design <ul style="list-style-type: none"> • User Registration • Feedback Database • Cartitem.cs • Products.cs • ShoppingCart.cs | 51 |
| 3.13 | Software Description <ul style="list-style-type: none"> • The Control Properties • The Text Box • The Label • The Command Buttons • The Picture Box • The Image Box • The List Box • The Option Box • The Directory List Box | 55 |
| 3.14 | Database File | 56 |

| | | |
|-----|---|-------|
| 4 | SYSTEM TESTING AND IMPLEMENTATIONS | 57-66 |
| 4.1 | Testing | 58 |
| 4.2 | Types Of Testing <ul style="list-style-type: none"> • Unit Testing • Integration Testing • Validation Testing • Recovery Testing • Security Testing • Stress Testing • Black Box Testing • Test Data Output | 58 |
| 4.3 | Screen Shots <ul style="list-style-type: none"> • Home Page • Cart Empty • Clothes-Sub Menu • Games-Sub Menu • Items Sub Sub Menu • Register • Login • Cart Full • Contact Us | 60 |
| 4.4 | Conclusion | 65 |
| 4.5 | References | 66 |

LIST OF FIGURES

| Figure No. | Description | Page No. |
|-------------------|---|-----------------|
| 1 | An Electronic Cart | 5 |
| 2 | Manage Account-Use Case | 30 |
| 3 | Manage Products-Use Case | 31 |
| 4 | Search Products-Use Case | 32 |
| 5 | Place Order-Use Case | 33 |
| 6 | Checkout-Use Case | 34 |
| 7 | Courier Company-Use Case | 35 |
| 8 | Class Diagram | 40 |
| 9 | Just Shop-Class Diagram | 41 |
| 10 | Symbols used in Entity Relationship Diagram | 42 |
| 11 | Just Shop-ER Diagram | 43 |
| 12 | Symbols Used In DFD | 45 |
| 13 | Context level DFD | 46 |
| 14 | New User Registration-DFD | 47 |
| 15 | Login-DFD | 48 |
| 16 | User Activity-DFD | 49 |
| 17 | Home Page-Snapshot | 60 |
| 18 | Cart Empty-Snapshot | 60 |
| 19 | Clothes-Sub Menu-Snapshot | 61 |
| 20 | Games-Sub Menu-Snapshot | 61 |
| 21 | Items-Sub sub Menu-Snapshot | 62 |
| 22 | Register-Snapshot | 62 |
| 23 | Login-Snapshot | 63 |
| 24 | Cart Full-Snapshot | 63 |
| 25 | Contact Us-Snapshot | 64 |

LIST OF TABLES

| Table No. | Description | Page No. |
|------------------|----------------------------|-----------------|
| 1 | Table of Content | 6 |
| 2 | List Of Figures | 9 |
| 3 | List of Tables | 10 |
| 4 | Manage Account-Use Case | 30 |
| 5 | Manage Products-Use Case | 31 |
| 6 | Search Products-Use Case | 32 |
| 7 | Place Order-Use Case | 33 |
| 8 | Checkout-Use Case | 34 |
| 9 | Courier Company-Use Case | 35 |
| 10 | Basic COCOMO | 36 |
| 11 | Attributes Ratings | 37 |
| 12 | Intermediate COCOMO | 38 |
| 13 | User Registration-Database | 51 |
| 14 | Feedback-Database | 51 |
| 15 | Cartitem.cs-Database | 52 |
| 16 | Products.cs-Database | 53 |
| 17 | ShoppingCart.cs-Database | 54 |

OVERVIEW OF THE PROJECT

1.1 OBJECTIVE

The goal of the introduction of E-commerce or E-shopping is to shop while in the comfort of your own home, without having to step out of the door. E-commerce led to the increase of the knowledge of the users as now they have a wide range of products to choose and select from. Earlier the users had only a few branded outlets to buy the products from. They had to pay whatever price was fixed by the store keeper. But now the users can buy the desired products after comparing a number of Just Shop websites or portals active now a days.

The other goals of introduction of the system are:

- i. To be able to easily save money and compare prices from website to website.
- ii. Online resellers tend to sell at a lower price due to less overhead expenses.
- iii. The users get their products delivered to their desired addresses.
- iv. The user can also buy the products and get them delivered to the desired address as a gift or present for their family and friends.
- v. Just Shop also provides with 30 Days Money Back Guarantee as well as 100% customer satisfaction.
- vi. The products sold are all branded and new unlike some stores which sometimes sell old products to the customers.

The system came as a substitute to the earlier shopping practice of going from one store to another in search of the desired products.

1.2 INTRODUCTION

The project entitled “Just Shop” enables customer to buy products and accessories from the comfort of their homes through online website. This website advertises the products for shopping. To buy products, customer has to create an account with the website. Those who do not have an account, they can only view the available product. They can’t buy it. Once the customer has created account, not only he can view the products, he can also add the product to the cart and can place an order to buy those products. This application then generates bill for that particular customer. After the confirmation, the customer has to enter his credit card details to buy those products.

This project is a business to consumer, e-commerce web site. Users can search among the varieties of products and choose among their favorite brands. The consumer can place orders as required and can specify the delivery address if other than the address registered earlier with the website.

Just Shop is a well-known company serving India and other countries, with a variety in all the available products. These include laptop, hard disk, clothing, books, foot wear, gaming devices, kitchen appliances and many more.

This project is a B2C (business to costumer) e- commerce website, which means that online money transactions is taking place between consumer and business house.

This project is aimed at developing a Website that depicts online shopping of many products for example clothes, footwear, mobiles, books, novels, kitchen ware etc. from a single website and purchasing using the Payment Gateway or Cash On Delivery (COD).

Using this website, companies can improve the efficiency of their services. Just Shop is one of the applications to improve the marketing of the company’s products. This web application involves all the features of the online shopping.

This project is modularized as the following:

- i. User Account Creation (Sign Up)
- ii. User Login (Login)
- iii. Product Categories
- iv. Accessories
- v. Search Module
- vi. Cart
- vii. Payment Module
- viii. Stock

1.2.1 User Account Creation (Sign Up)

The Sign Up module is for the users who do not have an existing account on the website. Here user is allowed to create an account to login. The account creation is done by filling the registration form with user details such as First Name, Last Name, Email, Username, Password etc.

1.2.2 User Login (Login)

There are two field in this module, namely, Username and Password. This module helps the user to present his or her identity to the website. The user can Log In once they have Signed Up in the above module. If the username and password entered by the user are correct then the user is directed to next page or the home page. Login feature helps the user to save his or her cart, i.e. the user can save the items in there and return to them on the next login.

1.2.3 Product Categories

This module has information regarding all the products available on the website such as its name, price information, its features, its categories etc. The USER can view the products, and can add them to his her cart. Product categories include Clothes, Mobiles, Footwear, Computer, Watches N Bags, Books, Home N Kitchen, Toys, and Gaming.

1.2.4 Accessories

This module consists of various available accessories on various products with its name and picture, price information etc.

1.2.5 Search Module

This module helps the customer to ease his or her search based on his interest or the product of need. The search can be done on the basis of product name.

1.2.6 Cart

The user can select any number of products and add them to his or her cart one by one. Bulk addition is not allowed. The user can also remove products from the cart if he or she wants it no more, or got a better product.

1.2.7 Payment Module

This module describes the mode of payment to be selected by the customer before the products are sent to the desired address of the user. The payment information can include information like the model purchased, quantity, mode of payment etc. We support two types of payment, COD or online via PayPal or through Debit or credit card.

1.2.8 Stock

This module gives the information regarding the products availability at that time.

1.3 PROBLEM STATEMENT

Customer can browse through the product catalog and add the items to shopping cart. He/she can proceed to checkout as long as his/her shopping cart is not empty. Customer will require to login to the system when he/she proceeds to checkout, or the user can create an account if not yet having one. The order can be paid via COD, PayPal or through credit card or Debit Card.

Customer need to provide full name, email address, phone number, username, gender, the billing address details when creating an account.

Customer can login to the system to maintain their account information, such as changing phone number, address, etc. and check the status of their orders.

Upon order received, the sales staff will process the order by charge to customer's credit card. Once the order has been charged, the order will then be marked as paid and passed to courier company to be delivered to the customer. If the items customer ordered is out of stock, then the order will mark as on hold.

Once the item becomes in stock, the order will passed to the courier company for delivery. Courier Company will pack the item with standard packaging, but if the order is marked as gift, then the items will pack as gifts.

If the items arrived with damage, customer can return it by registering a complaint in the online shop. Courier Company will collect the item from customer and sales staff will refund the money for that item.

Marketing staff is responsible to maintain the product catalog. They can also setup the promotion item list and send promotion email to customer.

SYSTEM STUDY AND ANALYSIS

2.1 INTRODUCTION

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

Here in the project E-Ticketing, a detailed study of existing system is carried along with all the steps in system analysis. An idea for creating a better project was carried and the next steps were followed.

2.2 FEASIBILITY STUDY

An important outcome of the preliminary investigation is the determination that the system requested is feasible. 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. It involves preliminary investigation of the project and examines whether the designed system will be useful to the organization. Months or years of effort, thousand for millions of money and untold professional embarrassment can be averted if an in-conceived system is recognized early in the definition phase.

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of the existing business or proposed venture, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to be attained. As such, a well-designed feasibility study should provide a historical background of the business or project, description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations.

Generally, feasibility studies precede technical development and project implementation.

The different types of feasibility are:

- i. Technical feasibility
- ii. Operational feasibility
- iii. Economical feasibility
- iv. Legal feasibility
- v. Schedule feasibility

2.2.1 Technical feasibility

Technical Feasibility deals with the hardware as well as software requirements. Technology is not a constraint to type system development. We have to find out whether the necessary technology, the proposed equipment's have the capacity to hold the data, which is used in the project, should be checked to carry out this technical feasibility.

The technical feasibility issues usually raised during the feasibility stage of investigation includes these.

This software is running in windows 7 Operating System, which can be easily installed.

The hardware required is Pentium based server.

The system can be expanded.

2.2.2 Operational Feasibility

This feasibility test asks if the system will work when it is developed and installed.

Operational feasibility in this project:

- The proposed system offers greater level of user-friendliness.
- The proposed system produces best results and gives high performance.
- It can be implemented easily.

So this project is operationally feasible.

2.2.3 Economical feasibility

Economical Feasibility deals about the economical impact faced by the organization to implement a new system. Financial benefits must equal or exceed the costs. The cost of conducting a full system, including software and hardware cost for the class of application being considered should be evaluated.

Economic Feasibility in this project:

- The cost to conduct a full system investigation is possible.
- There is no additional manpower requirement.
- There is no additional cost involved in maintaining the proposed system.

2.2.4 Legal feasibility

Determines whether the proposed system conflicts with legal requirements, e.g. a data processing system must comply with the local Data Protection Acts.

2.2.5 Schedule feasibility

A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is. Given our technical expertise, are the project

deadlines reasonable? Some projects are initiated with specific deadlines. You need to determine whether the deadlines are mandatory or desirable.

2.3 OTHER FEASIBILITY FACTORS

2.3.1 Market and real estate feasibility

Market feasibility studies typically involve testing geographic locations for a real estate development project, and usually involve parcels of real estate land. Developers often conduct market studies to determine the best location within a jurisdiction, and to test alternative land uses for given parcels. Jurisdictions often require developers to complete feasibility studies before they will approve a permit application for retail, commercial, industrial, manufacturing, housing, office or mixed-use project. Market Feasibility takes into account the importance of the business in the selected area.

2.3.2 Resource feasibility

This involves questions such as how much time is available to build the new system, when it can be built, whether it interferes with normal business operations, type and amount of resources required, dependencies,

2.3.3 Cultural feasibility

In this stage, the project's alternatives are evaluated for their impact on the local and general culture. For example, environmental factors need to be considered and these factors are to be well known. Further an enterprise's own culture can clash with the results of the project.

2.3.4 Financial feasibility

In case of a new project, financial viability can be judged on the following parameters:

Total estimated cost of the project

Financing of the project in terms of its capital structure, debt equity ratio and promoter's share of total cost

Existing investment by the promoter in any other business

Projected cash flow and profitability

2.4 **OUTPUT**

The feasibility study outputs the **feasibility study report**, a report detailing the evaluation criteria, the study findings, and the recommendations.

2.5 HARDWARE AND SOFTWARE REQUIREMENT

2.5.1 Hardware Requirements

Hardware requirement are the basic need of the system or the package, which is been developed and will be deployed upon the system, which should have these basic components or fulfill these basic hardware needs of these package.

The following hardware is recommended for the user.

| | |
|--------------|-----------------------------------|
| Processor | Intel Pentium IV 2.4 GHZ or above |
| Clock speed | 500 MHZ |
| RAM | 32MB or above |
| System bus | 32 bits |
| Monitor | Color monitor |
| Hard disk | 8 GB or above |
| CD Drive | Any CD ROM |
| Input device | Key board, Mouse |

2.5.2 Software Specification

| | |
|---------------------|-----------------------------|
| Operating System | MS WINDOWS XP SP2 |
| Front End | Visual Studio 2012, ASP.Net |
| Back End | SQL Server |
| Additional Software | DOTNet framework 3.5 |

2.6 EXISTING SYSTEM

Existing system refers to the system that is being followed till now or being followed now. The existing system requires more computational time, more manual calculations, and the complexity involved in Selection of features is high. The other disadvantages are lack of security of data, lack of security of money, deficiency of data accuracy, time consuming etc. To avoid all these limitations and make the working more accurately the system needs to be computerized and manual work load has to be minimized. Here in the Just Shop website, a detailed study of existing system is carried along with all the steps in system analysis.

2.6.1 DRAW BACKS OF EXISTING SYSTEM

Here in the Just Shop Website, a detailed study of existing system is carried along with all the steps in system analysis. An idea for creating a better project was carried and the next steps were followed.

- i. Lack of security of data
- ii. Lack of security of Money
- iii. More man power
- iv. Many a times the store keepers hide the service tax (VAT etc) which helps in the working of the government
- v. Time consuming
- vi. Consumes large volume of paper work
- vii. Needs manual calculations
- viii. A number of In-Between people (suppliers, Distributers) are involved. They include their own profits and perks in the Cost Price.
- ix. Sometimes the store keepers sell duplicate items for the cost of original ones.

To avoid all these limitations and make the shopping and the working of the system more accurate, the system needs to be computerized.

2.7 PROPOSED SYSTEM

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work. The existing system has several disadvantages and many more difficulties to work well. The proposed system tries to eliminate or reduce these difficulties up to some extent. The proposed system will help the user to reduce the workload and mental conflict. The proposed system helps the user to work user friendly and he can easily do his jobs without time lagging.

2.7.1 ADVANTAGES OF PROPOSED SYSTEM

The website has a brilliant design and implementation. The website requires very low system resources and the website will work in almost all configurations with an internet connection. It has got following features:

- i. Insure data accuracy
- ii. Insure secure money transfer
- iii. Minimize manual data entry
- iv. Greater efficiency
- v. Better Service
- vi. Minimum time required
- vii. Just Shop transactions include system generated invoices that include taxes (VAT)
- viii. Minimum paper work
- ix. No In-Between peoples (Distributers, Suppliers), the goods are directly supplied from the manufacturers to the end users
- x. 24x7 Customer Support available & 30 days Money Back Guarantee
- xi. All items are branded and brand new
- xii. COD available for Non Card Holders
- xiii. Handsome offers on selected items

SYSTEM DESIGN

3.1 INTRODUCTION

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

3.1.1 Logical Design

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

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

3.1.2 Physical Design

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

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

3.1.3 Design/Specifications Activities

- i. Concept formulation.
- ii. Problem understanding.
- iii. High level requirements proposals.
- iv. Feasibility study.
- v. Requirements engineering.
- vi. Architectural design.

3.2 INPUT DESIGN

Input Design deals with what data should be given as input, how the data should be arranged or code, the dialog to guide the operating personnel in providing input, methods for preparing input validations and steps to follow when error occur. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user friendly GUI (Graphical User Interface). The goal of the designing is to make the interface as simple and attractive as possible. The user can browse through the items available on the website, add or remove them to and from the cart as required. It provides invoices also.

When the user click on the Checkout button the session checks for a valid login. If the user has logged in it goes to the payment gateway else it redirects to the login page. After a successful login the payment gateway launches.

3.3 OUTPUT DESIGN

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

3.4 USE CASE DIAGRAM

3.4.1 MANAGE ACCOUNT

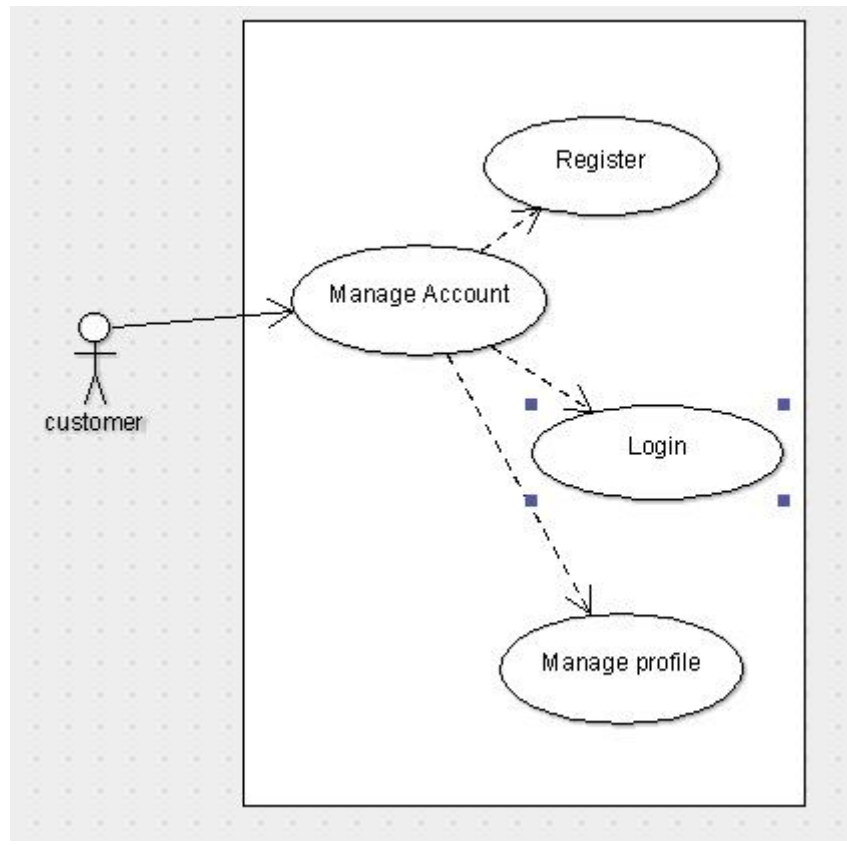


Figure: 3.1

| | |
|----------------|--|
| Use Case Name | <i>Account Management</i> |
| Pre-Condition | New User Existing User |
| Basic Path | Successful Login Successful Registration |
| Alternate Path | Incorrect Username/Password combination Incorrect Email Id Username Already exists |
| Post-Condition | Successful Login-User Goes to Home Page Successful Registration-Success Message |
| Exception Path | User Already Exists Password Recovery |

Table 3.1

3.4.2 MANAGE PRODUCTS

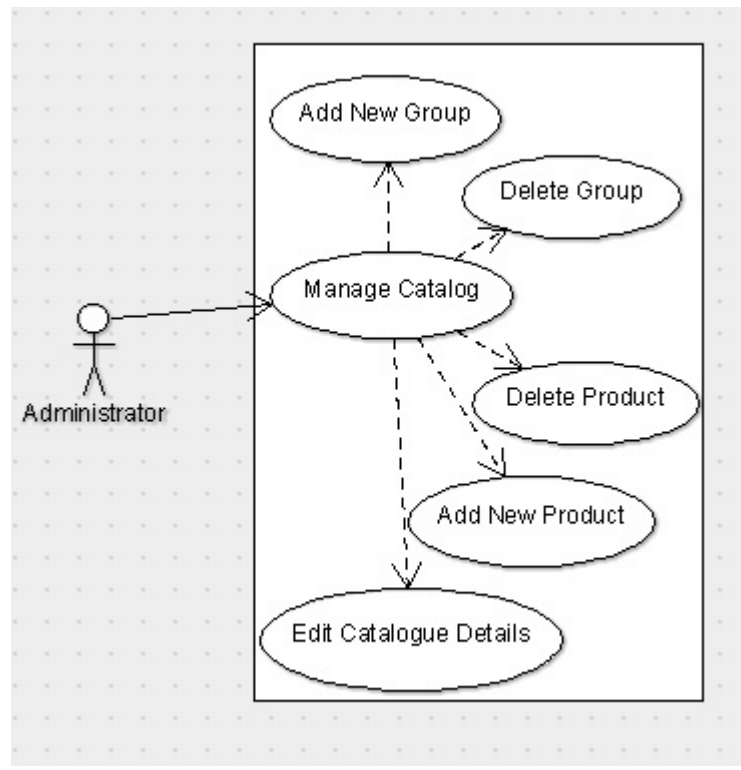


Figure: 3.2

| | |
|----------------|---|
| Use Case Name | <i>Manage Products</i> |
| Pre-Condition | Existing User (Admin) |
| Basic Path | Successful Login Add Edit Delete Products |
| Alternate Path | Incorrect Username/Password combination |
| Post-Condition | Successful Login Successful Product Addition/Deletion Etc. |
| Exception Path | User Does Not Exist Rights Not Available |

Table 3.2

3.4.3 SEARCH PRODUCTS

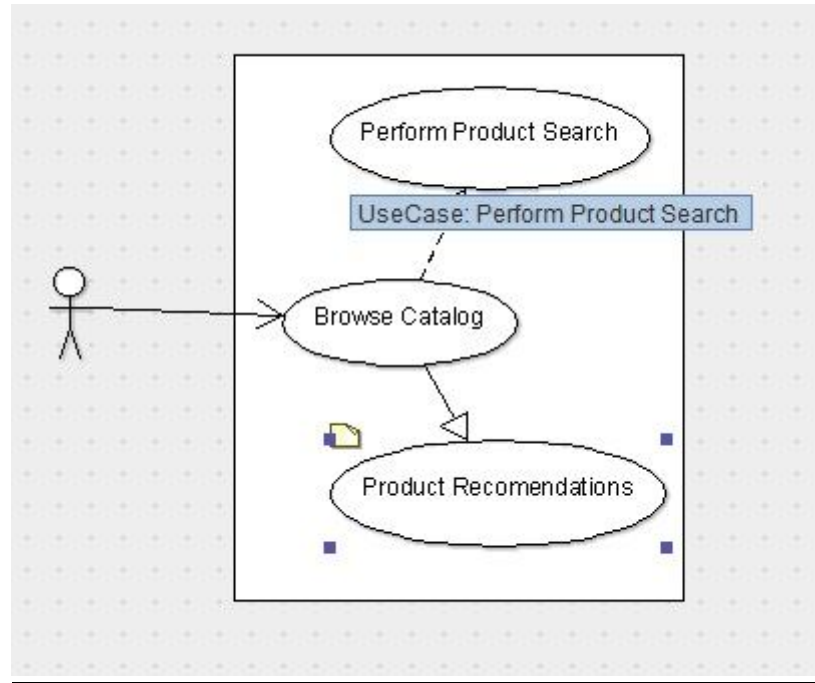


Figure: 3.3

| | |
|----------------|---|
| Use Case Name | <i>Search Products</i> |
| Pre-Condition | New User Existing User |
| Basic Path | Search Successful Items Displayed |
| Alternate Path | Search Returned Null Value Displaying Similar Products |
| Post-Condition | Search Successful Proceed to products |
| Exception Path | No Such Products Found |

Table 3.3

3.4.4 PLACE ORDER

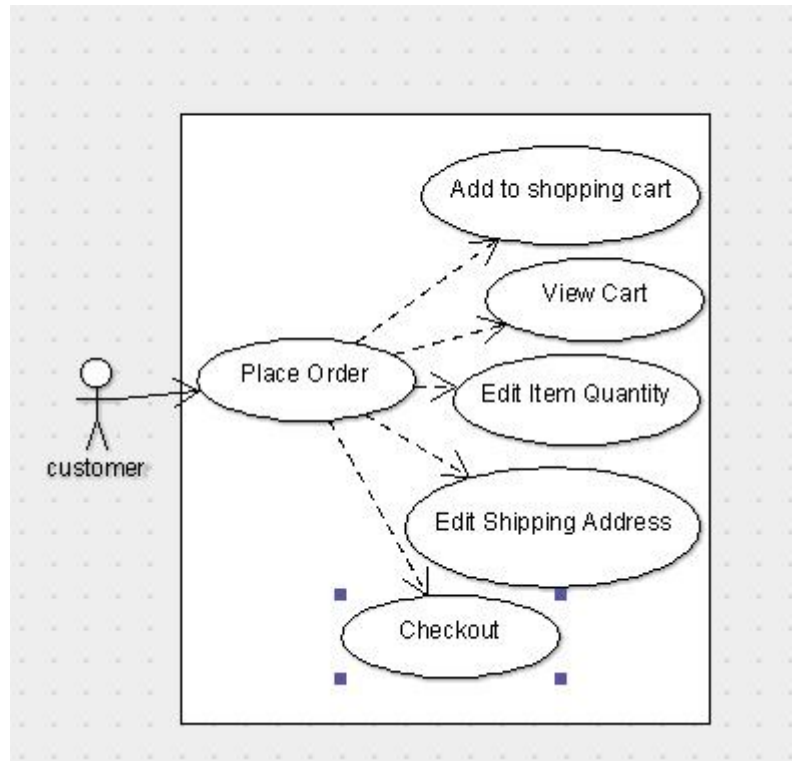


Figure: 3.4

| | |
|----------------|--|
| Use Case Name | <i>Place Order</i> |
| Pre-Condition | New User Existing User |
| Basic Path | Successful Login Adding Items To Cart View Cart Check Out |
| Alternate Path | Incorrect Username/Password combination Incorrect Email Id Username Already exists |
| Post-Condition | Successful Login Successful Registration Order Placed Successfully Check Out |
| Exception Path | User Already Exists Password Recovery Out Of Stock |

Table 3.4

3.4.5 CHECKOUT

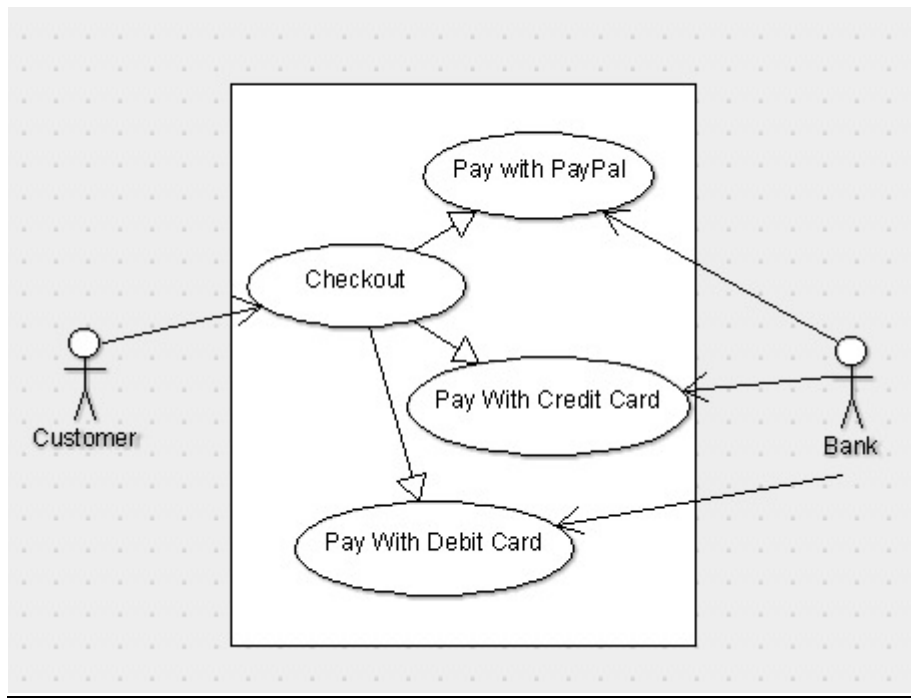


Figure: 3.5

| | |
|----------------|--|
| Use Case Name | <i>Check Out</i> |
| Pre-Condition | Existing User Items In Cart User Logged In |
| Basic Path | Successfully Connected To bank Successful Payment |
| Alternate Path | Invalid Bank Credentials Payment Failed/Retry |
| Post-Condition | Order Completed Successfully |
| Exception Path | Bank Returns Low Balance |

Table 3.5

3.4.6 COURIER COMPANY

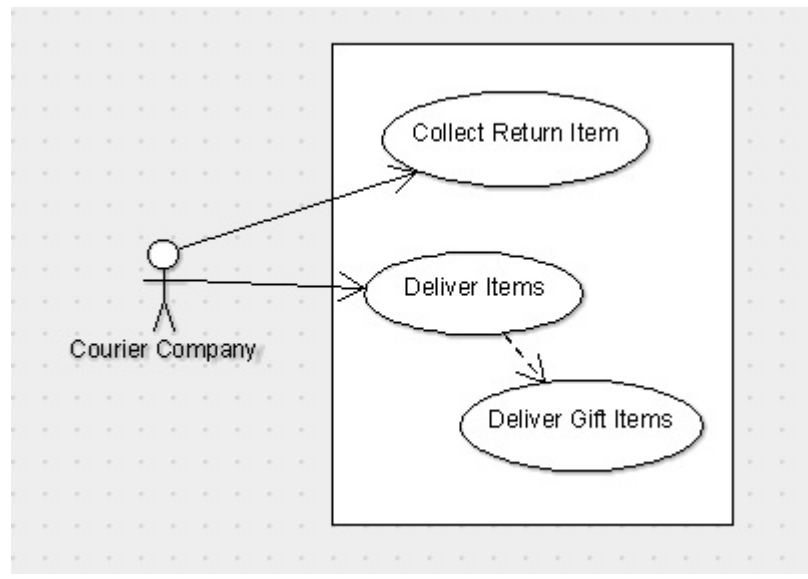


Figure: 3.6

| | |
|----------------|---|
| Use Case Name | <i>Courier Company</i> |
| Pre-Condition | List of Items To Be Delivered Delivery Address |
| Basic Path | Successfully Retrieved Lists/Addresses Delivery Successful |
| Alternate Path | Delivery Failed Address Not Found |
| Post-Condition | Order Delivery Completed Successfully |
| Exception Path | Delivery Delayed |

Table 3.6

3.5 COST/EFFORT ESTIMATION

Basic COCOMO

Basic COCOMO computes software development effort (and cost) as a function of program size. Program size is expressed in estimated thousands of source lines of code (SLOC)

COCOMO applies to three classes of software projects:

- Organic projects - "small" teams with "good" experience working with "less than rigid" requirements
- Semi-detached projects - "medium" teams with mixed experience working with a mix of rigid and less than rigid requirements
- Embedded projects - developed within a set of "tight" constraints. It is also combination of organic and semi-detached projects.(hardware, software, operational, ...)

The basic COCOMO equations take the form

$$\text{Effort Applied (E)} = a_b(\text{KLOC})^{b_b} \text{ [man-months]}$$

$$\text{Development Time (D)} = c_b(\text{Effort Applied})^{d_b} \text{ [months]}$$

$$\text{People required (P)} = \text{Effort Applied} / \text{Development Time [count]}$$

where, **KLOC** is the estimated number of delivered lines (expressed in thousands) of code for project. The coefficients a_b , b_b , c_b and d_b are given in the following table:

| Software project | a_b | b_b | c_b | d_b |
|------------------|-------|-------|-------|-------|
| Organic | 2.4 | 1.05 | 2.5 | 0.38 |
| Semi-detached | 3.0 | 1.12 | 2.5 | 0.35 |
| Embedded | 3.6 | 1.20 | 2.5 | 0.32 |

Table 3.7

Basic COCOMO is good for quick estimate of software costs. However it does not account for differences in hardware constraints, personnel quality and experience, use of modern tools and techniques, and so on.

Intermediate COCOMOs

Intermediate COCOMO computes software development effort as function of program size and a set of "cost drivers" that include subjective assessment of product, hardware, personnel and project attributes. This extension considers a set of four "cost drivers", each with a number of subsidiary attributes:-

- Product attributes
 - Required software reliability
 - Size of application database
 - Complexity of the product
- Hardware attributes
 - Run-time performance constraints
 - Memory constraints
 - Volatility of the virtual machine environment
 - Required turnabout time
- Personnel attributes
 - Software engineering capability
 - Applications experience
 - Virtual machine experience
 - Programming language experience
- Project attributes
 - Use of software tools
 - Application of software engineering methods
 - Required development schedule

Each of the 15 attributes receives a rating on a six-point scale that ranges from "very low" to "extra high" (in importance or value). An effort multiplier from the table below applies to the rating. The product of all effort multipliers results in an *effort adjustment factor (EAF)*. Typical values for EAF range from 0.9 to 1.4.

| Cost Drivers | Ratings | | | | | |
|----------------------------------|----------|------|---------|------|-----------|------------|
| | Very Low | Low | Nominal | High | Very High | Extra High |
| Product attributes | | | | | | |
| Required software reliability | 0.75 | 0.88 | 1.00 | 1.15 | 1.40 | |
| Size of application database | | 0.94 | 1.00 | 1.08 | 1.16 | |
| Complexity of the product | 0.70 | 0.85 | 1.00 | 1.15 | 1.30 | 1.65 |
| Hardware attributes | | | | | | |
| Run-time performance constraints | | | 1.00 | 1.11 | 1.30 | 1.66 |
| Memory constraints | | | 1.00 | 1.06 | 1.21 | 1.56 |

| | | | | |
|---|------|------|------|-----------|
| Volatility of the virtual machine environment | 0.87 | 1.00 | 1.15 | 1.30 |
| Required turnabout time | 0.87 | 1.00 | 1.07 | 1.15 |
| Personnel attributes | | | | |
| Analyst capability | 1.46 | 1.19 | 1.00 | 0.86 0.71 |
| Applications experience | 1.29 | 1.13 | 1.00 | 0.91 0.82 |
| Software engineer capability | 1.42 | 1.17 | 1.00 | 0.86 0.70 |
| Virtual machine experience | 1.21 | 1.10 | 1.00 | 0.90 |
| Programming language experience | 1.14 | 1.07 | 1.00 | 0.95 |
| Project attributes | | | | |
| Application of software engineering methods | 1.24 | 1.10 | 1.00 | 0.91 0.82 |
| Use of software tools | 1.24 | 1.10 | 1.00 | 0.91 0.83 |
| Required development schedule | 1.23 | 1.08 | 1.00 | 1.04 1.10 |

Table 3.8

The Intermediate Cocomo formula now takes the form:

$$E = a_i (KLoC)^{b_i} \cdot EAF$$

where E is the effort applied in person-months, **KLoC** is the estimated number of thousands of delivered lines of code for the project, and **EAF** is the factor calculated above. The coefficient **a_i** and the exponent **b_i** are given in the next table.

| Software project | a _i | b _i |
|------------------|----------------|----------------|
| Organic | 3.2 | 1.05 |
| Semi-detached | 3.0 | 1.12 |
| Embedded | 2.8 | 1.20 |

Table 3.9

Cost Effort Calculation For The Project

According to the table 3.7

The EAF of the project is calculated as

$$EAF = 1.1238189767$$

Now,

Using the formula

$$E = a_i (KLoC)^{b_i} \cdot EAF$$

And

$$a_i = 3.0$$

$$b_i = 1.12$$

$$\& KLoc = 111.6K$$

Therefore the Cost Effort Estimation

$$\begin{aligned} E &= 3.0(111.6)^{1.12} \cdot 1.124 \\ &= 3 (198.482854601) \cdot 1.124 \\ &= 669.2847 \end{aligned}$$

3.6 CLASS DIAGRAM

A **class diagram** in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes.

The class diagram is the main building block of object oriented modelling. It is used both for general conceptual modelling of the systematics of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed.

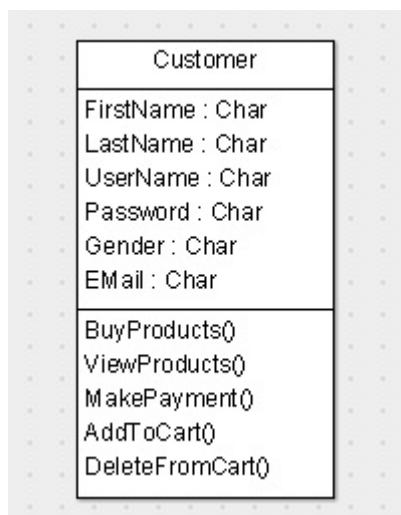


Figure: 3.7

A Customer class with three sections.

In the diagram, classes are represented with boxes which contain three parts-

- The upper part holds the name of the class
- The middle part contains the attributes of the class
- The bottom part gives the methods or operations the class can take or undertake

In the design of a system, a number of classes are identified and grouped together in a class diagram which helps to determine the static relations between those objects. With detailed modeling, the classes of the conceptual design are often split into a number of subclasses.

3.6.1 CLASS DIAGRAM FOR JUST SHOP SYSTEM

Classes: Admin, Products, Guest, Customer, Cart, Payment.

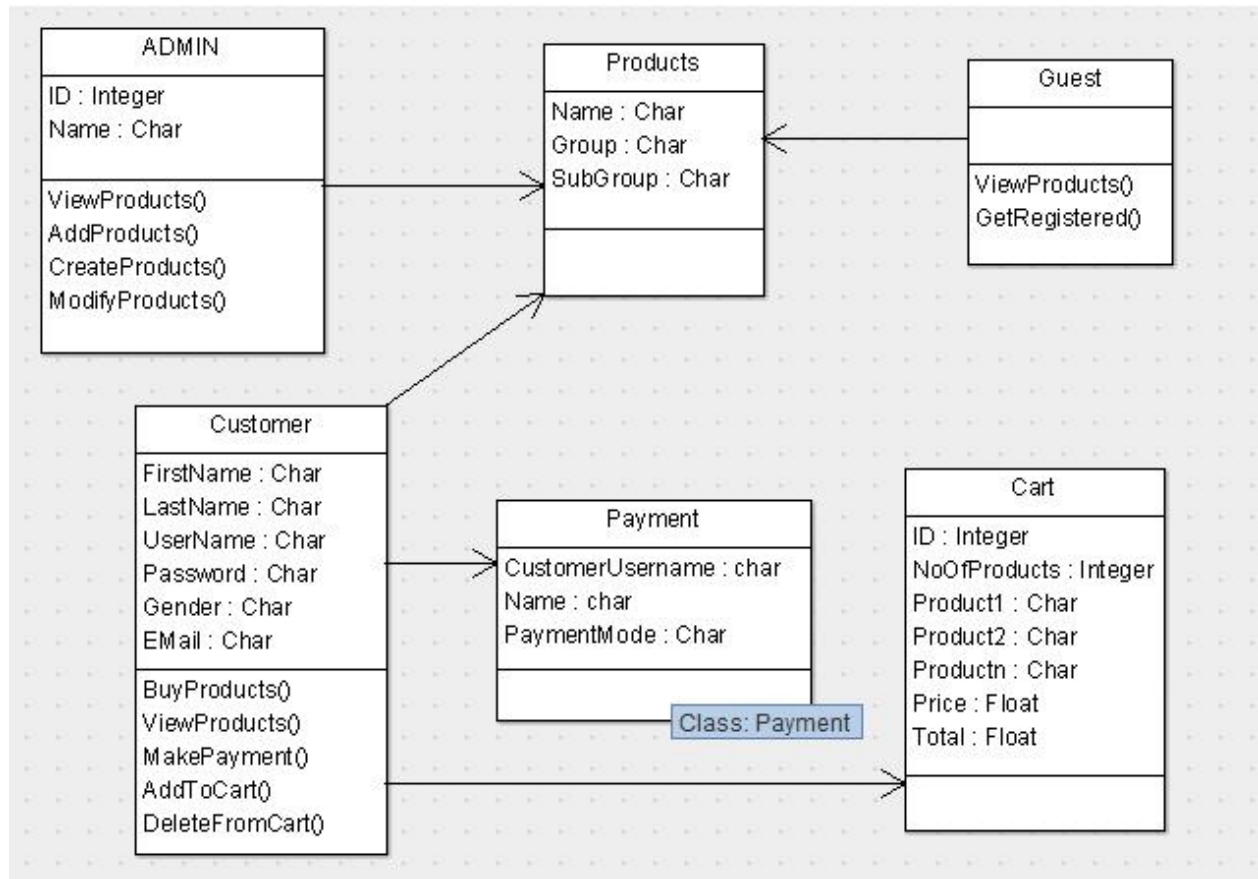


Figure: 3.8

Class diagram pros:

- discovering related data and attributes
- getting a quick picture of the important entities in a system
- seeing whether you have too few/many classes
- seeing whether the relationships between objects are too complex, too many in number, simple enough, etc.
- spotting dependencies between one class/object and another

Class diagram cons:

- discovering algorithmic (not data-driven) behavior
- finding the flow of steps for objects to solve a given problem
- understanding the app's overall control flow.

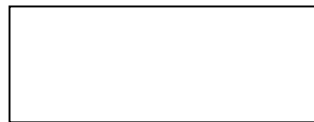
3.7 ENTITY RELATIONSHIP DIAGRAMS (ER-DIAGRAMS)

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes

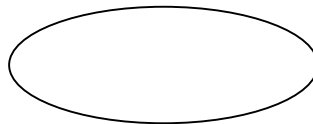
An **entity-relationship model** (ERM) in software engineering is an abstract and conceptual representation of data. Entity-relationship modeling is a relational schema database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion.

Symbols used in this E-R Diagram

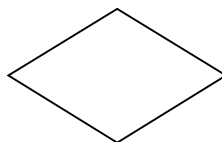
Entity: Entity is a “thing” in the real world with an independent existence. An entity may be an object with a physical existence such as person, car or employee. Entity symbol is as follows



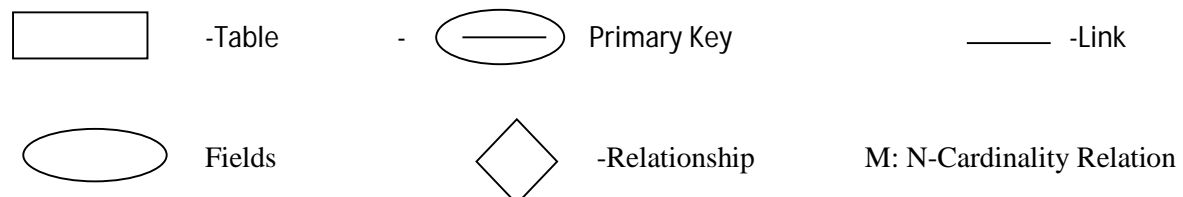
Attribute: Attribute is a particular property that describes the entity. Attribute symbol is



Relationship: Relationship will be several implicit relationships among various entity types whenever an attribute of one entity refers to another entity type some relationship exists. Relationship symbol is:



Key attributes: An entity type usually has an attribute whose values are distinct for each individual entity in the collection. Such an attribute is called key attribute. Key attribute symbol is as follows



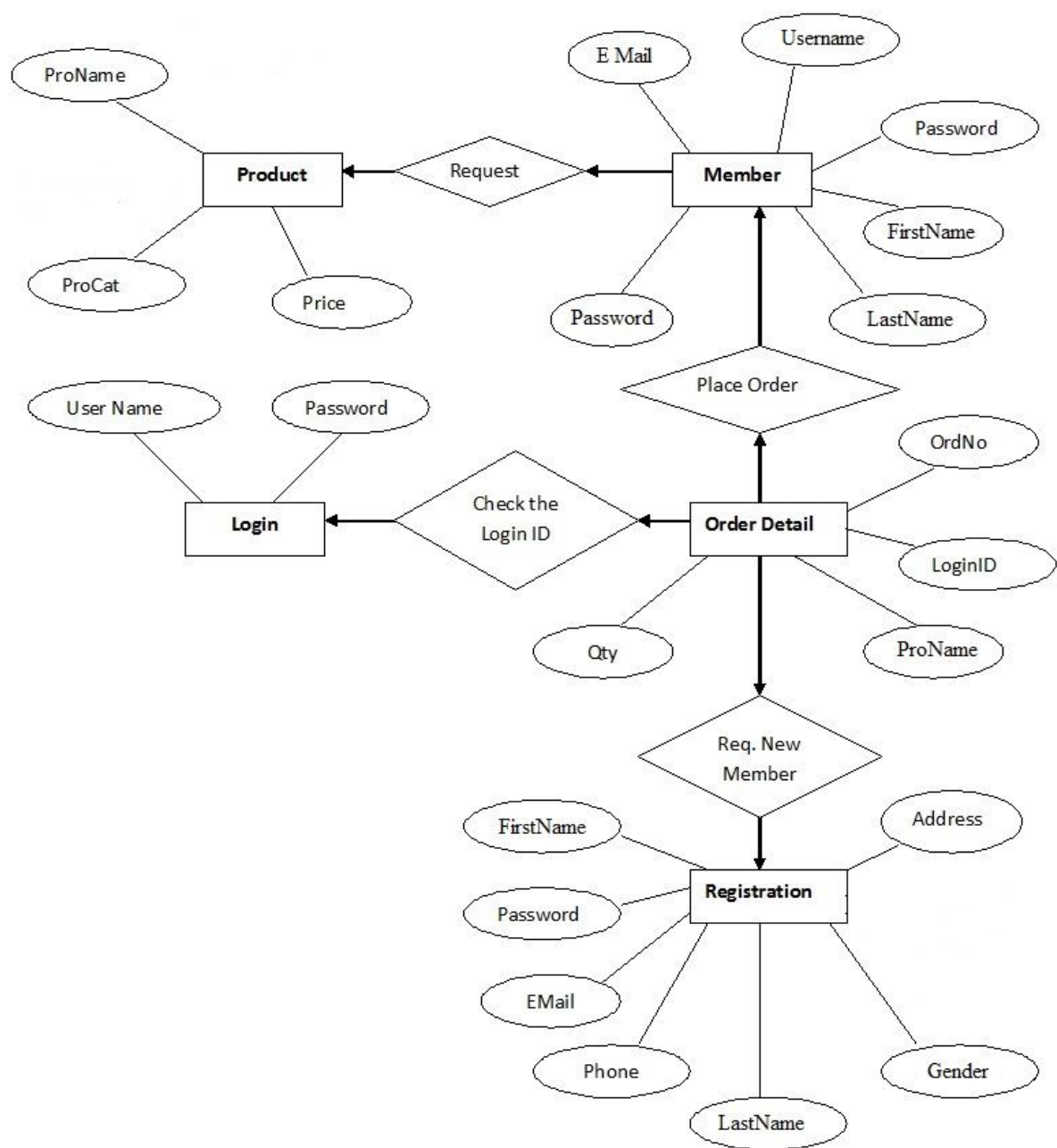


Figure: 3.9

3.8 DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kinds of data will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).

Data Flow Diagram serves two purposes:

- i. To provide annunciation of how data are transformed as they move through the system.
- ii. To depict the functions that transforms the data flow.

DFDs are an excellent mechanism for communicating with the customer during requirement analysis and are widely used for the representation of external and top-level internal design specification. In the latter situations, DFDs are quite valuable for subsystem, files and data links. The DFD methodology is quite effective, especially when the required design is unclear. In the process, many levels of DFDs are created depending upon the level of details needed

The Level 0 DFD is also called Context Level DFD. It depicts the overview of the entire system. The major external entities, a single process and the output stores constitute the level-0 DFD. Though this diagram does not depict the system in detail, it represents the overall inputs, process and output of the entire system at a very high level.

The Level 0 DFD is now expended into a level 1 model. It should be noted that information flow continuity is maintained between level 0 and level 1. The process represents at DFD level 1 further refined into lower levels. This further refinement is continued until an easily implement able program component is reached.

3.8.1 SYMBOLS USED IN DFD

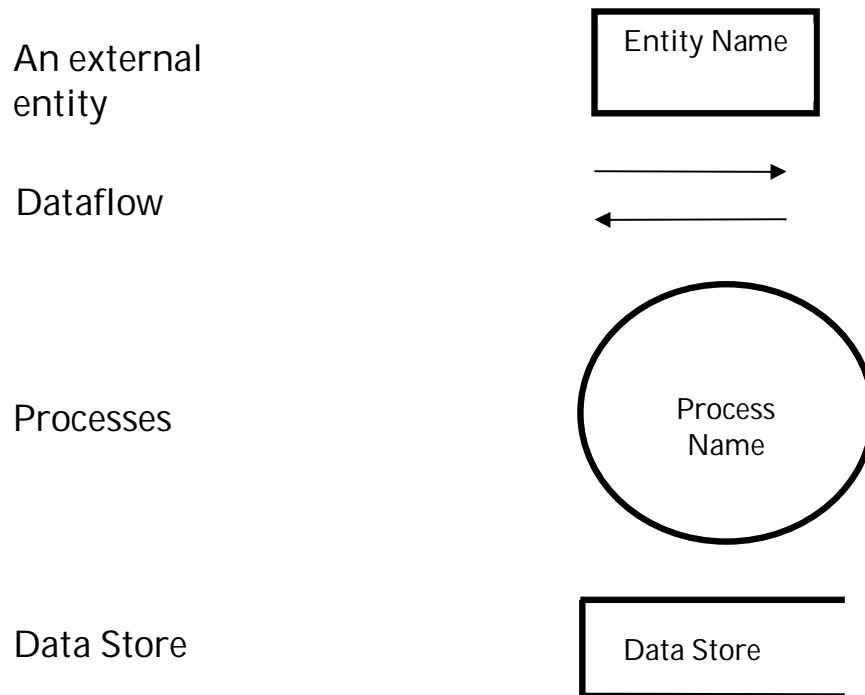


Figure: 3.10

3.8.2 CONTEXT LEVEL DIAGRAM

This level shows the overall context of the system and its operating environment and shows the whole system as just one process. It does not usually show data stores, it only shows that the data is stored and access from the database.

3.9 CONTEXT LEVEL DFD

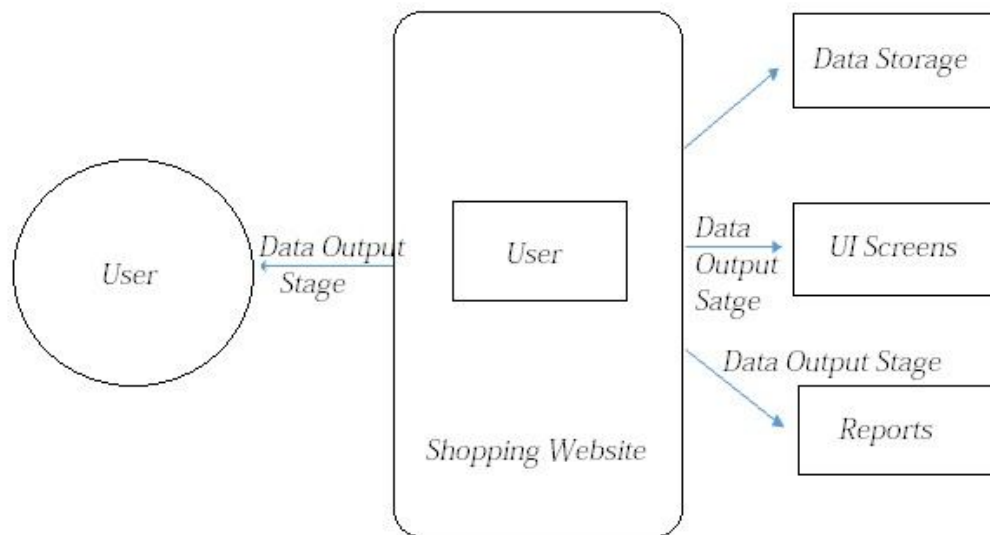


Figure: 3.11

3.10 LEVEL 1 DFD ADMINISTRATOR

3.10.1 New User Registration DFD

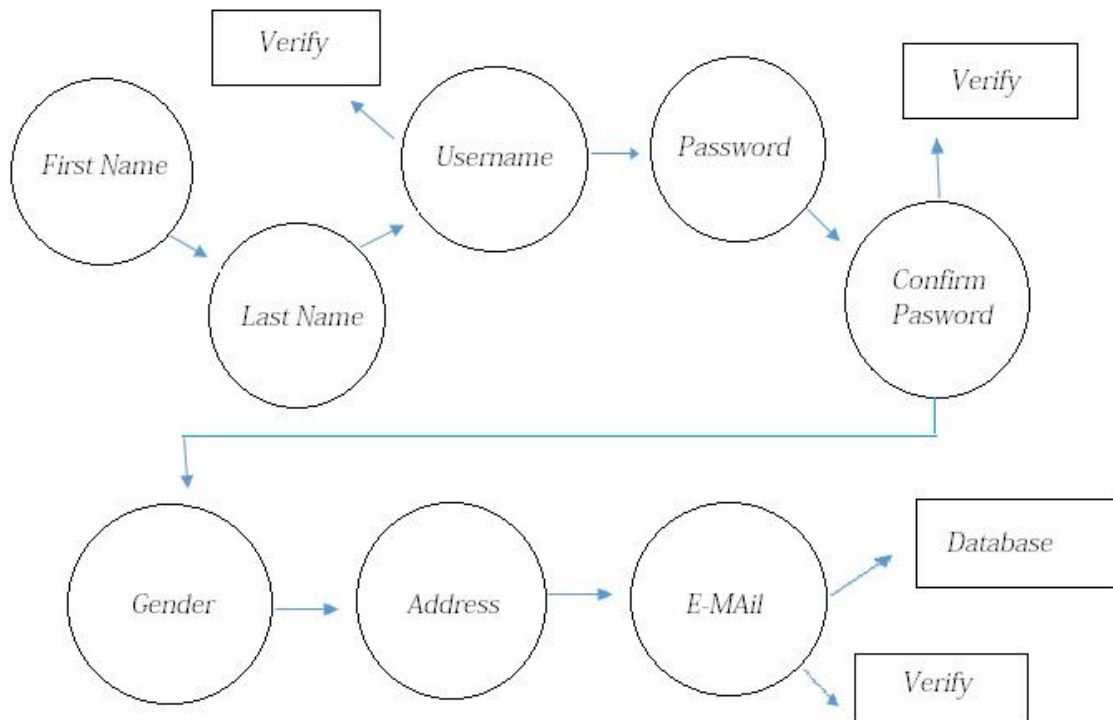


Figure: 3.12

3.10.2 Login DFD

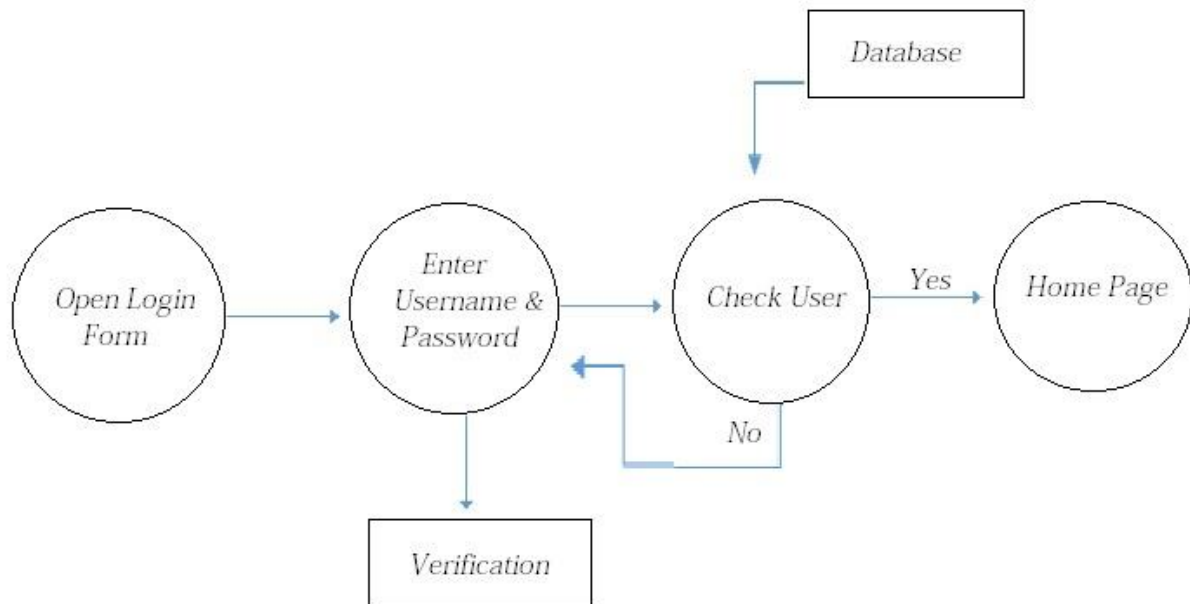


Figure: 3.13

3.10.3 User Activity DFD

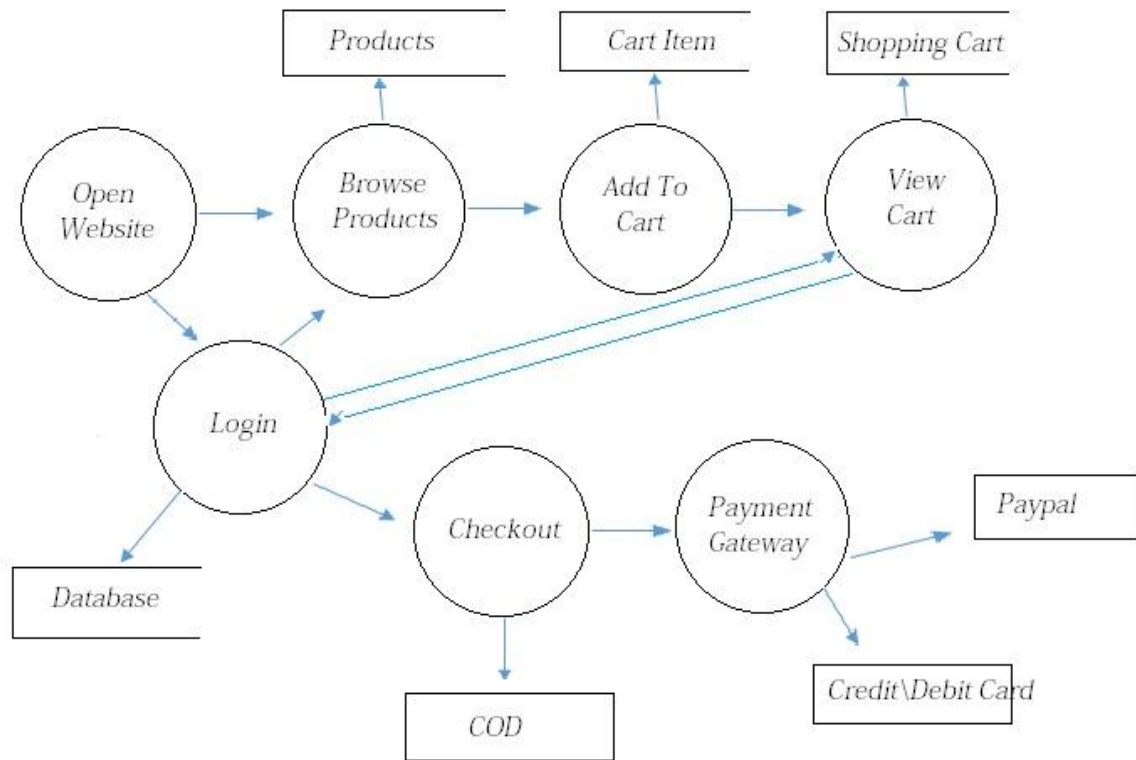


Figure: 3.14

3.11 DATABASE DESIGN

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. Keeping and storing data is the purpose of any database and the data must be protected.

The database design is a two level process. In the first step

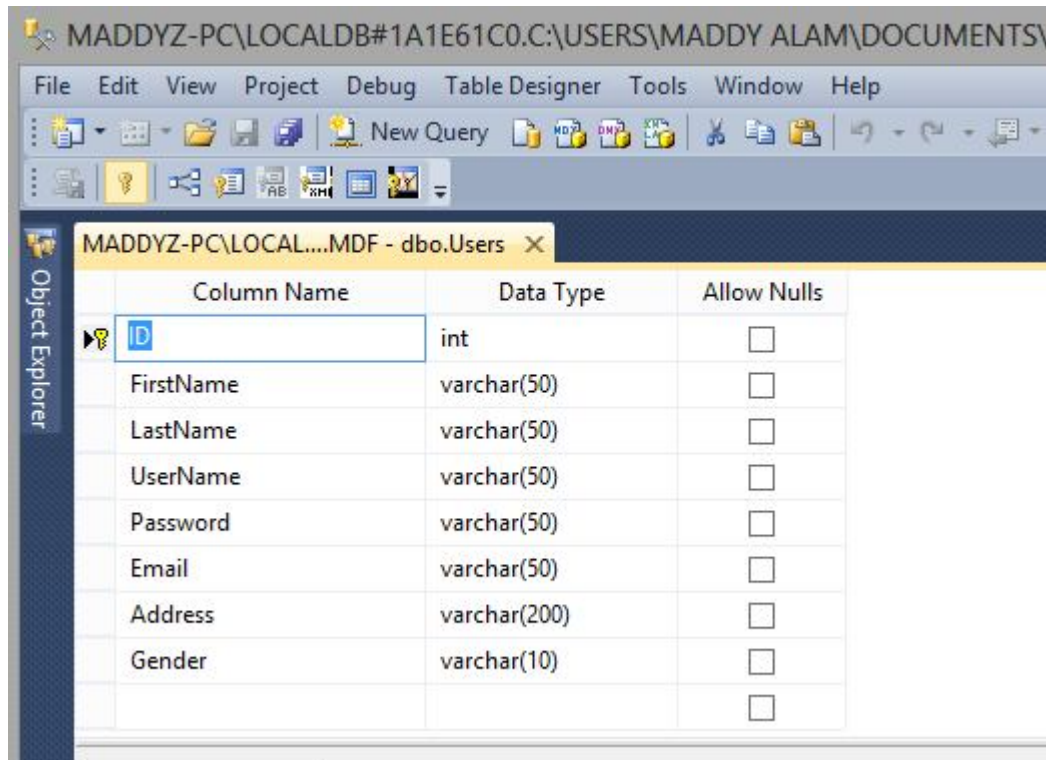
- i. User requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual Database Management System (DBMS).
- ii. In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used.

A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- i. Data Integrity
- ii. Data independence

3.12 DATABASE TABLE DESIGN

3.12.1 User Registration

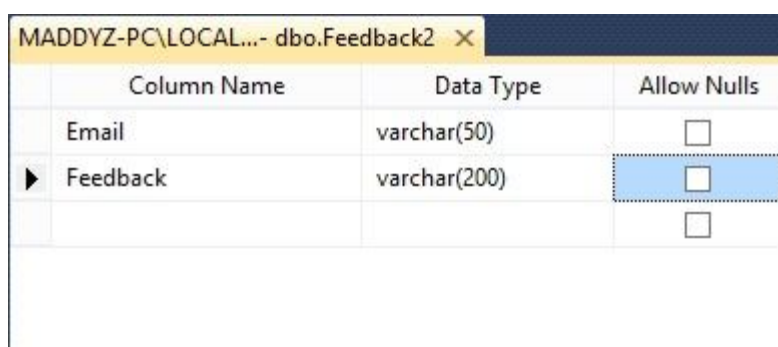


The screenshot shows the SQL Server Enterprise Designer interface. The title bar indicates the file path: MADDYZ-PC\LOCALDB#1A1E61C0.C:\USERS\MADDY ALAM\DOCUMENTS\... The menu bar includes File, Edit, View, Project, Debug, Table Designer, Tools, Window, and Help. The toolbar contains icons for New Query, Save, Undo, Redo, and other standard database operations. The Object Explorer on the left shows the 'MADDYZ-PC\LOCAL...MDF - dbo.Users' table. The main area displays the table design for 'dbo.Users' with the following columns:

| Column Name | Data Type | Allow Nulls |
|-------------|--------------|--------------------------|
| ID | int | <input type="checkbox"/> |
| FirstName | varchar(50) | <input type="checkbox"/> |
| LastName | varchar(50) | <input type="checkbox"/> |
| UserName | varchar(50) | <input type="checkbox"/> |
| Password | varchar(50) | <input type="checkbox"/> |
| Email | varchar(50) | <input type="checkbox"/> |
| Address | varchar(200) | <input type="checkbox"/> |
| Gender | varchar(10) | <input type="checkbox"/> |
| | | <input type="checkbox"/> |

Table 3.10

3.12.2 Feedback Database

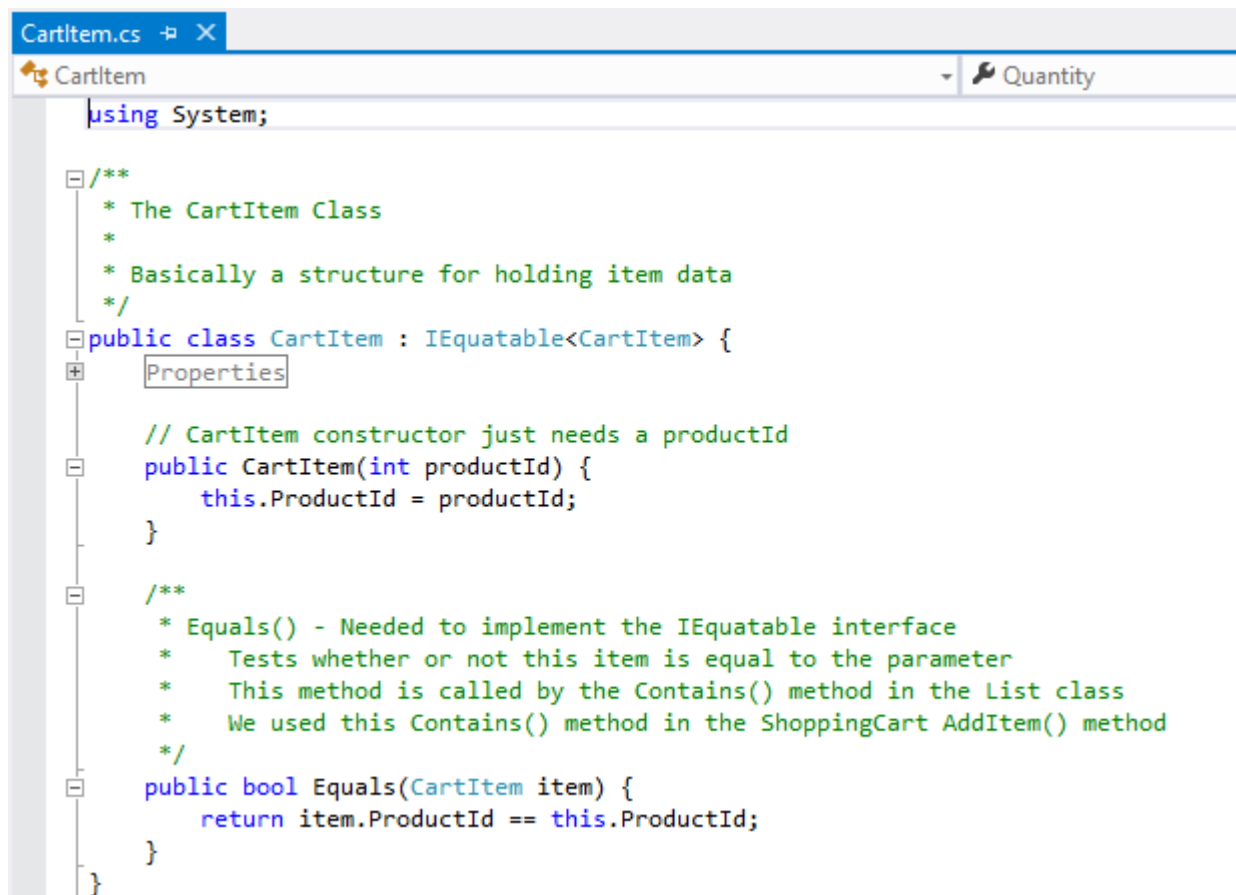


The screenshot shows the SQL Server Enterprise Designer interface for the 'dbo.Feedback2' table. The title bar indicates the file path: MADDYZ-PC\LOCAL... - dbo.Feedback2. The menu bar and toolbar are the same as in the previous screenshot. The Object Explorer on the left shows the 'MADDYZ-PC\LOCAL... - dbo.Feedback2' table. The main area displays the table design for 'dbo.Feedback2' with the following columns:

| Column Name | Data Type | Allow Nulls |
|-------------|--------------|--------------------------|
| Email | varchar(50) | <input type="checkbox"/> |
| Feedback | varchar(200) | <input type="checkbox"/> |
| | | <input type="checkbox"/> |

Table 3.11

3.12.3 Cartitem.cs



```
using System;

/**
 * The CartItem Class
 *
 * Basically a structure for holding item data
 */
public class CartItem : IEquatable<CartItem> {
    Properties

    // CartItem constructor just needs a productId
    public CartItem(int productId) {
        this.ProductId = productId;
    }

    /**
     * Equals() - Needed to implement the IEquatable interface
     * Tests whether or not this item is equal to the parameter
     * This method is called by the Contains() method in the List class
     * We used this Contains() method in the ShoppingCart AddItem() method
     */
    public bool Equals(CartItem item) {
        return item.ProductId == this.ProductId;
    }
}
```

Table 3.12

3.12.4 Product.cs

The image shows a screenshot of a Visual Studio code editor window. The title bar at the top says 'Product.cs'. Below the title bar, there's a dropdown menu showing 'Product' and 'Product(int id)'. The main area of the editor contains the following C# code:

```
/**
 * The Product class
 */
public class Product
{
    public int Id { get; set; }
    public decimal Price { get; set; }
    public string Description { get; set; }

    public Product(int id)
    {
        this.Id = id;
        switch (id) {
            case 1:
                this.Price = 5;
                this.Description = "Electrical Engineering";
                break;
            case 2:
                this.Price = 6;
                this.Description = "Electronic & Comm Engg.";
                break;
            case 3:
                this.Price = 7;
                this.Description = "Mechanics";
                break;
        }
    }
}
```

Table 3.13

3.12.5Shoppingcart.cs

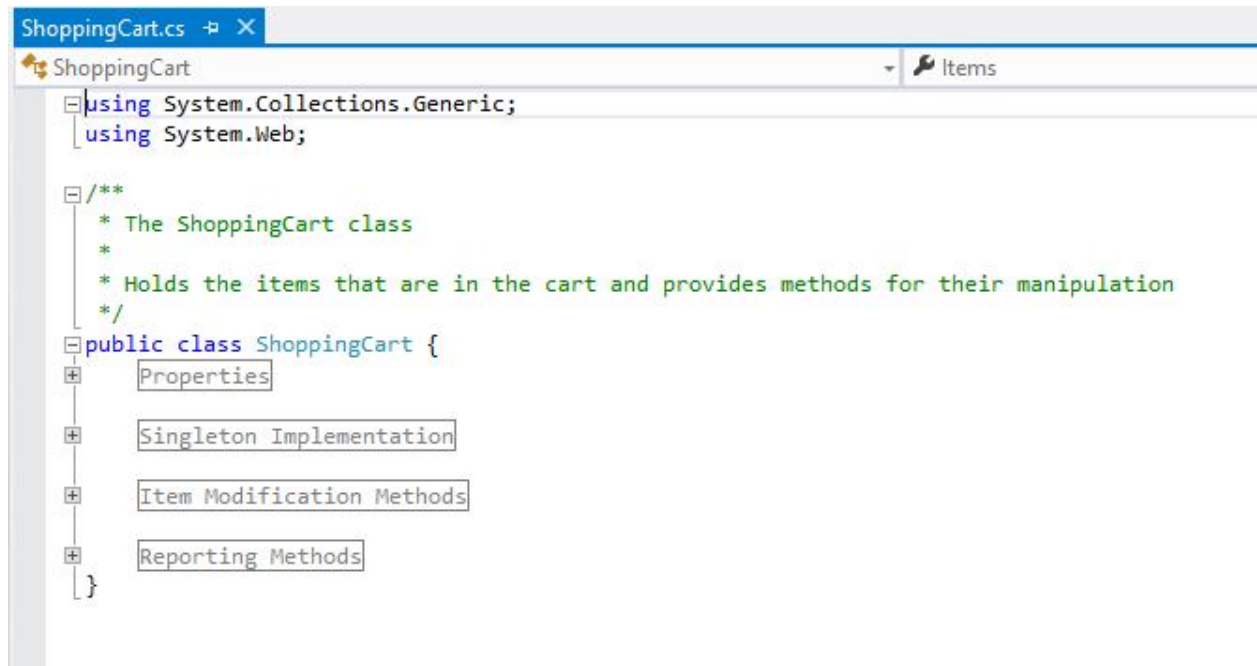


Table 3.14

3.13 SOFTWARE DESCRIPTION

OVERVIEW OF ASP.Net AND SQL SERVER

i. The Control Properties

Before writing an event procedure for the control to response to a user's input, you have to set certain properties for the control to determine its appearance and how it will work with the event procedure. You can set the properties of the controls in the properties window or at runtime.

ii. The Text Box

The text box is the standard control for accepting input from the user as well as to display the output. It can handle string (text) and numeric data but not images or pictures. String in a text box can be converted to a numeric data by using the function `Val(text)`.

iii. The Label

The label is a very useful control for Visual Basic, as it is not only used to provide instructions and guides to the users, it can also be used to display outputs. One of its most important properties is Caption. Using the syntax `label. Caption`, it can display text and numeric data. You can change its caption in the properties window and also at runtime.

iv. The Command Button

The command button is one of the most important controls as it is used to execute commands. It displays an illusion that the button is pressed when the user click on it. The most common event associated with the command button is the Click event.

v. The Picture Box

The Picture Box is one of the controls that is used to handle graphics. You can load a picture at design phase by clicking on the picture item in the properties window and select the picture from the selected folder. You can also load the picture at runtime using the `LoadPicture` method.

vi. The Image Box

The Image Box is another control that handles images and pictures. It functions almost identically to the picture box. However, there is one major difference, the image in an Image Box is stretchable, which means it can be resized. This feature is not available in the Picture Box. Similar to the Picture Box, it can also use the `LoadPicture` method to load the picture.

vii. The List Box

The function of the List Box is to present a list of items where the user can click and select the items from the list. In order to add items to the list, we can use the `AddItem` method.

viii. The Check Box

The Check Box control lets the user select or unselect an option. When the Check Box is checked, its value is set to 1 and when it is unchecked, the value is set to 0. You can include the statements `Check1.Value=1` to mark the Check Box and `Check1.Value=0` to unmark the Check Box, as well as use them to initiate certain actions.

ix. The Option Box

The Option Box control also lets the user select one of the choices. However, two or more Option Boxes must work together because as one of the Option Boxes is selected, the other Option Boxes will be unselected. In fact, only one Option Box can be selected at one time. When an option box is selected, its value is set to “True” and when it is unselected; its value is set to “False”. In the following example, the shape control is placed in the form together with six Option Boxes. When the user clicks on different option boxes, different shapes will appear. The values of the shape control are 0, 1, and 2,3,4,5 which will make it appear as a rectangle, a square, an oval shape, a rounded rectangle and a rounded square respectively.

x. The Directory List Box

The Directory List Box is for displaying the list of directories or folders in a selected drive. When you place this control into the form and run the program, you will be able to select different directories from a selected drive in your computer.

3.14 DATABASE FILE

This is the main file that encompasses the entire database and that is saved to the hard-drive or floppy disk.

Example: StudentDatabase.mdb

- i. Table: A table is a collection of data about a specific topic. There can be multiple tables in a database.
- ii. Field: Fields are the different categories within a Table. Tables usually contain multiple fields.
- iii. Datatypes: Datatypes are the properties of each field. A field only has 1 datatype.

SYSTEM TESTING AND IMPLEMEN- TATIONS

4.1 TESTING

Testing is a process of executing a program with the interest of finding an error. A good test is one that has high probability of finding the yet undiscovered error. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements. Testing should systematically uncover different classes of errors in a minimum amount of time with a minimum amount of efforts. Two classes of inputs are provided to test the process:

- i. A software configuration that includes a software requirement specification, a design specification and source code.
- ii. A software configuration that includes a test plan and procedure, any testing tool and test cases and their expected results.

4.2 TYPES OF TESTING

Testing is divided into several distinct operations:

4.2.1 Unit Testing

Unit test comprises of a set tests performed by an individual program prior to the integration of the unit into large system. A program unit is usually the smallest free functioning part of the whole system. Module unit testing should be as exhaustive as possible to ensure that each representation handled by each module has been tested. All the units that makeup the system must be tested independently to ensure that they work as required.

4.2.2 Integration Testing

Integration testing is a system technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested modules and build a program structure that has been dictated by design. Bottom-up integration is the traditional strategy used to integrate the components of a software system into functioning whole. Bottom-up integration consists of unit test followed by testing of the entire system. A sub-system consists of several modules that communicated with other defined interface.

4.2.3 Validation Testing

After validation testing, software is completely assembled as a package, interfacing errors that have been uncovered and corrected and the final series of software test; the validation test begins. Steps taken during software design and testing can greatly improve the probability of successful integration in the larger system. System testing is actually a series of different tests whose primary purpose is to fully exercise the compute –based system.

4.2.4 Recovery Testing

It is a system that forces the software to fail in a variety of ways and verifies that the recovery is properly performed.

4.2.5 Security Testing

It attempts to verify that protection mechanisms built into a system will in fact protect it from improper penetration. The system's security must of course be tested from in vulnerability form frontal attack.

4.2.6 Stress Testing

Stress tools are designed to confront programs with abnormal situations. Stress testing executes a system in a manner that demands resources in abnormal quantity and volume.

4.2.7 Black Box Testing

Black box testing is done to find out the following information as shown in below:

- Incorrect or missing functions.
- Interface errors.
- Errors or database access.
- Performance error.
- Termination error.

The mentioned testing is carried out successfully for this application according to the user's requirement specification.

4.2.8 Test Data Output

After preparing test data, the system under study is tested using the test data. While testing the system using test data, errors are again uncovered and corrected by using above testing and corrections are also noted for future use.

4.3 SCREEN SHOTS

HOME PAGE

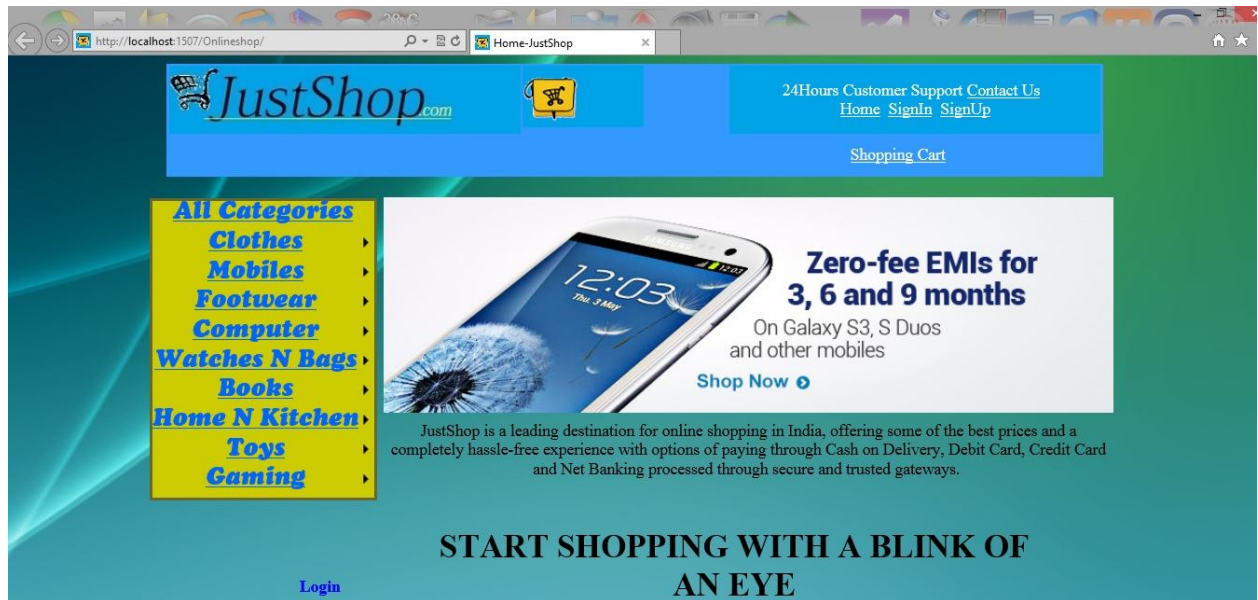


Figure: 4.1

CART EMPTY

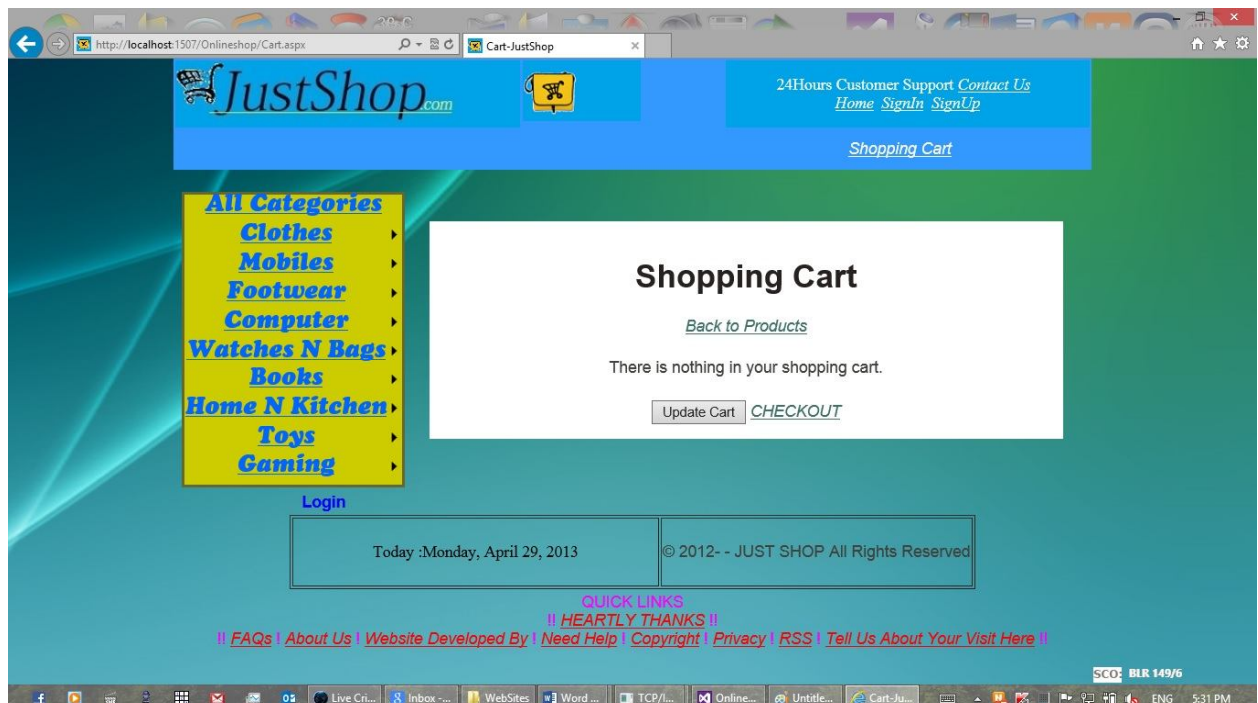


Figure: 4.2

CLOTHES-SUB MENU

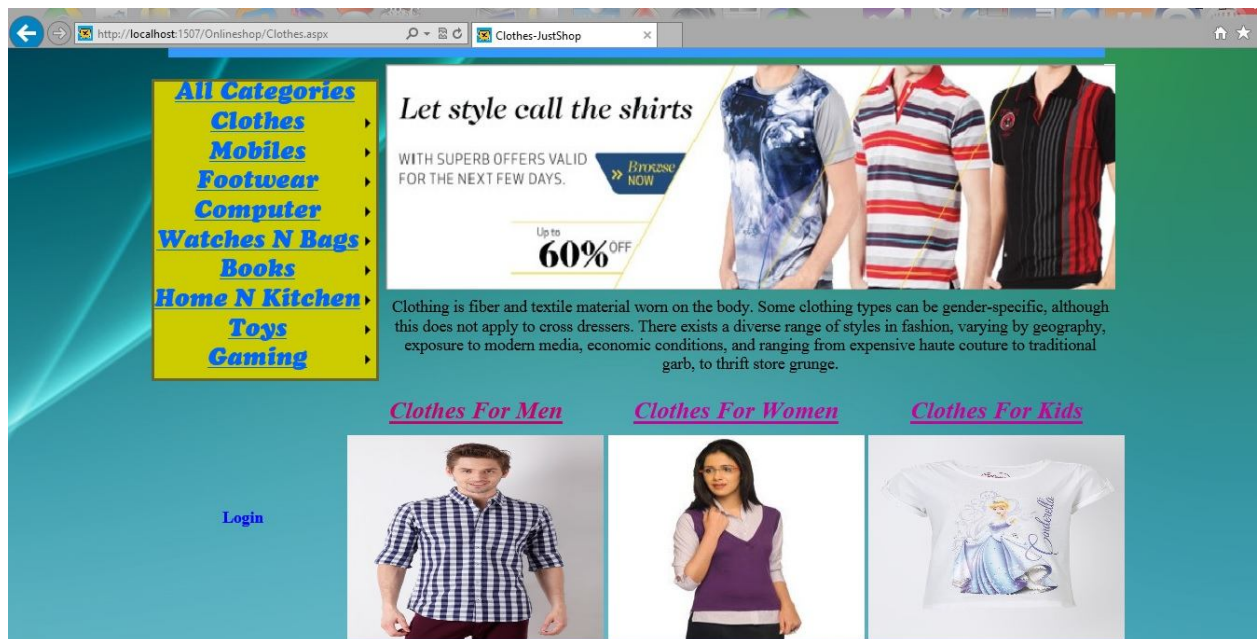


Figure: 4.3

GAMES-SUB MENU

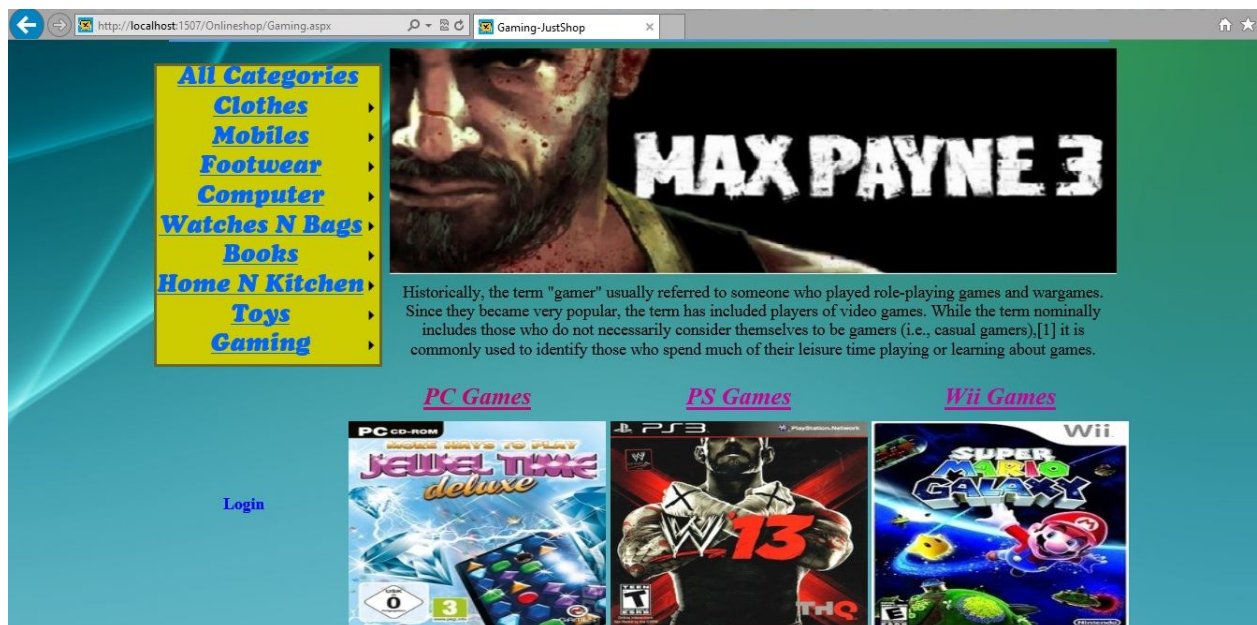


Figure: 4.4

ITEMS-SUB SUB MENU

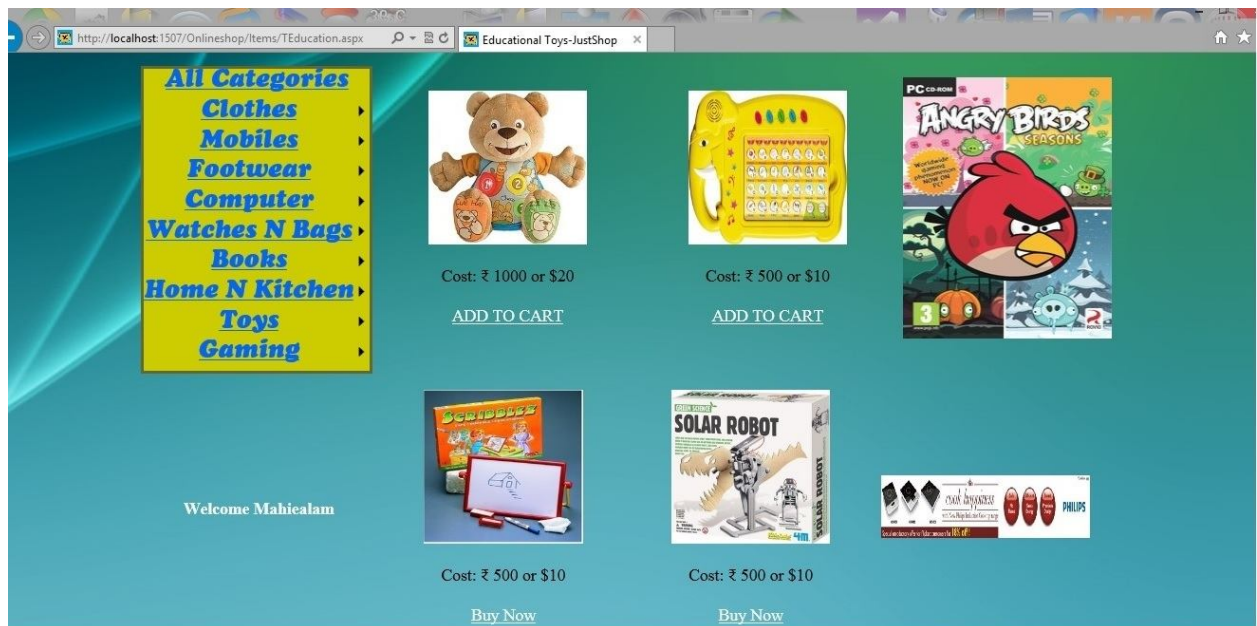


Figure: 4.5

REGISTER



Figure: 4.6

LOGIN

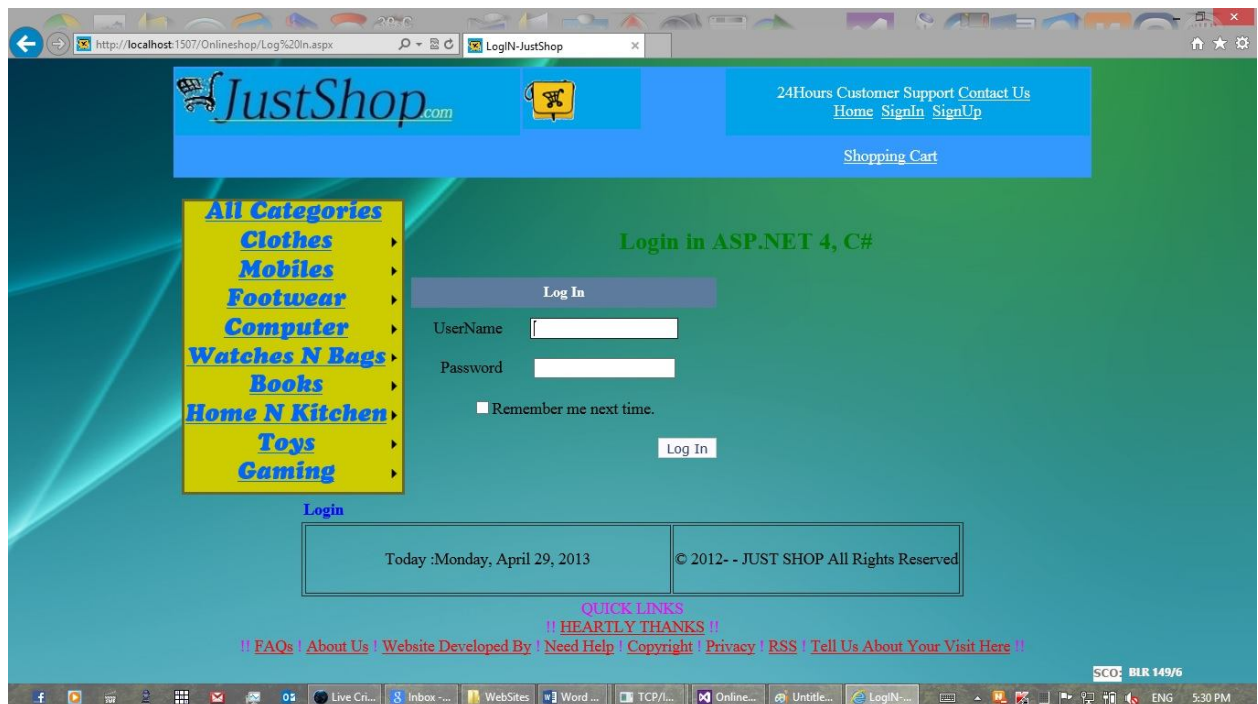


Figure: 4.7

CART FULL

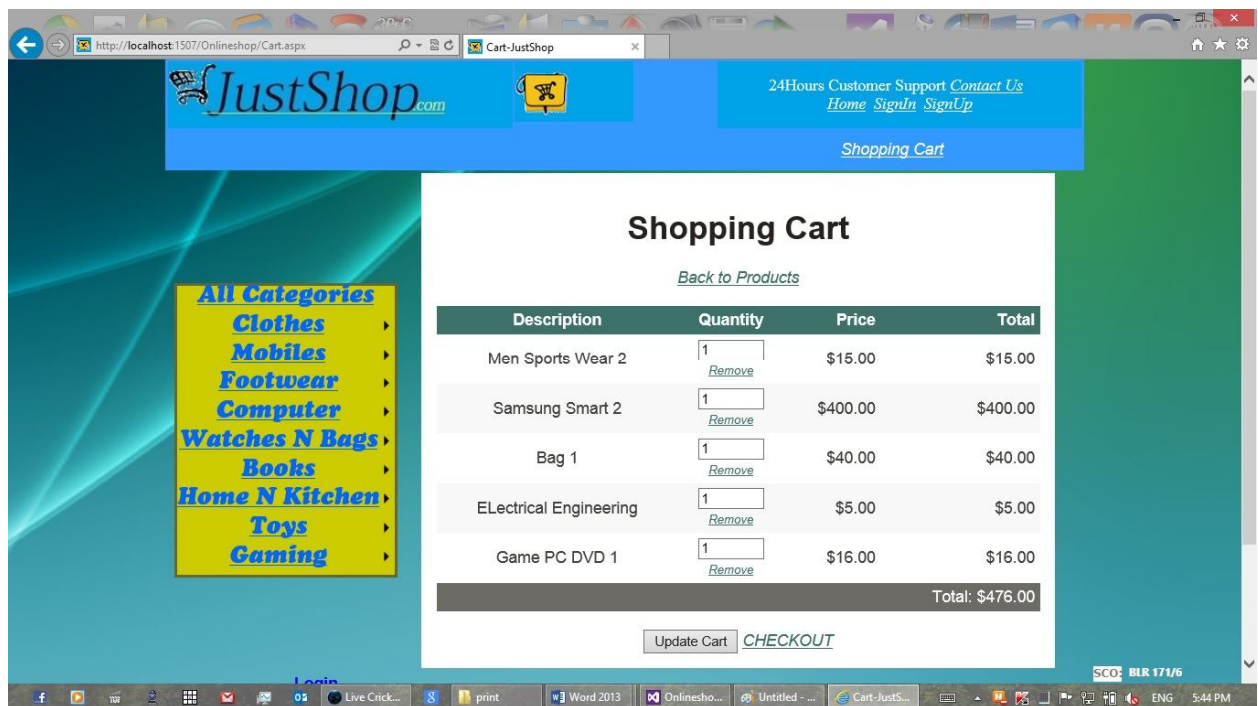


Figure: 4.8

CONTACT US



Figure: 4.9

5. CONCLUSION & FUTURE ENHANCEMENT

The internet has become a major resource in modern business thus electronic shopping or Online Shopping has gained significance not only from the entrepreneur's but also from the customer's point of view. For the entrepreneur, electronic shopping generates new business opportunities and for the customers, it makes comparative shopping possible. As per a survey, most consumers of online stores are impulsive and usually make a decision to stay on a site within a few seconds. The design of the project to provide the user with easy navigation, retrieval of data and necessary feedback as much as possible. A good shopping cart design must be accompanied with user-friendly shopping cart application logic. It should be convenient for the customer to view the contents of their cart and to be able to remove and add items to their cart. The shopping cart application described in this project provides a number of features that are designed to make the customer more comfortable.

The project Just Shop is completed, satisfying the required design specifications. The system provides a user-friendly interface. The software is developed with modular approach. All modules in the system have been tested with valid data and invalid data and everything work successfully. Thus the system has fulfilled all the objectives identified and is able to replace the existing system. The constraints are met and overcome successfully. The system is designed as like it was decided in the design phase.

The application has been tested with live data and has provided a successful result. Hence the software has proved to work efficiently.

The project helps in understanding the creation of an interactive web page and the technologies used to implement it. The building of the project has given me the precise knowledge about how VISUAL STUDIO is used to develop a website, how it connects to the database to access the data and how the data and web pages are modified to provide the user with a shopping cart application.

The basic future enhancement to be deployed later can be

- i. The shopping online will require more secure encryption to provide more secure transactions
- ii. Customers should be given reward points for frequent purchases
- iii. Live support should be enabled
- iv. More products to be added
- v. Product details to be added
- vi. Inclusion Of SMS system
- vii. Email Check

REFERENCES

1. Charles Hampf (2000) 'Visual Basic' University of Toronto
2. Herbert Schildt (2000) 'Visual Basic 6.0' Tata McGraw Hill
3. John Zukowski (2000) 'Visual Basic 6.0' 'BPB Publications
4. Jamie Jaworsky 'Visual Basic 6.0' Techmedia
5. Alam, Mehtab. (2011). Online Banking. ISBN: 978-620-3-86302-4
6. Stefan Denninger 'Visual Basic 6.0' Author's Press
7. Ian Somerville 'Software engineering'
8. Alam, M. (2011). Online Banking (1st ed.). New Delhi: LAP LAMBERT Academic Publishing, ISBN: 978-620-3-86302-4. DOI: 10.6084/m9.figshare.14612127.
9. Microsoft Press eBook Programming Windows 8 Apps with HTML CSS and JavaScript
10. SQL Server 2012 Product Guide
11. Alam, M. (2012). Electronic Ticket Machine (1st ed.). New Delhi. LAP LAMBERT Academic Publishing, ISBN: 978-620-3-86332-1, DOI: 10.6084/m9.figshare.14661354.
12. Elmasri Navathe 'Fundamentals of database systems'
13. www.wikipedia.com
14. www.stackoverflow.com
15. www.w3schools.com
16. <http://msdn.microsoft.com/en-US/>
17. <http://msdn.microsoft.com/en-IN/library/hh234692.aspx>

**More
Books!**

yes
I want morebooks!

Buy your books fast and straightforward online - at one of world's fastest growing online book stores! Environmentally sound due to Print-on-Demand technologies.

Buy your books online at
www.morebooks.shop

Kaufen Sie Ihre Bücher schnell und unkompliziert online – auf einer der am schnellsten wachsenden Buchhandelsplattformen weltweit! Dank Print-On-Demand umwelt- und ressourcenschonend produziert.

Bücher schneller online kaufen
www.morebooks.shop

KS OmniScriptum Publishing
Brīvības gatve 197
LV-1039 Rīga, Latvia
Telefax: +371 686 204 55

info@omniscryptum.com
www.omniscryptum.com

OMNIScriptum



