Project Report On

Shopping Cart

Submitted in the partial fulfillment of the requirement for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering

By:

Bibrat Ranjan Pradhan (1201105065)

Department of Computer Science, Engineering & Applications

Indira Gandhi Institute of Technology, Sarang

Dhenkanal, Odisha-759146

MAY-2016

Project Report On

Shopping Cart

Submitted in the partial fulfillment of the requirement for the award of degree of

Bachelor of Technology
in
Computer Science & Engineering

By:

Bibrat Ranjan Pradhan (1201105065)

Guided By:

Prof. Sarojananda Mishra

Prof. Sucharita Natha Sharma

Department of Computer Science, Engineering & Applications

Indira Gandhi Institute of Technology, Sarang

Dhenkanal, Odisha-759146

MAY-2016

Certificate

This is to certify that this project entitled ***Shopping Cart*** *submitted by* ***Bibrat Ranjan Pradhan*** *(31806), of Computer Science Engineering & Applications Department, Indira Gandhi Institute of Technology, Sarang in the partial fulfillment of the requirement for the award of Bachelor of Technology (Computer Science & Engineering) Degree of BPUT, Odisha, is a record of student's own study carried under my supervision & guidance.*

This report has not been submitted to any other university or institution for the award of any degree.

Date:

(Prof. Sarojananda Mishra)
Project Supervisor, Head of
Dept. of CSEA, IGIT, Sarang

(Prof. Sarojananda Mishra)
Dept. of CSEA, IGIT, Sarang
Dhenkanal, Odisha

(Prof Sucharita Natha Sharma)
Project Supervisor, Prof.
Dept. of CSEA, IGIT, Sarang

Signature of External Examiner

Acknowledgement

On the submission of our project, **Shopping Cart**, I feel it is our foremost duty to acknowledge the help and assistance rendered by various persons and our institute "Indira Gandhi Institute of Technology".

Iam also grateful to Prof. S.N Mishra, HOD, Department of Computer Science & Engineering, and Prof. Sucharita Natha Sharma for their active involvement in the entire process. I would like to thank all respected teachers for helping us in successfully completing this project. I also take the opportunity to thank all our family members and friends for their consistent support.

BIBRAT RANJAN PRADHAN

Abstract

OBJECTIVE: Create an active and closed system for selling of books in I.G.I.T from previous semesters which the seller no longer requires in current or upcoming semesters. This is for people who find it difficult to carry all the books accumulated at the end 4 years back home or for those who really find it “disheartening” to lend their books to juniors for free only to avoid the previous scenario.

DESCRIPTION:

- The administrator can upload items into the website after proper verification.
- It will contain the books menu as per category.
- It will notify you about the current status of your delivery.

Multiple items can be added in a cart

LIST OF FIGURES

LIST OF TABLES

Table of Contents

CHAPTER-1

INTRODUCTION

1.1. An online shopping system that permits a customer to submit online orders for items and/or services from a store that serves both walk-in customers and online customers. The online shopping system presents an online display of an order cut off time and an associated delivery window for items selected by the customer. The system accepts the customer's submission of a purchase order for the item in response to a time of submission being before the order cut off time. The online shopping system does not settle with a credit supplier of the customer until the item selected by the customer is picked from inventory but before it is delivered. Therefore, the customer can go online and make changes to the order. In addition, available service windows are presented to the customer as a function of customer selected order and service types and further, the order picking is assigned in accordance with a picker's preference. When ordering goods, many shopping systems provide a virtual shopping cart for holding items selected for purchase. Successive items selected for purchase are placed into the virtual shopping cart until a customer completes their shopping trip. Virtual shopping carts may be examined at any time, and their contents can be edited or deleted at the option of the customer. Once the customer decides to submit a purchase order, the customer may print the contents of the virtual shopping basket in order to obtain a hard copy record of the transaction.

1.2. OVERVIEW OF PROJECT:

The Online Shopping system (OSS) application enables vendors to set up online shops, customers to browse through the shops, and a system administrator to approve and reject requests for new shops and maintain lists of shop categories. Also the developer is designing an online shopping site to manage the items in the shop and also help customers purchase them online without having to visit the shop physically.The online shopping system will use the internet as the sole method for selling goods to its consumers.

CHAPTER-2

SYSTEM ANALYSIS

2.1. PURPOSE OF THE SYSTEM

Create an active and closed system for selling of books in I.G.I.T from previous semesters which the seller no longer requires in current or upcoming semesters. This is for people who find it difficult to carry all the books accumulated at the end 4 years back home or for those who really find it "disheartening" to lend their books to juniors for free only to avoid the previous scenario. Users are free to sell other items also.

2.2. EXISTING SYSTEM:

Currently no system exists for online shopping in I.G.I.T.

CHAPTER-3

FEASIBILITY STUDY

Feasibility is the measure of how beneficial or practical the development of an Information System will be to client. Feasibility study is an important phase in the software development process. It enables the developer to have an assessment of the product being developed. It refers to the feasibility study of the product in terms of outcome of the product, operational use and technical support required for implementing it. Feasibility study should be performed on the basis of various criteria and parameters. The various feasibility studies are given below.

- Economic Feasibility
- Operational Feasibility
- Technical Feasibility

3.1. ECONOMICAL FEASIBILITY:

It is a measure of the cost effectiveness of a project or a solution. It refers to the benefits or out comes. If the benefits are more or less the same as the older system, then it is not feasible to develop product.The Website is highly efficient as it is developed to run on

IIS server and the frontend is coded in Asp.Net and the backend in C#. The Database is developed in SQLServer database.

3.2. OPERATIONAL FEASIBILITY:

It is a measure of how well the solution will work in the organization. It is a measure of how people feel about the system. It measures the urgency of the problem or acceptability of a solution. It refers to the feasibility of the product to be operational. Some products may work very well at design and implementation but may fail in the real time environment.

It includes the study of additional human resource required and their technical expertise. The product is operationally viable as it is designed specifically for the home users. My product is highly operationally feasible because of its user friendliness graphical user interface and it is developed in such a way it makes even the novices feel comfortable in using it.

3.3 TECHNICAL FEASIBILITY:

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at the point in time is not too many detailed design of the system, making it difficult to access issues like performance, costs on etc. A number of issues have to be considered while doing a technical analysis.

Understand the different technologies involved in the proposed system before commencing the project we have to be very clear about what are the technologies that are to be required for the development of the new system.

3.4 HARDWARE REQUIREMENTS:

Processor	: Pentium2 (266MHz)
Architecture	: x86 or x64
RAM	: Preferable 1GB
Hard disk	: Minimum of 512MB

3.5 SOFTWARE REQUIREMENTS:

Operating System	: windows (XP/Vista/7/8/8.1/10)
User Interface	: Active Server Page(ASP)

Programming Language	: C#
IDE	: Visual Studio
Database	: SQLServer 2015
Other	: .Net Framework 4.0 & above

CHAPTER-4

SOFTWARE REQUIREMENTS SPECIFICATIONS

Software Requirement Specification (SRS) is a complete description of the behaviour of system to be developed. It includes a set of use cases that describe all of the interactions that the users will have with the software. Data flow diagrams are also known as functional requirements. In addition to these, the SRS also contains non-functional requirements. Non-functional requirements are requirements which impose constraints on the design or implementation.

An SRS is basically an organization understands of a customer or potential clients' system requirements and dependencies at a particular point in time prior to actual design or development work. It is two way insurance policy that assures that the both client and the organization understand the other's requirements from that perspective at given point in time. The SRS document itself states in precise and explicit language those function and capabilities software must provide, as well as states any required constraints by which the system must abide. The SRS also functions as a blueprint for completing a project with a little cost of growth as possible. The SRS is often referred to as the "Parent" document because all subsequent project management documents, such as design specifications, statements of work, software architecture specifications, testing and validation plans and documentation plans are related to it. It's important to note that an SRS contains functional and non-functional requirements only, it doesn't offer design suggestions, possible solution to technology or business issues or any other information other than what the development team understands the costumer's system requirements to be. A well-designed, well-written SRS accomplishes four major goals:

It provides feedback to customer's assurance that the development organization understands the issues or problems to be solved and the software behaviour necessary to address those problems. Therefore, the SRS should be written in natural language, in an unambiguous manner that may also include charts, tables, data flow diagrams, and decision tables and so on.

It decomposes the problems into component parts. The simple act of writing down software requirements in a well-designed format organizes information, places borders around the problem, solidifies ideas, and helps breakdown the problem into its component parts in an orderly fashion. It serves as an input to the design specification. As mentioned previously, the SRS serves as the parent to the document to subsequent documents, such as the software design specification and statement of work. Therefore, the SRS must contain sufficient detail in the functional system requirements so that the design solution can be devised.

It serves as a product validation check. The SRS also serves as the parent document for testing and validation strategies that will applied to the requirements for verification. SRS's are typically developed during the first stages of "Requirement Development", which is initial product development phase in which information is gathered about what requirements are needed and not. This information gathering stage can include onsite visits, questionnaires, surveys interviews and perhaps a return on investment analysis of the customer or client's current business environment.

4.1. FUNCTIONAL REQUIREMENTS:

It deals with the functionalities required from the system which are as follows:

4.1.1. Administrator:

- Description: This module aims at admin user which a set of functionalities that supercede features available to the user. Those include:
    - Adding Products
    - Adding New Category of Products
    - Handle Customer Orders
    - View Products Details like Stock

- Operation: The admin first has to login with username and password as set in the web.config file. After that he can browse through the webpages available in the admin module.

4.1.2. User End:

- Description: The basic facilities that a user can avail in this module in he/she can shop products. The user can do the following operations:
  - Add Products to the cart
  - Remove from cart.
  - Fill in Customer Details
  - Track the Product Delivery Status.
- Operation: The Customer Opens the Home Page. He/She can browse the products based on Categories. If he wishes he can add products to the cart by clicking on the Add to Cart Button. Once added to cart he can click on the link and browse the added products. He/She can fill in the details about himself and place order for the product.

4.2. NON FUNCTIONAL REQUIREMENTS:

They are the quality requirements that stipulate how well software does what it has to do.

Performance

No. of client to be supported is dependent on the Web server(IIS) and database server(SQLServer) that we will use at the time of deployment.

Availability

Shopping Cart Website has 24*7 availability. It can be accessed for 24 hours a day. For this UPS support must be on the server site with a backup of at least 8 hours in case of power failure.

Reliability

It means the extent to which program performs with required precision. The Website developed should be extremely reliable and secure.

CHAPTER-5

SYSTEM DESIGN

Several popular software engineering approaches are based on the notion of data flow. The structured analysis/structured design (SA/SD) methodology is representative data approach. SA/SD begins with a single process or function that represents the overall purpose of desired software.SA/SD recursively divides complex processes, until one is left with many small functions that are easy to implement.

Analysis is concerned with the understanding and modelling the application and domain within which it operates. The initial input to the analysis phase is the problem statement, which describes the problem to be solved and provides a conceptual view of the proposal system. Subsequent dialog with the customer and real world background knowledge are additional inputs to analysis. The output from analysis is a formal model that captures the three essential aspects of the system: the objects and their relationships, the dynamic flow of control, and functional transformation of the data subject to constraints.

The following steps are performed in constructing the object model:

Identify the object process.

Prepare the data dictionary.

Identify the association between the objects.

Identify attributes of objects and links.

Organize or simplify the objects.

Verify the access paths.

Iterate and refine the model.

Group process into model.

During the analysis, the focus is on what needs to done, independent of how it is done. During design, decisions are made about how the problem will be solved, first at high level then increasing in detailed levels.

5.1. BREAKING THE SYSTEM INTO SUBSYSTEMS

The first step in subsystem design is to divide the system into subsystem into small number of component of a system is called a sub system. Each subsystem encompasses aspects of system that share some common property – similar functionality, the same physical location or execution in the same kind of hardware.

5.2. IDENTIFYING CONCURRENCY

One important goal of system design identifies which objects must have active concurrently and which objects have activity that is mutually exclusive. The latter objects can be folded together in single threads of control or task. But there is no part that is concurrent in our system.

5.3. ALLOCATING SUBSYSTEM TO PROCESSOR:

In this step system designer estimates the hardware resources required and the implementation choice either hardware or software. The hardware requirements are pentium2 (266MHz), minimum 1GB of RAM.

5.4. MANAGEMENT OF DATA STORES:

In this stage the system designer decides what format is used to store the data. There are DBMS system or file systems and others. Here in my project there are pictures and texts. We then definitely prefer pictures and text to store in SQLServer database and retrieve pictures and the texts.

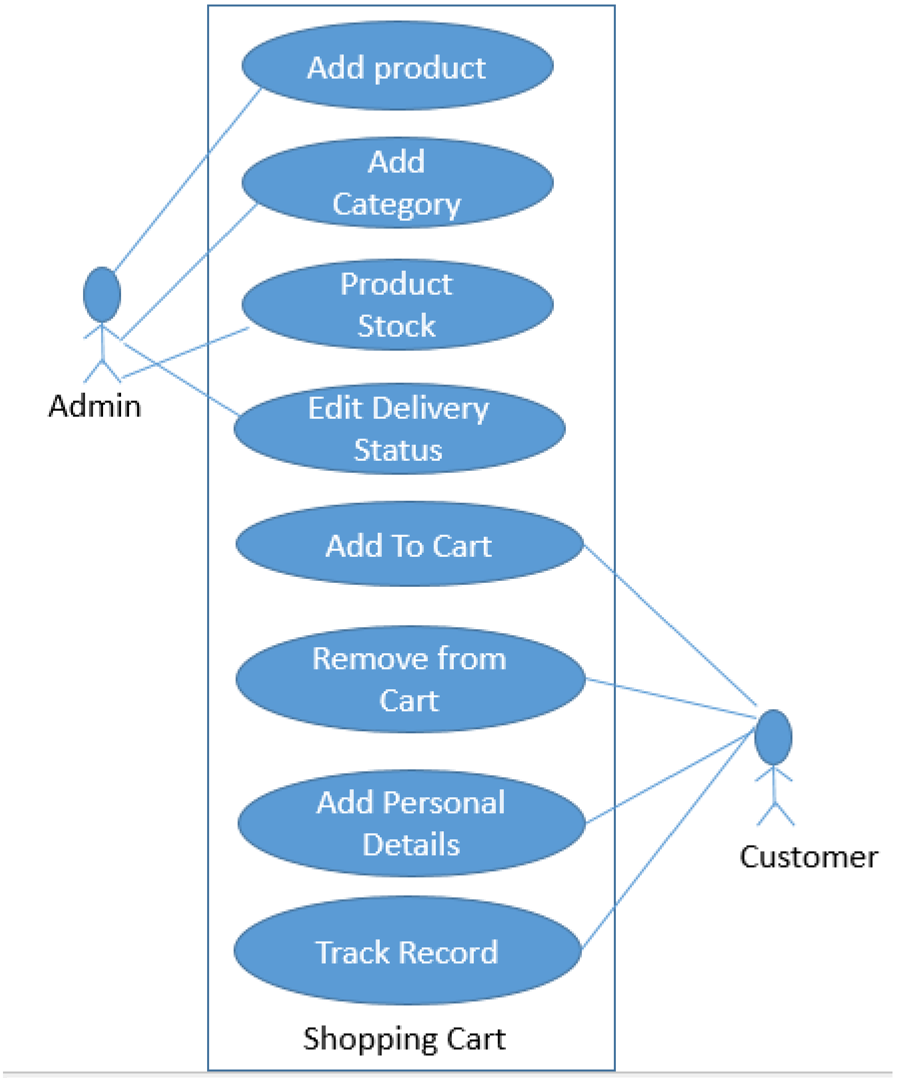
5.5. FUNCTIONAL MAPPING:

The functional model shows how the values computed without regard for the sequencing, decision or structures. The functional model shows which value depend on which other values and the functions that relate them.

5.6. SOFTWARE ARCHITECTURE:

Use case Diagram is used to specify the relationship between use cases and the users. It describe what function a user can perform using the system. Where use cases are the function and the users are said to actor those who are allowed to perform functions using system.

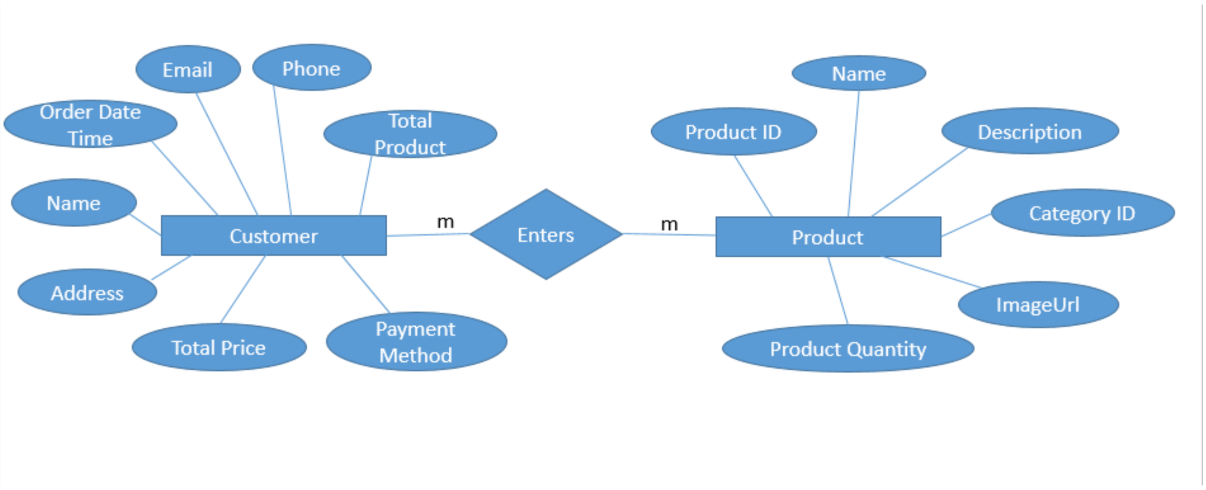
FRONTEND DESIGN-



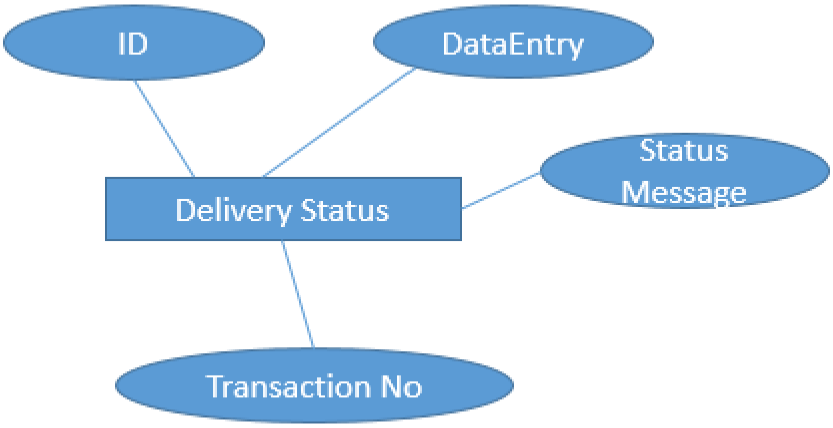
(Level 0)

BACKEND DESIGN-

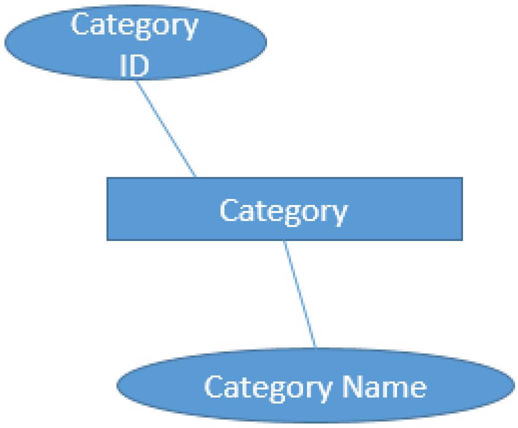
For backend design E-R diagram is used, which describes relationship between different entities and their characteristic.



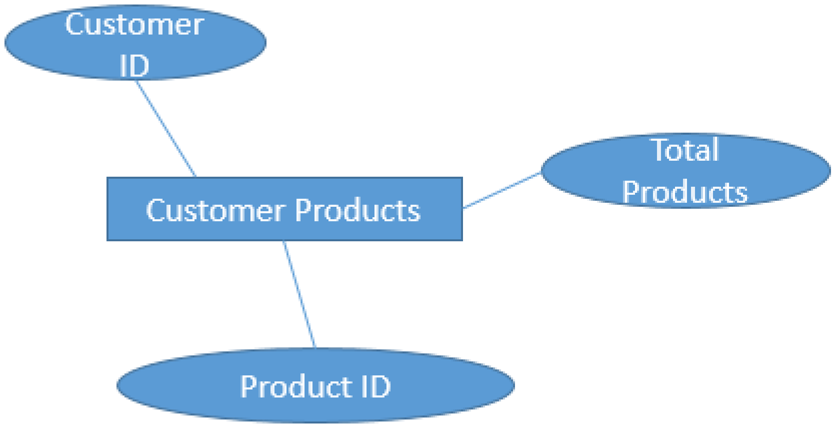
ER Diagram for Customer and Product Table



ER Diagram for Delivery Status

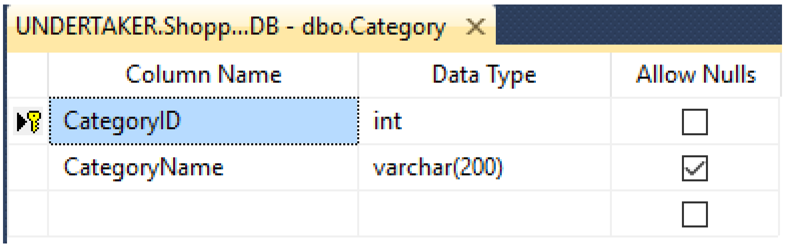


ER Diagram for Category



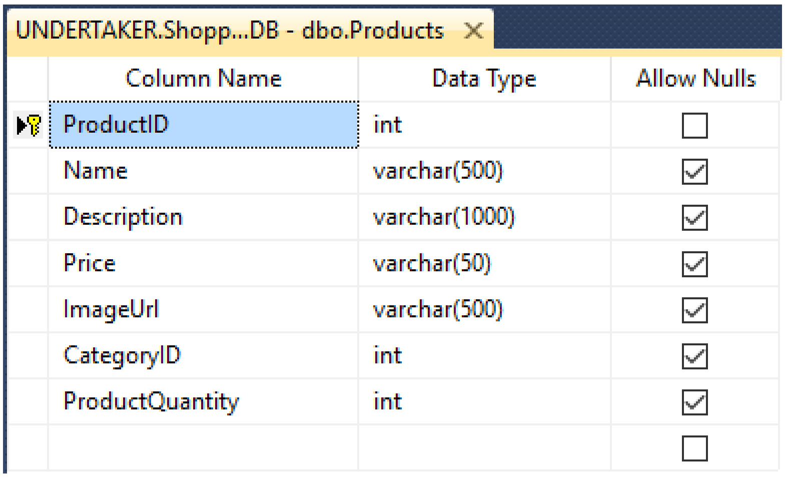
ER Diagram for Customer Products

TABLES-



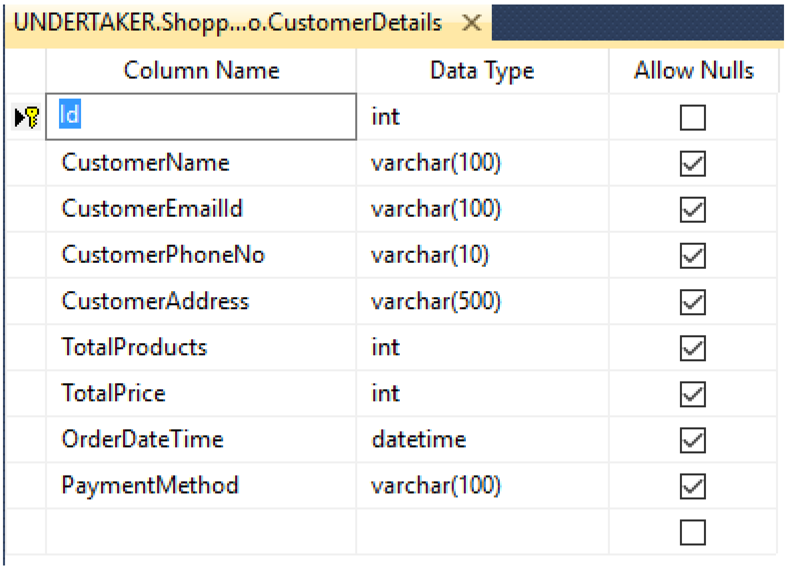
Column Name	Data Type	Allow Nulls
CategoryID	int	☐
CategoryName	varchar(200)	☑
		☐

Table for ProductCategory



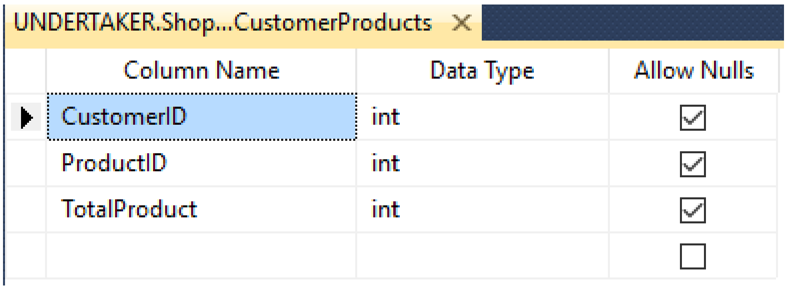
Column Name	Data Type	Allow Nulls
ProductID	int	☐
Name	varchar(500)	☑
Description	varchar(1000)	☑
Price	varchar(50)	☑
ImageUrl	varchar(500)	☑
CategoryID	int	☑
ProductQuantity	int	☑
		☐

Table for Products



Column Name	Data Type	Allow Nulls
Id	int	☐
CustomerName	varchar(100)	☑
CustomerEmailId	varchar(100)	☑
CustomerPhoneNo	varchar(10)	☑
CustomerAddress	varchar(500)	☑
TotalProducts	int	☑
TotalPrice	int	☑
OrderDateTime	datetime	☑
PaymentMethod	varchar(100)	☑
		☐

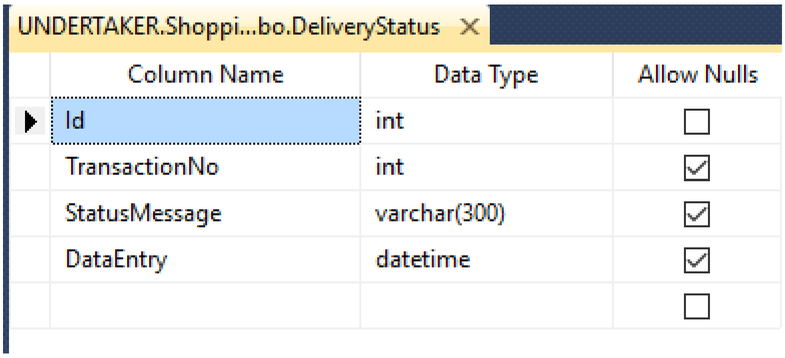
Table for Customer Details



UNDERTAKER.Shop...CustomerProducts

Column Name	Data Type	Allow Nulls
CustomerID	int	☑
ProductID	int	☑
TotalProduct	int	☑
		□

Table for Customer Products



UNDERTAKER.Shoppi...bo.DeliveryStatus

Column Name	Data Type	Allow Nulls
Id	int	□
TransactionNo	int	☑
StatusMessage	varchar(300)	☑
DataEntry	datetime	☑
		□

Table for Delivery Status

CHAPTER-6

CODING

6.1. WEBSITE PART:

LogIn.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Login.aspx.cs"
Inherits="ShoppingCart.Admin.Login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <style type="text/css">
        .auto-style1 {
            width: 149px;
        }
    </style>
</head>
<body style="height: 401px">
    <form id="form1" runat="server">
        <table style="width: 75%; height: 194px; margin-left: 104px; margin-top:
92px;">
            <tr>
                <th class="auto-style1" colspan="2">
```

```
                    <asp:Label ID="Label2" runat="server" Text="OLD BOOKS SHOPPING 
CART"></asp:Label>
                  <hr />
                </th>
            </tr>

            <tr>
                <th class="auto-style1">LoginId:</th>
                <td>
                    <asp:TextBox ID="txtLoginId" runat="server" 
Width="151px"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <th class="auto-style1">Password:</th>
                <td>
                    <asp:TextBox ID="txtPassword" runat="server" Width="149px" 
TextMode="Password"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <th class="auto-style1">&nbsp;</th>
                <th>
                    <asp:Button ID="btnLogin" runat="server" OnClick="btnLogin_Click" 
Text="LOGIN" />
                </th>
            </tr>
            <tr>
                <th class="auto-style1" colspan="2">
                    <hr />
                    <asp:Label ID="lblAlert" runat="server"></asp:Label>

                </th>
                <th>

                    <asp:Button ID="Button1" runat="server" Text="Back To Home" 
OnClick="Button1_Click" />

                </th>

            </tr>
        </table>
    <div>

    </div>
    </form>
</body>
</html>
```

AddNewProduct.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Login.aspx.cs" 
Inherits="ShoppingCart.Admin.Login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <style type="text/css">
```

```
        .auto-style1 {
            width: 149px;
        }
    </style>
</head>
<body style="height: 401px">
    <form id="form1" runat="server">
        <table style="width: 75%; height: 194px; margin-left: 104px; margin-top: 
92px;">
            <tr>
                <th class="auto-style1" colspan="2">
                    <asp:Label ID="Label2" runat="server" Text="OLD BOOKS SHOPPING 
CART"></asp:Label>
                  <hr />
                </th>
            </tr>

            <tr>
                <th class="auto-style1">LoginId:</th>
                <td>
                    <asp:TextBox ID="txtLoginId" runat="server" 
Width="151px"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <th class="auto-style1">Password:</th>
                <td>
                    <asp:TextBox ID="txtPassword" runat="server" Width="149px" 
TextMode="Password"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <th class="auto-style1">&nbsp;</th>
                <th>
                    <asp:Button ID="btnLogin" runat="server" OnClick="btnLogin_Click" 
Text="LOGIN" />
                </th>
            </tr>
            <tr>
                <th class="auto-style1" colspan="2">
                    <hr />
                    <asp:Label ID="lblAlert" runat="server"></asp:Label>

                </th>
                <th>

                    <asp:Button ID="Button1" runat="server" Text="Back To Home" 
OnClick="Button1_Click" />

                </th>

            </tr>
        </table>
    <div>

    </div>
    </form>
</body>
</html>
```

AddEditCategory.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Login.aspx.cs"
Inherits="ShoppingCart.Admin.Login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <style type="text/css">
        .auto-style1 {
            width: 149px;
        }
    </style>
</head>
<body style="height: 401px">
    <form id="form1" runat="server">
        <table style="width: 75%; height: 194px; margin-left: 104px; margin-top:
92px;">
            <tr>
                <th class="auto-style1" colspan="2">
                    <asp:Label ID="Label2" runat="server" Text="OLD BOOKS SHOPPING
CART"></asp:Label>
                  <hr />
                </th>
            </tr>

            <tr>
                <th class="auto-style1">LoginId:</th>
                <td>
                    <asp:TextBox ID="txtLoginId" runat="server"
Width="151px"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <th class="auto-style1">Password:</th>
                <td>
                    <asp:TextBox ID="txtPassword" runat="server" Width="149px"
TextMode="Password"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <th class="auto-style1">&nbsp;</th>
                <th>
                    <asp:Button ID="btnLogin" runat="server" OnClick="btnLogin_Click"
Text="LOGIN" />
                </th>
            </tr>
            <tr>
                <th class="auto-style1" colspan="2">
                    <hr />
                    <asp:Label ID="lblAlert" runat="server"></asp:Label>

                </th>
                <th>
```

```
                    <asp:Button ID="Button1" runat="server" Text="Back To Home" 
OnClick="Button1_Click" />

                </th>

            </tr>
        </table>
    <div>
    
    </div>
    </form>
</body>
</html>
```

Category.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/AdminMaster.Master" 
AutoEventWireup="true" CodeBehind="Category.aspx.cs" 
Inherits="ShoppingCart.Admin.Category" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    <tr>
    <td>
        <div>
            <asp:Label ID="lblTitle" runat="server" Text="Label">All 
Categories</asp:Label>
            <hr />
        </div>
        <table style="width: 100%;">
            <tr>
                <td align="center">
                    <asp:GridView ID="gvAvailableCategories" runat="server" 
BackColor="White" BorderColor="#999999" BorderStyle="None" Width="100%" 
CellPadding="3" GridLines="Vertical">
                        <AlternatingRowStyle BackColor="#DCDCDC" />
                        <FooterStyle BackColor="#CCCCCC" ForeColor="Black" />
                        <HeaderStyle BackColor="#000084" Font-Bold="True" 
ForeColor="White" />
                        <PagerStyle BackColor="#999999" ForeColor="Black" 
HorizontalAlign="Center" />
                        <RowStyle BackColor="#EEEEEE" ForeColor="Black" />
                        <SelectedRowStyle BackColor="#008A8C" Font-Bold="True" 
ForeColor="White" />
                        <SortedAscendingCellStyle BackColor="#F1F1F1" />
                        <SortedAscendingHeaderStyle BackColor="#0000A9" />
                        <SortedDescendingCellStyle BackColor="#CAC9C9" />
                        <SortedDescendingHeaderStyle BackColor="#000065" />
                    </asp:GridView>
                </td>
            </tr>
        </table>
    </td>
   </tr>


</asp:Content>
```

ProductStock.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/AdminMaster.Master"
AutoEventWireup="true" CodeBehind="ProductStock.aspx.cs"
Inherits="ShoppingCart.Admin.ProductStock" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    <tr>
        <td>
            <asp:Label ID="lblTitle" runat="server"
Text="Label">ProductStock</asp:Label>
            <hr />
            <table style="width: 100%;">
                <tr>
                    <td align="center">
                        <asp:DropDownList ID="ddlCategory" runat="server"
AutoPostBack="true"
                            OnSelectedIndexChanged="ddlCategory_SelectedIndexChanged"
Height="16px" Width="197px" ></asp:DropDownList>
                    </td>

                </tr>
                <tr>
                    <td align="center">
                        <asp:Panel ID="pnlProductStockReport" runat="server">
                            <hr />
                            <asp:RadioButtonList ID="rblProductStock" runat="server"
AutoPostBack="true"

OnSelectedIndexChanged="rblProductStock_SelectedIndexChanged"
RepeatDirection="Horizontal">
                                <asp:ListItem Value="0"
Selected="True">OutOfStock</asp:ListItem>
                                <asp:ListItem Value="1">LimitedStock</asp:ListItem>
                                <asp:ListItem Value="2">EnoughStock</asp:ListItem>
                            </asp:RadioButtonList>
                            <hr />
                        </asp:Panel>
                    </td>

                </tr>
                <tr>
                    <td align="center">

                            <asp:Label ID="NoRecordsToDisplay" runat="server"
Text="Label">No Records Available</asp:Label>

                        <asp:GridView ID="gvAvailableStock" runat="server"
BackColor="White" BorderColor="#CC9966"
                             BorderStyle="None" BorderWidth="1px" CellPadding="4"
AutoGenerateColumns="false">
                            <Columns>
                                <asp:BoundField DataField="CategoryName"
HeaderText="ProductCategory" />
                                <asp:BoundField DataField="AvailableStock"
HeaderText="AvailableStock" />

                                <asp:BoundField DataField="Price" HeaderText="Price"
/>
                                <asp:ImageField DataImageUrlField="ImageUrl">
                                </asp:ImageField>

                            </Columns>
                            <FooterStyle BackColor="#FFFFCC" ForeColor="#330099" />
                            <HeaderStyle BackColor="#990000" Font-Bold="True" 
ForeColor="#FFFFCC" />
                            <PagerStyle BackColor="#FFFFCC" ForeColor="#330099" 
HorizontalAlign="Center" />
                            <RowStyle BackColor="White" ForeColor="#330099" />
                            <SelectedRowStyle BackColor="#FFCC66" Font-Bold="True" 
ForeColor="#663399" />
                            <SortedAscendingCellStyle BackColor="#FEFCEB" />
                            <SortedAscendingHeaderStyle BackColor="#AF0101" />
                            <SortedDescendingCellStyle BackColor="#F6F0C0" />
                            <SortedDescendingHeaderStyle BackColor="#7E0000" />

                        </asp:GridView>
                    </td>

                </tr>
            </table>
        </td>
    </tr>
</asp:Content>
```

Products.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/AdminMaster.Master" 
AutoEventWireup="true" CodeBehind="Products.aspx.cs" 
Inherits="ShoppingCart.Admin.Products" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    <tr>
     <td align="center">
         <asp:Label ID="lblTitle" runat="server" Text="Label">All Products</asp:Label>
         <asp:GridView ID="gvAvailableProducts" runat="server" Width="100%" 
BackColor="White" BorderColor="White" 
             BorderStyle="Ridge" BorderWidth="2px" CellPadding="3" CellSpacing="1" 
GridLines="None" AutoGenerateColumns="false">
             <Columns>
                 <asp:BoundField DataField="Name" HeaderText="ProductName" />
                 <asp:BoundField DataField="CategoryName" HeaderText="ProductCategory" 
/>
                 <asp:BoundField DataField="AvailableStock" 
HeaderText="AvailableStock" />
                 <asp:BoundField DataField="Price" HeaderText="Price" />
                 <asp:ImageField DataImageUrlField="ImageUrl">
                 </asp:ImageField>
             </Columns>
             <FooterStyle BackColor="#C6C3C6" ForeColor="Black" />
             <HeaderStyle BackColor="#4A3C8C" Font-Bold="True" ForeColor="#E7E7FF" />
             <PagerStyle BackColor="#C6C3C6" ForeColor="Black" HorizontalAlign="Right" 
/>
             <RowStyle BackColor="#DEDFDE" ForeColor="Black" />
             <SelectedRowStyle BackColor="#9471DE" Font-Bold="True" ForeColor="White" 
/>
             <SortedAscendingCellStyle BackColor="#F1F1F1" />
             <SortedAscendingHeaderStyle BackColor="#594B9C" />
             <SortedDescendingCellStyle BackColor="#CAC9C9" />
```

```
                <SortedDescendingHeaderStyle BackColor="#33276A" />
            </asp:GridView>
            <hr />

        </td>
      </tr>
</asp:Content>
```

CustomerOrders.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Admin/AdminMaster.Master"
AutoEventWireup="true" CodeBehind="CustomerOrders.aspx.cs"
Inherits="ShoppingCart.Admin.CustomerOrders" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    <tr>
        <td>
             <div>
                 <asp:Label ID="lblTitle" runat="server" Text="Label">Customer
Orders</asp:Label>
                 <hr />
            </div>
    <table style="width: 100%;">
        <tr>
            <td>
                <asp:GridView ID="gvCustomerOrders" runat="server" BackColor="White"
BorderColor="#999999" BorderStyle="None" BorderWidth="1px" CellPadding="3"
                    GridLines="Vertical" width="100%" AutoGenerateColumns="false"
OnSelectedIndexChanged="gvCustomerOrders_SelectedIndexChanged">
                    <AlternatingRowStyle BackColor="#DCDCDC" />
                    <Columns>
                        <asp:BoundField DataField="Id" HeaderText="Id" />
                        <asp:BoundField DataField="CustomerName" HeaderText="Name" />
                        <asp:BoundField DataField="CustomerPhoneNo"
HeaderText="PhoneNo" />
                        <asp:BoundField DataField="TotalProducts"
HeaderText="Products" />
                        <asp:BoundField DataField="TotalPrice" HeaderText="Price" />
                        <asp:TemplateField>
                            <ItemTemplate>
                                <asp:HyperLink ID="HyperLink1" runat="server"
Text="View Details" NavigateUrl='<%# Eval("Id","~/Admin/OrderDetails.aspx?Id={0}")
%>'>

                                </asp:HyperLink>
                            </ItemTemplate>
                        </asp:TemplateField>
                    </Columns>
                    <FooterStyle BackColor="#CCCCCC" ForeColor="Black" />
                    <HeaderStyle BackColor="#000084" Font-Bold="True"
ForeColor="White" />
                    <PagerStyle BackColor="#999999" ForeColor="Black"
HorizontalAlign="Center" />
                    <RowStyle BackColor="#EEEEEE" ForeColor="Black" />
                    <SelectedRowStyle BackColor="#008A8C" Font-Bold="True"
ForeColor="White" />
                    <SortedAscendingCellStyle BackColor="#F1F1F1" />
                    <SortedAscendingHeaderStyle BackColor="#0000A9" />
                    <SortedDescendingCellStyle BackColor="#CAC9C9" />
                    <SortedDescendingHeaderStyle BackColor="#000065" />

                </asp:GridView>
            </td>

        </tr>

    </table>
            </td>
        </tr>
</asp:Content>
```

OrderDetails.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="OrderDetails.aspx.cs" 
Inherits="ShoppingCart.Admin.OrderDetails" %>

<%@ Register src="../usCustomerOrder.ascx" tagname="usCustomerOrder" tagprefix="uc1" 
%>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Button ID="btnClear" runat="server" Text="Clear & Search Other 
Transactions" OnClick="btnClear_Click" />
        </div>
        <hr />
        <br />
        <uc1:usCustomerOrder ID="usCustomerOrder1" runat="server" />



    </form>
</body>
</html>
```

AdminMaster.Master

```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="AdminMaster.master.cs" 
Inherits="ShoppingCart.Admin.AdminMaster" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <asp:ContentPlaceHolder ID="head" runat="server">
    </asp:ContentPlaceHolder>
    <style type="text/css">
        .auto-style1 {
            width: 276px;
        }
    </style>
```

```
</head>
<body>
    <form id="form1" runat="server">
        <asp:ScriptManager ID="ScriptManager1" runat="server">
        </asp:ScriptManager>
        <asp:UpdatePanel ID="AdminUpdatePanel" runat="server">
                <ContentTemplate>
                         <table style="width: 100%;">
            <tr>
               <th><h1>SHOPPING CART</h1> <a href="../Default.aspx">Back to Home</a>
                   <hr />
               </th>

            </tr>
            <tr>
                <td>
                <table style="width: 100%;">
                    <tr>
                        <td><a href="AddEditCategory.aspx">Category</a>&nbsp
                            <a href="AddNewProducts.aspx">Products</a>&nbsp
                            <a href="Category.aspx">Categories</a>&nbsp
                            <a href="Products.aspx">All Products</a>&nbsp
                            <a href="CustomerOrders.aspx">Customer Orders</a>&nbsp
                            <a href="ProductStock.aspx">Products Stocks</a>&nbsp


                        </td>
                    </tr>
                </table>
                </td>
            </tr>
                             <asp:ContentPlaceHolder ID="ContentPlaceHolder1"
runat="server">
                             </asp:ContentPlaceHolder>
            <tr>
                <th>
                    BRP BOB@ oldbooksshoppingcart.com
                </th>

            </tr>
        </table>
                </ContentTemplate>
            </asp:UpdatePanel>


    </form>
</body>
</html>
```

Default.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="ShoppingCart.Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">    <title>Shopping Cart Home Page</title>
    <style type="text/css">
        .style1{
            width:916px;
            text-align:center;
        }
        .style2{
            width: 633px;
            text-align:left;
        }
        .style3{
            width: 257px;
            text-align:center;
        }
        .style4{
            width: 185px;
            text-align:center;
        }
        .style6{
            width: 260px;
            text-align:left;
        }
        .style8{
            width: 180px;
            text-align:center;
        }
        .style7{
            width: 427px;
            text-align:center;
        }

        .auto-style1 {
            width: 257px;
        }

        .auto-style2 {
            width: 64px;
        }
        style8{
            width:100%;
        }

        .auto-style3 {
            width: 633px;
            text-align: left;
            height: 23px;
        }
        .auto-style4 {
            width: 257px;
            text-align: center;
            height: 23px;
        }

        .auto-style5 {
            width: 796px;
        }
        #imgProductPhoto {
            height: 294px;
        }

    </style>
</head>
<body>
```

```
    <form id="form1" runat="server">
        <asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>
        <asp:UpdatePanel ID="UpdatePanel1" runat="server">
        <ContentTemplate >
            <table width="100%" style="align-content:center">
                <tr>
                    <td>
                        <table class="style1" width="100%">
                            <tr>
                                <td class="style3">
                                    <asp:Image ID="Image1" runat="server" 
ImageUrl="~/Images/KK--shrt-blue.gif" Width="111px"/>
                                    &nbsp</td>
                                <td class="style6">
                                    <asp:LinkButton ID="lblLogo" Text="Shopping Cart" 
runat="server" OnClick="lblLogo_Click">
                                        <br />
                                        <h2>For Those who Love Online Shopping!!!</h2>
                                    </asp:LinkButton>

                                </td>
                                <td class="style7" rowspan="2">
                                    <asp:Image ID="Image2" runat="server" 
ImageUrl="~/books.png" Width="356px" />

                                </td>
                                <td style="text-align:right" rowspan="2" class="auto-
style2">
                                    <asp:Image ID="Image3" runat="server" 
ImageUrl="~/images1.png" Height="55px" Width="64px" />

                                </td>
                                <td>
                                    <asp:LinkButton ID="btnShoppingCart" 
runat="server" OnClick="btnShoppingCart_Click">0</asp:LinkButton>

                                </td>
                            </tr>
                            <tr>
                                <td class="style3">&nbsp;</td>

                            </tr>
                            <tr>
                                <td class="auto-style1">&nbsp;</td>
                                <td>&nbsp;</td>
                                <td>&nbsp;</td>
                            </tr>
                        </table>
                    </td>
                </tr>
                <tr>
                    <td>
                        <table class="style1">
                            <tr>
                                <td class="auto-style3">
                                    <asp:Label ID="lblCategoryName" 
runat="server"></asp:Label>

                                </td>
                                <td class="auto-style4">
                                    <asp:Label ID="lblProducts" runat="server"
Text="Products"></asp:Label>
                                </td>
                            </tr>
                        </table>
                    </td>

                </tr>
                <tr>
                    <td>
                        <table class="style1">
                            <tr>
                                <td class="style2">
                                    <asp:Panel ID="pnlProducts" ScrollBars="Auto"
Height="487px" BorderColor="Black"
                                        BorderStyle="Inset" BorderWidth="1px"
runat="server">
                                         <asp:DataList ID="dlProducts"
RepeatColumns="3" Width="100%" runat="server" BackColor="White" BorderColor="#3366CC"
BorderStyle="None" BorderWidth="1px" CellPadding="4" GridLines="Both" Height="1077px"
style="margin-right: 194px" OnSelectedIndexChanged="dlProducts_SelectedIndexChanged">
                                             <FooterStyle BackColor="#99CCCC"
ForeColor="#003399" />
                                             <HeaderStyle BackColor="#003399" Font-
Bold="True" ForeColor="#CCCCFF" />
                                             <ItemStyle BackColor="White"
ForeColor="#003399" />
                                             <ItemTemplate>
                                                 <table style="width:61%;
height:399px; align-items: center" class="style4" cellspacing="1">
                                                     <tr>
                                                         <td>
                                                             <asp:Label
ID="lblProductName" runat="server" Text='<%# Bind("Name") %>'></asp:Label>
                                                         </td>
                                                     </tr>
                                                     <tr >
                                                         <td>
                                                             <img alt="" src='<%#
Eval("ImageUrl") %>' id="imgProductPhoto"/> <!--DataBinding is not possible-->

                                                         </td>
                                                     </tr>

                                                     <tr>
                                                         <td>Price:
                                                             <asp:Label ID="lblPrice"
runat="server" Text='<%# Bind("Price") %>'></asp:Label><br />
                                                             Stock=&nbsp;
                                                             <asp:Label
ID="lblAvailableStock" runat="server" Text='<%# Eval("AvailableStock")
%>'></asp:Label>
                                                             <asp:HiddenField
ID="hfProductID" runat="server" Value='<%# Bind("ProductID") %>' />
                                                         </td>
                                                     </tr>
                                                     <tr>
                                                         <td>
```

```
                                                             <asp:Button
ID="btnAddToCart" runat="server" Height="24px" OnClick="btnAddToCart_Click"
Width="100%" CommandArgument='<%# Eval("ProductID") %>' CausesValidation="false"
Text="ADD TO CART" />
                                                         </td>
                                                     </tr>
                                                 </table>
                                             </ItemTemplate>
                                             <SelectedItemStyle BackColor="#009999"
Font-Bold="True" ForeColor="#CCFF99" />
                                         </asp:DataList>
                                    </asp:Panel>

                                    <asp:Panel ID="pnlMyCart" runat="server"
ScrollBars="Auto" Height="500px" BorderColor="Black"
                                         BorderStyle="Inset" BorderWidth="1px"
Visible="false">
                                        <table cellspacing="1" style="align-
content:center" >
                                            <tr>
                                                <td style="align-content:center"
class="auto-style5">
                                                    <asp:Label
ID="lblAvailableStockAlert" runat="server"></asp:Label><!-- alerting about the
inavailability of stock ||

Binding the products that are added to the cart-->
                                                    <asp:DataList ID="dlCartProducts"
runat="server" RepeatColumns="3" Width="100%"
OnSelectedIndexChanged="dlCartProducts_SelectedIndexChanged">
                                                        <ItemTemplate>
                                                            <table style="width:100%;
height:256px; align-items: center" class="style4" cellspacing="1">
                                                     <tr>
                                                         <td>
                                                             <asp:Label
ID="lblProductName" runat="server" Text='<%# Eval("Name") %>'></asp:Label>
                                                         </td>
                                                     </tr>
                                                     <tr >
                                                         <td>
                                                             <img alt=""
src='<%#Eval("ImageUrl") %>' id="imgProductPhoto" width="173px" height="160px" /> <!--
DataBinding is not possible-->

                                                         </td>
                                                     </tr>

                                                     <tr>
                                                         <td>
                                                              Stock=&nbsp;
                                                             <asp:Label
ID="lblAvailableStock" runat="server" Text='<%# Eval("AvailableStock")
%>'></asp:Label>
                                                             <br />
                                                             Price:
                                                             <asp:Label ID="lblPrice"
runat="server" Text='<%# Eval("Price") %>'></asp:Label>&nbsp;&nbsp;<br />
                                                             Quantity:
```

```
                                                             <asp:TextBox
ID="txtProductQuantity" Width="10px" Height="10px" MaxLength="1" AutoPostBack="True"
runat="server"

OnTextChanged="txtProductQuantity_TextChanged" Text='<%# Eval("ProductQuantity")
%>'></asp:TextBox>
                                                             <br />

                                                             <asp:HiddenField
ID="hfProductID" runat="server" Value='<%# Eval("ProductID") %>' />
                                                         </td>
                                                     </tr>
                                                     <tr>
                                                         <td>
                                                             <asp:Button
ID="btnRemoveFromCart" runat="server" Height="24px" OnClick="btnRemoveFromCart_Click"

CausesValidation="false" Width="100%" CommandArgument='<%# Eval("ProductID") %>'
                                                                  Text="REMOVE FROM
CART" />
                                                         </td>
                                                     </tr>
                                                 </table>
                                                        </ItemTemplate>
                                                        <ItemStyle Width="33%" />
                                                    </asp:DataList>
                                                </td>

                                            </tr>
                                            <tr>
                                                <td align="center" class="auto-
style5">&nbsp;</td>

                                            </tr>
                                            <tr>
                                                <td align="center" class="auto-
style5">&nbsp;</td>

                                            </tr>
                                        </table>
                                    </asp:Panel>
                                </td>
                                <td class=" style3" style="align-content:center">
                                    <asp:Panel ID="pnlCategories" ScrollBars="Auto"
Height="500px" BorderColor="Black"
                                         BorderStyle="Inset" BorderWidth="1px"
runat="server" style="margin-left: 0px">

                                        <asp:DataList ID="dlCategories"
CellPadding="3" Width="252px" GridLines="Vertical" runat="server"

OnSelectedIndexChanged="dlCategories_SelectedIndexChanged" BackColor="White"
BorderColor="#999999"
                                            BorderStyle="None" BorderWidth="1px"
Height="229px" style="margin-left: 0px">
                                            <AlternatingItemStyle BackColor="#DCDCDC"
/>
                                            <FooterStyle BackColor="#CCCCCC"
ForeColor="Black" />
```

```
                                            <HeaderStyle BackColor="#000084" Font-
Bold="True" ForeColor="White" />
                                            <ItemStyle BackColor="#EEEEEE"
ForeColor="Black" />
                                            <ItemTemplate>
                                                <asp:LinkButton ID="lbtnCategory"
                                                     OnClick="lbtnCategory_Click"
runat="server" CommandArgument='<%# Bind("CategoryID") %>' Text='<%#
Bind("CategoryName") %>'></asp:LinkButton>
                                            </ItemTemplate>
                                            <ItemStyle Width="33%" />
                                            <SelectedItemStyle BackColor="#008A8C"
Font-Bold="true" ForeColor="White"/>
                                        </asp:DataList>
                                    </asp:Panel>


                                    <asp:Panel ID="pnlCheckOut" runat="server"
ScrollBars="Auto" Height="500px" BorderColor="Black"
                                        BorderStyle="Inset" BorderWidth="1px"
Visible="false">

                                        <table width="100">
                                            <tr>
                                                <td align="left">Name</td>

                                            </tr>
                                            <tr>
                                                <td>
                                                    <asp:TextBox ID="txtCustomerName"
runat="server"></asp:TextBox>
                                                    <asp:RequiredFieldValidator
ID="RequiredFieldValidator1" runat="server"
                                                        ErrorMessage="*"
ControlToValidate="txtCustomerName" ForeColor="Red">
                                                    </asp:RequiredFieldValidator>
                                                </td>

                                            </tr>
                                            <tr>
                                                <td align="left">Phone No:</td>
                                            </tr>
                                             <tr>
                                                <td>
                                                    <asp:TextBox
ID="txtCustomerPhoneNo" MaxLength="10" runat="server"></asp:TextBox>
                                                    <asp:RequiredFieldValidator
ID="RequiredFieldValidator2" runat="server"
                                                        ErrorMessage="*"
ControlToValidate="txtCustomerPhoneNo" ForeColor="Red">
                                                    </asp:RequiredFieldValidator>

                                                </td>

                                            </tr>
                                            <tr>
                                                <td align="left">Email ID</td>

                                            </tr>
                                            <tr>
                                                <td>
                                                    <asp:TextBox
ID="txtCustomerEmailID" runat="server"></asp:TextBox>
                                                     <asp:RequiredFieldValidator

ID="RequiredFieldValidator3" runat="server"
                                                        ErrorMessage="*"

ControlToValidate="txtCustomerEmailID" ForeColor="Red">
                                                    </asp:RequiredFieldValidator>
                                                </td>
                                            </tr>
                                             <tr>
                                                <td>Address</td>

                                            </tr>
                                            <tr>
                                                <td>
                                                    <asp:TextBox
ID="txtCustomerAddress" runat="server" TextMode="MultiLine" Height="81px"
Width="250px"></asp:TextBox>
                                                     <asp:RequiredFieldValidator

ID="RequiredFieldValidator4" runat="server"
                                                        ErrorMessage="*"

ControlToValidate="txtCustomerAddress" ForeColor="Red">
                                                    </asp:RequiredFieldValidator>
                                                </td>

                                            </tr>
                                            <tr>
                                                <td align="left">Total Products</td>

                                            </tr>
                                             <tr>
                                                <td>
                                                    <asp:TextBox ID="txtTotalProducts"
runat="server" ReadOnly="true"></asp:TextBox>
                                                     <asp:RequiredFieldValidator

ID="RequiredFieldValidator5" runat="server"
                                                        ErrorMessage="*"

ControlToValidate="txtTotalProducts" ForeColor="Red">
                                                    </asp:RequiredFieldValidator>
                                                 </td>

                                            </tr>
                                            <tr>
                                                <td align="left">Total Price</td>
                                            </tr>
                                            <tr>
                                                <td>
                                                    <asp:TextBox ID="txtTotalPrice"
runat="server"></asp:TextBox>

                                                    <asp:RequiredFieldValidator

ID="RequiredFieldValidator6" runat="server"
                                                        ErrorMessage="*"

ControlToValidate="txtTotalPrice" ForeColor="Red">
                                                    </asp:RequiredFieldValidator>
                                                </td>

                                            </tr>
```

```
                                             <tr>
                                                <td>
                                                    Payment Mode:</td>

                                            </tr>
                                            <tr>
                                                <td>
                                                    <asp:RadioButtonList
ID="rblPaymentMethod" runat="server">
                                                        <asp:ListItem Value="1">Cash
On Delivery</asp:ListItem>
                                                        <asp:ListItem Value="2"
Selected="False">Payment Gateway</asp:ListItem>
                                                    </asp:RadioButtonList>
                                                </td>

                                            </tr>
                                            <tr>
                                                <td align="left">
                                                    <asp:Button ID="btnPlaceOrder"
runat="server" OnClick="btnPlaceOrder_Click" Text="Place Order" style="height: 26px"
/>

                                                    <td>&nbsp;</td>
                                                </td>

                                            </tr>
                                            <tr>
                                                <td>
                                                    <asp:RegularExpressionValidator
ID="RegularExpressionValidator1" runat="server"
                                                        ErrorMessage="Please Enter a
Valid Email ID" ControlToValidate="txtCustomerEmailID"
                                                        ForeColor="Red"
ValidationExpression="\w+([-+.']\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*">
                                                    </asp:RegularExpressionValidator>
                                                </td>
                                            </tr>
                                       </table>

                                    </asp:Panel>
                                </td>
                            </tr>
                            <tr>
                                <td colspan="2">
                                    <asp:Panel ID="pnlEmptyCart" runat="server"
Visible="false">
                                        <br />
                                        <asp:Image ID="Image4"
ImageUrl="~/Images/emptycart.jpg" runat="server" />
                                        <br />
                                    </asp:Panel>
                                    <asp:Panel ID="pnlOrderPlacedSuccessfully"
runat="server" Visible="false">
                                        <div style="text-align:center">
                                            <h1> HAPPY SHOPPING</h1> <!-- Put an image
here-->
                                            <label> Order Placed
Successfully</label><br /><br />
```

```
                                             Transaction Details are sent at the email
ID provided by you.<br /><br />
                                             <asp:Label ID="lblTransactionNo"
runat="server" Text="Label"></asp:Label>
                                             <br />
                                             <br />
                                             <br />
                                             <a href="TrackYourOrder.aspx"> Track Your
Order</a>
                                             <br />
                                             <br />
                                             <br />
                                         </div>
                                     </asp:Panel>
                                 </td>
                             </tr>
                             <tr>
                                 <td colspan="2" style="align-
content:center">&nbsp;&copy; BRP BOB@ oldbooksshoppingcart.com
                                     ||<a href="Admin/Login.aspx">ADMIN PANEL</a> ||
<a href="TrackYourOrder.aspx"> Track Your Order</a></td>
                             </tr>
                         </table>
                     </td>
                 </tr>

            </table>
        </ContentTemplate>
        </asp:UpdatePanel>
    </form>
</body>
</html>
```

TrackYourOrder.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="TrackYourOrder.aspx.cs"
Inherits="ShoppingCart.TrackYourOrder" %>

<%@ Register src="usCustomerOrder.ascx" tagname="usCustomerOrder" tagprefix="uc1" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
             <uc1:usCustomerOrder ID="usCustomerOrder1" runat="server" /> <!-- User
Control-->
         </div>
    </form>
</body>
</html>
```

usCustomerOrder.ascx

```
<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="usCustomerOrder.ascx.cs"
Inherits="ShoppingCart.usCustomerOrder" %>
<asp:ScriptManager ID="ScriptManager1" runat="server"></asp:ScriptManager>
<asp:UpdatePanel ID="UpdatePanel1" runat="server">
    <ContentTemplate>
        <div style="align-content:center">
            <asp:Label ID="lblTransactionNo" runat="server" Text="Transaction
No:"></asp:Label>
            <asp:TextBox ID="txtTransactionNo" runat="server"
Width="90px"></asp:TextBox>
            <asp:Button ID="btnGo" runat="server" Text="Go" OnClick="btnGo_Click"
/>&nbsp;
        </div>
        <table style="width: 100%;" align="center">
            <tr>
                <td align="center">
                    <hr />
                    <asp:RadioButtonList ID="rblProductDetails" runat="server"
RepeatDirection="Horizontal"
                        AutoPostBack="true"
OnSelectedIndexChanged="rblProductDetails_SelectedIndexChanged" >

                        <asp:ListItem Selected="True"
Value="1">CustomerDetails</asp:ListItem>
                        <asp:ListItem Value="2">ProductDetails</asp:ListItem>
                        <asp:ListItem Value="3">DeliveryStatus</asp:ListItem>

                    </asp:RadioButtonList>
                    <hr />
                </td>

            </tr>
            <tr>
                <td align="center">
                    <asp:Panel ID="Panel1" runat="server">
                        <table width="600px">
                            <tr>
                                <td align="center" colspan="2">
                                    <asp:Image ID="Image1" runat="server"
Height="150px" ImageUrl="~\Images\photo_emptyProfile.png" />
                                    <hr />
                                </td>

                            </tr>
                            <tr>
                                <td align="right" style="width:50%;padding-
right:30px">&nbsp;Name:
                                </td>
                                <td align="left" style="width:50%;">
                                    <asp:Label ID="lblCustomerName" runat="server">
                                    </asp:Label>
                                </td>

                            </tr>
                            <tr>
                                <td align="right" style="width:50%;padding-
right:30px">&nbsp;EmailID:
                                </td>
                                <td align="left" style="width:50%;">
```

```
                                    <asp:Label ID="lblCustomerEmailId" runat="server">
                                    </asp:Label>
                                </td>
                            </tr>
                            <tr>
                                <td align="right" style="width:50%;padding-
right:30px">&nbsp;PhoneNo:
                                </td>
                                <td align="left" style="width:50%;">
                                    <asp:Label ID="lblCustomerPhoneNo" runat="server">
                                    </asp:Label>
                                </td>
                            </tr>
                            <tr>
                                <td align="right" style="width:50%;padding-
right:30px">Total Products:
                                </td>
                                <td align="left" style="width:50%;">
                                    <asp:Label ID="lblTotalProducts" runat="server">
                                    </asp:Label>
                                </td>
                            </tr>
                            <tr>
                                <td align="right" style="width:50%;padding-
right:30px">Total Price:
                                </td>
                                <td align="left" style="width:50%;">
                                    <asp:Label ID="lblTotalPrice" runat="server">
                                    </asp:Label>
                                </td>
                            </tr>
                            <tr>
                                <td align="right" style="width:50%;padding-
right:30px">&nbsp;Address:
                                </td>
                                <td align="left" style="width:50%;">
                                    <asp:TextBox ID="txtCustomerAddress"
runat="server" Height="70px" ReadOnly="true" TextMode="MultiLine" Width="260px">

                                    </asp:TextBox>
                                </td>
                            </tr>
                            <tr>
                                <td align="right" style="width:50%;padding-
right:30px">&nbsp;Payment Method:
                                </td>
                                <td align="left" style="width:50%;">
                                    <asp:Label ID="lblPaymentMethod" runat="server">
                                    </asp:Label>
                                </td>
                            </tr>
                            <tr>
                                <td align="center" style="width:50%;">&nbsp;</td>
                                <td align="left" style="width:50%;">
                                    &nbsp;
                                </td>
                            </tr>
                            <tr>
                                <td align="center" style="width:50%;">&nbsp;&nbsp;
                                </td>
                                <td align="left" style="width:50%;">
                                    &nbsp;
                                </td>
                            </tr>
                        </table>
                    </asp:Panel>
                    <asp:Panel ID="Panel2" runat="server">
                        <table style="width: 600px" >
                            <tr>
                                <td align="center">
                                    <asp:Image ID="Image2" runat="server" 
ImageUrl="~\Images\girl_holding_travel_bag.jpg" height="150px"/><hr />
                                </td>
                            </tr>
                            <tr>
                                <td align="center">
                                    <asp:DataList ID="dlProducts" runat="server" 
RepeatColumns="3" width="500px" RepeatDirection="Horizontal" >
                                        <ItemTemplate>
                                            <div>
                                                <table style="width: 100%;">
                                                    <tr>
                                                        <td colspan="2">
                                                            <asp:Label 
ID="lblProductName" runat="server" Text='<%# Eval("Name") %>'></asp:Label>
                                                        </td>

                                                    </tr>
                                                    <tr>
                                                        <td colspan="2">
                                                            <img alt="" src="" 
runat="server" id="ImageProductPhoto" style="width:170px;height:160px"
                                                                ImageUrl='<%# 
Eval("ImageUrl") %>'/>
                                                        </td>

                                                    </tr>
                                                    <tr>
                                                        <td>Price:<asp:Label 
ID="lblPrice" runat="server" Text='<%# Eval("Price") %>'></asp:Label></td>
                                                        <td>Quantity:<asp:Label 
ID="lblQuantity" runat="server" Text='<%# Eval("ProductQuantity") 
%>'></asp:Label></td>

                                                    </tr>
                                                </table>
                                            </div>
                                        </ItemTemplate>
                                    </asp:DataList>
                                </td>

                            </tr>

                        </table>
                    </asp:Panel>
                    <asp:Panel ID="Panel3" runat="server">
                        <table style="width:600px">
                            <tr>
```

```
                                <td align="center">
                                    <asp:Image ID="Image3" runat="server"
Height="150px" ImageUrl="~/Images/delivery-boy.gif"/><!-- delivery status img-->
                                    <hr />
                                </td>

                            </tr>
                            <tr>
                                <td align="center">
                                    <asp:GridView ID="gvOrderStatus"
runat="server"></asp:GridView>
                                </td>

                            </tr>
                            <tr>
                                <td align="center">
                                    <asp:TextBox ID="txtStatus" runat="server"
Width="300px"></asp:TextBox>&nbsp;
                                    <asp:Button ID="btnAdd" runat="server" Text="Add"
OnClick="btnAdd_Click" />
                                </td>

                            </tr>
                        </table>
                    </asp:Panel>
                    <asp:Panel ID="Panel4" runat="server">
                        <table style="width: 100%;">
                            <tr>
                                <td>
                                    <h1>NO Results Found</h1>
                                </td>

                            </tr>

                        </table>
                    </asp:Panel>

                </td>

            </tr>

        </table>
    </ContentTemplate>
</asp:UpdatePanel>
```

Login.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Configuration;

namespace ShoppingCart.Admin
{
    public partial class Login : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            txtLoginId.Focus();
        }

        protected void btnLogin_Click(object sender, EventArgs e)
        {
            string LoginId = WebConfigurationManager.AppSettings["AdminLoginID"];
            string Password = WebConfigurationManager.AppSettings["AdminPassword"];
            if (txtLoginId.Text == LoginId && txtPassword.Text == Password)
            {
                Session["ShoppingCartAdmin"] = "ShoppingCartAdmin";
                Response.Redirect("~/Admin/AddNewProducts.aspx");

            }
            else
                lblAlert.Text = "INVALID LOGIN ID OR PASSWORD";
        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            Session["ShoppingCartAdmin"] = null;
            Response.Redirect("~/Default.aspx");
        }


    }
}
```

AddEditCategory.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using ShoppingCart.BusinessLayer;

namespace ShoppingCart.Admin
{
    public partial class AddEditCategory : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
            ShoppingCartclass k = new ShoppingCartclass
            {
                CategoryName=txtCategoryName.Text
            };
            k.AddNewCategory();
            txtCategoryName.Text = string.Empty;
            Response.Redirect("~/Admin/AddNewProducts.aspx");
        }
    }
}
```

AddNewProducts.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using ShoppingCart.BusinessLayer;
using System.Data;

namespace ShoppingCart.Admin
{
    public partial class AddNewProducts : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if(!IsPostBack) //to check if this page has been loaded for the first time
            {
                GetCategories();
                AddSubmitEvent();
                if(Request.QueryString["alert"]=="success")
                {
                    Response.Write("<script>alert('record Saved 
Successfully')</script>");
                }
            }
        }
        private void AddSubmitEvent()  //This function will allow only this page to 
refresh..Others will not refresh on postback
        {
            UpdatePanel updatePanel = Page.Master.FindControl("AdminUpdatePanel") as 
UpdatePanel;
            UpdatePanelControlTrigger trigger = new PostBackTrigger();
            trigger.ControlID = btnsubmit.UniqueID;
            updatePanel.Triggers.Add(trigger);
        }
        public void GetCategories()
        {
            ShoppingCartclass k = new ShoppingCartclass();
            DataTable dt = k.GetCategories();
            if(dt.Rows.Count>0)
            {
                ddlProductCategory.DataValueField = "CategoryID"; //for binding the 
data to the dropdown list all four lines
                ddlProductCategory.DataTextField = "CategoryName";
                ddlProductCategory.DataSource = dt;
                ddlProductCategory.DataBind();
            }

        }

        protected void btnsubmit_Click(object sender, EventArgs e)
        {
            if(uploadProductPhoto.PostedFile!=null)
            {
                SaveProductPhoto();
                ShoppingCartclass k = new ShoppingCartclass()
                {
                    ProductName = txtProductName.Text,
                    ProductImage = "~/ProductImages/" + uploadProductPhoto.FileName,
                    ProductPrice = txtProductPrice.Text,
                    ProductDescription = txtProductDescription.Text,
                    CategoryID = Convert.ToInt32(ddlProductCategory.SelectedValue),
                    TotalProducts=Convert.ToInt32(txtProductQuantity.Text)

                };
                k.AddNewProduct();
                //Alert.Show("REcord Saved Successfully");
                ClearText();
                Response.Redirect("~/Admin/AddNewProducts.aspx?alert=success");
            }
                else
                {
                    Response.Write("<script>alert('Please upload the 
photo');</script>");
                }

            }
            private void ClearText()
            {
                uploadProductPhoto = null;
                txtProductDescription.Text = null;
                txtProductName.Text = null;
                txtProductPrice.Text = null;
                ddlProductCategory.SelectedValue = null;
                txtProductQuantity.Text = null;
            }
            private void SaveProductPhoto()
            {
                if(uploadProductPhoto.PostedFile!=null)
                {
                    string filename = 
uploadProductPhoto.PostedFile.FileName.ToString();
                    string fileExt = 
System.IO.Path.GetExtension(uploadProductPhoto.FileName);

                    //check filename length
                    if(filename.Length>97)
                    {
                        //show("Size shouldn't exceed 97 characters");
                        return;
                    }
                    // check file type
                    else if(fileExt !=".jpeg" && fileExt !=".jpg" && fileExt !=".png" 
&& fileExt !=".bmp" )
                    {
                        //alert("Only the above formats are allowed");
                        return;
                    }
                    // check file size
                    else if(uploadProductPhoto.PostedFile.ContentLength>4000000)
                    {
                        //alert("the image size shouldn't exceed 4MB");
                        return;
                    }
                    else
                    {
                        uploadProductPhoto.SaveAs(Server.MapPath("~/ProductImages/" + 
filename));
```

```
                        }
                    }

                }



            }
        }
```

AdminMaster.Master.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace ShoppingCart.Admin
{
    public partial class AdminMaster : System.Web.UI.MasterPage
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["ShoppingCartAdmin"] == null)
                Response.Redirect("~/Admin/Login.aspx");
        }
    }
}
```

Category.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using ShoppingCart.BusinessLayer;
using System.Data;

namespace ShoppingCart.Admin
{
    public partial class Category : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if(!IsPostBack)//loaded for the first time
            {
                GetCategories();
            }
        }
        private void GetCategories()
        {
            ShoppingCartclass k = new ShoppingCartclass();
            DataTable dt = k.GetCategories();
            if(dt.Rows.Count>0)
            {
                gvAvailableCategories.DataSource = dt;
                gvAvailableCategories.DataBind();
            }
        }
    }
}
```

CustomerOrders.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using ShoppingCart.BusinessLayer;
using System.Data;
namespace ShoppingCart.Admin
{
    public partial class CustomerOrders : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if(!IsPostBack)
            {
                GetOrdersList();
            }
        }
        private void GetOrdersList()
        {
            ShoppingCartclass k = new ShoppingCartclass
            {
                Flag=0
            };
            DataTable dt = k.GetOrdersList();
            gvCustomerOrders.DataSource = dt;
            gvCustomerOrders.DataBind();
        }

        protected void gvCustomerOrders_SelectedIndexChanged(object sender, EventArgs
e)
        {

        }
    }
}
```

OrderDetails.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace ShoppingCart.Admin
{
    public partial class OrderDetails : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!string.IsNullOrEmpty(Request.QueryString["Id"]))
            {
                string TransactionNo = Request.QueryString["Id"];
                usCustomerOrder1.TransactionNoText = TransactionNo;
            }
            if (!string.IsNullOrEmpty(Convert.ToString(Session["ShoppingCartAdmin"])))
            {
                usCustomerOrder1.IsAuthorisedToAddStatus = true;
            }
            else
            {
                usCustomerOrder1.IsAuthorisedToAddStatus = false;
                Response.Redirect("~/Admin/Login.aspx");
            }
        }

        protected void btnClear_Click(object sender, EventArgs e)
        {
            Response.Redirect("~/Admin/Login.aspx");
        }
    }
}
```

Products.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using ShoppingCart.BusinessLayer;
using System.Data;

namespace ShoppingCart.Admin
{
    public partial class Products : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)//loaded for the first time
            {
                GetProducts(0);
            }
        }
        private void GetProducts(int CategoryID)
        {
            ShoppingCartclass k = new ShoppingCartclass() {
                CategoryID=CategoryID
            };
            gvAvailableProducts.DataSource = null;
            gvAvailableProducts.DataSource = k.GetAllProducts(); ;
            gvAvailableProducts.DataBind();


        }
    }
}
```

ProductStock.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using ShoppingCart.BusinessLayer;
using System.Data;


namespace ShoppingCart.Admin
{
    public partial class ProductStock : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if(!IsPostBack)
            {
                GetCategories();
                GetAvailbleStock();

            }
        }

        protected void ddlCategory_SelectedIndexChanged(object sender, EventArgs e)
        {
            GetAvailbleStock();
        }
        private void GetAvailbleStock()
        {
            ShoppingCartclass k = new ShoppingCartclass
            {
                CategoryID =Convert.ToInt32(ddlCategory.SelectedValue),
                StockType=Convert.ToInt32(rblProductStock.SelectedValue)
            };
            DataTable dt=k.GetAvailableStock();
            if(dt.Rows.Count>0)
            {
                gvAvailableStock.DataSource=dt;
                gvAvailableStock.DataBind();
                gvAvailableStock.Visible=true;
                NoRecordsToDisplay.Visible=false;
            }
            else
            {
                gvAvailableStock.Visible = false;
                NoRecordsToDisplay.Visible = true;
            }


        }

        protected void rblProductStock_SelectedIndexChanged(object sender, EventArgs
e)
        {
            GetAvailbleStock();

        }
        private void GetCategories()
        {
            ShoppingCartclass k = new ShoppingCartclass();
            DataTable dt = k.GetCategories();
            if(dt.Rows.Count>0)
            {
                ddlCategory.DataValueField = "CategoryID";
                ddlCategory.DataTextField = "CategoryName";
                ddlCategory.DataSource = dt;
                ddlCategory.DataBind();
                ddlCategory.Items.Add(new ListItem("All Products", "0", true));
                ddlCategory.SelectedValue = "0";

            }

        }
    }
}
```

Default.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using ShoppingCart.BusinessLayer;
using System.Configuration;
using System.Data.SqlClient;
using System.IO;

namespace ShoppingCart
{
    public partial class Default : System.Web.UI.Page
    {
        public int xl;
        public int ab;
        public int bc;
        protected void Page_Load(object sender, EventArgs e)
        {
            if(!IsPostBack)        // To ensure the data is binded only once to home 
page
            {
                lblCategoryName.Text = "Popular Products On Shopping Cart Website";
                GetProducts(0); //To get all products
                GetCategory();

            }
            lblAvailableStockAlert.Text = string.Empty;
        }
        private void GetCategory()
        {
            ShoppingCartclass k = new ShoppingCartclass();
            dlCategories.DataSource = null;
            dlCategories.DataSource = k.GetCategories();
            dlCategories.DataBind();
        }
        private void GetProducts(int CategoryID)
        {
            ShoppingCartclass k = new ShoppingCartclass()
            {
                CategoryID=CategoryID
            };
            dlProducts.DataSource=null;
            dlProducts.DataSource = k.GetAllProducts();
            dlProducts.DataBind();
        }
        protected void lblLogo_Click(object sender, EventArgs e)
        {
            lblCategoryName.Text = "Popular Products at Shopping Cart";
            lblProducts.Text = "Products";
            pnlCategories.Visible = true;
            pnlProducts.Visible = true;
            pnlCheckOut.Visible = false;
            pnlEmptyCart.Visible = false;
            pnlMyCart.Visible = false;
            pnlOrderPlacedSuccessfully.Visible = false;
            GetProducts(0);
            HighlightCartProducts();
        }

        protected void btnShoppingCart_Click(object sender, EventArgs e)
        {
            try
            {
                GetMyCart();
                lblCategoryName.Text = "Products in my Shopping Cart";
                lblProducts.Text = "CheckOut Form";
            }
            catch(Exception e1)
            {
                Response.Write("<script>alert('"+e1.StackTrace+"')</script>");
            }
        }

        protected void btnAddToCart_Click(object sender, EventArgs e)
        {
            string ProductID = 
Convert.ToInt16((((Button)sender).CommandArgument)).ToString();
            string ProductQuantity = "1";

            DataListItem currentItem = (sender as Button).NamingContainer as 
DataListItem;
            Label lblAvilableStock = currentItem.FindControl("lblAvailableStock") as 
Label;
            if(Session["MyCart"] !=null)
            {
                DataTable dt = (DataTable)Session["MyCart"];//Coverting Session from 
else part into DataTable
                var checkProduct = dt.AsEnumerable().Where(r => 
r.Field<string>("ProductID") == ProductID);
                if(checkProduct.Count()==0)
                {
                    string query = "Select * from products where ProductID=" + 
ProductID + ";";
                    DataTable dtProducts = GetData(query);
                    DataRow dr = dt.NewRow();
                    dr["ProductID"] = ProductID;
                    dr["Name"] = Convert.ToString(dtProducts.Rows[0]["Name"]);
                    dr["Description"] = 
Convert.ToString(dtProducts.Rows[0]["Description"]);
                    dr["Price"] = Convert.ToString(dtProducts.Rows[0]["Price"]);
                    dr["ImageUrl"] = Convert.ToString(dtProducts.Rows[0]["ImageUrl"]);
                    dr["ProductQuantity"] = ProductQuantity;
                    dr["AvailableStock"] = lblAvilableStock.Text;
                    dt.Rows.Add(dr);
                    Session["MyCart"] = dt;
                    btnShoppingCart.Text = dt.Rows.Count.ToString();
                }

            }
            else
            {
                string query="Select * from products where ProductID="+ ProductID+"";
                DataTable dtProducts=GetData(query);
                DataTable dt = new DataTable();
                dt.Columns.Add("ProductID",typeof(string));
                dt.Columns.Add("Name",typeof(string));
                dt.Columns.Add("Description",typeof(string));
                dt.Columns.Add("Price",typeof(string));
                dt.Columns.Add("ImageUrl",typeof(string));
                dt.Columns.Add("ProductQuantity",typeof(string));
                dt.Columns.Add("AvailableStock",typeof(string));

                DataRow dr=dt.NewRow();
                dr["ProductID"] = ProductID;
                dr["Name"]=Convert.ToString(dtProducts.Rows[0]["Name"]);
                dr["Description"]=Convert.ToString(dtProducts.Rows[0]["Description"]);
                dr["Price"]=Convert.ToString(dtProducts.Rows[0]["Price"]);
                dr["ImageUrl"]=Convert.ToString(dtProducts.Rows[0]["ImageUrl"]);
                dr["ProductQuantity"]=ProductQuantity;
                dr["AvailableStock"]=lblAvilableStock.Text;

                dt.Rows.Add(dr);
                Session["MyCart"] = dt;
                btnShoppingCart.Text = dt.Rows.Count.ToString(); //No of Rows Count is
equal to the number of products added to the cart
            }
            HighlightCartProducts();

        }
        private void HighlightCartProducts()
        {
            if(Session["MyCart"]!=null)
            {
                DataTable dtProductsAddedToCart = (DataTable)Session["MyCart"];
                if(dtProductsAddedToCart.Rows.Count>0)
                {
                    foreach(DataListItem item in dlProducts.Items)
                    {
                        HiddenField hfProductID = item.FindControl("hfProductID") as
HiddenField;
                        if (dtProductsAddedToCart.AsEnumerable().Any(row =>
hfProductID.Value == row.Field<String>("ProductID")))
                        {
                            Button btnAddToCart = item.FindControl("btnAddToCart") as
Button;
                            btnAddToCart.BackColor = System.Drawing.Color.Green;
                            btnAddToCart.Text = "Added To Cart";
                        }
                    }
                }
            }
        }
        protected void dlCategories_SelectedIndexChanged(object sender, EventArgs e)
        {

        }

        protected void lbtnCategory_Click(object sender, EventArgs e)//to get the
products specific to that category
        {
            pnlMyCart.Visible = false;
            pnlProducts.Visible = true;
            int CategoryID = Convert.ToInt16((((LinkButton)sender).CommandArgument));
            GetProducts(CategoryID);
            HighlightCartProducts();
        }

        protected void txtProductQuantity_TextChanged(object sender, EventArgs
e)//changing the product quantity
        {
            TextBox txtQuantity = (sender as TextBox);
            DataListItem currentItem = (sender as TextBox).NamingContainer as
DataListItem;
            HiddenField ProductID = currentItem.FindControl("hfProductID") as
HiddenField;
            Label lblAvailableStock=currentItem.FindControl("lblAvailbleStock") as
Label;
            if(txtQuantity.Text==string.Empty ||txtQuantity.Text=="0" ||
txtQuantity.Text=="1")
            {
                txtQuantity.Text = "1";
                DataTable dt = (DataTable)Session["MyCart"];
            }

            else
            {

                    if (Session["MyCart"] != null)
                    {
                        int ab, bc;
                        if (Int32.TryParse(txtQuantity.Text, out ab) &&
Int32.TryParse(lblAvailableStock.Text, out bc))
                        {
                            if (ab <= bc)
                            {
                                DataTable dt = (DataTable)Session["MyCart"];
                                DataRow[] rows = dt.Select("ProductID'" +
ProductID.Value + "'");

                                int index = dt.Rows.IndexOf(rows[0]);
                                dt.Rows[index]["ProductQuantity"] = txtQuantity.Text;
                                Session["MyCart"] = dt;

                            }
                            else
                            {
                                lblAvailableStockAlert.Text = "Alert:Product BuyOut
Should not be more than available Stock";
                                txtQuantity.Text = "1";

                            }
                        }
                    }

            }
            UpdateTotalBill();
        }

        protected void btnPlaceOrder_Click(object sender, EventArgs e)
        {
            string productids = string.Empty;
            DataTable dt;
            if(Session["MyCart"]!=null)
            {
                dt = (DataTable)Session["MyCart"];
                ShoppingCartclass k = new ShoppingCartclass()
                {
                    CustomerName=txtCustomerName.Text,
                    CustomerAddress=txtCustomerAddress.Text,
                    CustomerPhoneNo=txtCustomerPhoneNo.Text,
                    CustomerEmailID=txtCustomerEmailID.Text,
                    TotalProducts=Convert.ToInt32(txtTotalProducts.Text),
                    TotalPrice=Convert.ToInt32(txtTotalPrice.Text),
                    ProductList=productids,
                    PaymentMethod=rblPaymentMethod.SelectedItem.Text,
                };
                DataTable dtResult= k.SaveCustomerDetails();
                while(dt.Rows.Count>0)
                {
                    int i = 0;
                    ShoppingCartclass SaveProducts = new ShoppingCartclass()
                    {
                        CustomerID = Convert.ToInt32(dtResult.Rows[0][0]),
                        ProductID=Convert.ToInt32(dt.Rows[i]["ProductID"]),
                        TotalProducts=Convert.ToInt32(dt.Rows[i]["ProductQuantity"]),
                    };
                    SaveProducts.SaveCustomerProducts();
                }
                    Session.Clear();
                    GetMyCart();
                    lblTransactionNo.Text = "Your Transaction No is:" +
dtResult.Rows[0][0];
                    pnlProducts.Visible = false;
                    pnlMyCart.Visible = false;
                    pnlMyCart.Visible = false;
                    pnlEmptyCart.Visible = false;
                    pnlCheckOut.Visible = false;
                    pnlCategories.Visible = false;
                    pnlOrderPlacedSuccessfully.Visible = true;


                    SendOrderPlacedAlert(txtCustomerName.Text,
txtCustomerEmailID.Text, Convert.ToString(dtResult.Rows[0][0]));

                    txtCustomerAddress.Text = String.Empty;
                    txtCustomerEmailID.Text = String.Empty;
                    txtCustomerName.Text = String.Empty;
                    txtCustomerPhoneNo.Text = String.Empty;
                    txtTotalPrice.Text = "0";
                    txtTotalProducts.Text = "0";
```

```
    }

}
private void GetMyCart() //showing data in empty cart and mycart
{
    try
    {
        DataTable dtProducts=new DataTable();
        if (Session["MyCart"] != null)
        {
            dtProducts = (DataTable)Session["MyCart"];
        }
        else
        {
            dtProducts = new DataTable();
        }
        if (dtProducts.Rows.Count > 0)
        {

            txtTotalProducts.Text = dtProducts.Rows.Count.ToString();
            btnShoppingCart.Text = dtProducts.Rows.Count.ToString();
            dlCartProducts.DataSource = dtProducts;
            dlCartProducts.DataBind();
            UpdateTotalBill();

            pnlMyCart.Visible = true;
            pnlCheckOut.Visible = true;
            pnlEmptyCart.Visible = false;
            pnlOrderPlacedSuccessfully.Visible = false;
            pnlProducts.Visible = false;
            pnlCategories.Visible = false;

        }
        else
        {
            //lblCategoryName.Text = "";
            //lblProducts.Text = "";
            pnlEmptyCart.Visible = true;
            pnlMyCart.Visible = false;
            pnlCheckOut.Visible = false;
            pnlOrderPlacedSuccessfully.Visible = false;
            pnlProducts.Visible = false;
            pnlCategories.Visible = false;

            dlCartProducts.DataSource = null;
            dlCartProducts.DataBind();
            txtTotalProducts.Text = "0";
            txtTotalPrice.Text = "0";
            btnShoppingCart.Text = "0";
        }
    }
    catch (Exception e1)
    {
        Response.Write("<script>alert('" + e1.StackTrace + "')</script>");
    }

}
```

```
        private void UpdateTotalBill()
        {
            try
            {
                long TotalPrice = 0;
                long TotalProducts = 0;
                foreach (DataListItem item in dlCartProducts.Items)
                {
                    Label PriceLabel = item.FindControl("lblPrice") as Label;
                    TextBox ProductQuantity = item.FindControl("txtProductQuantity")
as TextBox;
                    long x = Convert.ToInt64(PriceLabel.Text);
                    long y = Convert.ToInt64(ProductQuantity.Text);
                    long ProductPrice = x * y;
                    TotalPrice = TotalPrice + ProductPrice;
                    TotalProducts = TotalProducts +
Convert.ToInt32(ProductQuantity.Text);
                }
                txtTotalPrice.Text = Convert.ToString(TotalPrice);
                txtTotalProducts.Text = Convert.ToString(TotalProducts);
            }
            catch (Exception e1)
            {

                Response.Write("<script>alert('" + e1.StackTrace + "')</script>");
            }

        }

       protected void btnRemoveFromCart_Click(object sender, EventArgs e)
       {
           string ProductID =
Convert.ToInt16(((Button)sender).CommandArgument).ToString();
           if(Session["MyCart"]!=null)
           {
               DataTable dt = (DataTable)Session["MyCart"];
               DataRow drr = dt.Select("ProductID=" + ProductID +
"").FirstOrDefault();
               if(drr!=null)
               {
                   dt.Rows.Remove(drr);
               }
               Session["MyCart"] = dt;
           }
           GetMyCart();
       }

       public DataTable GetData(string query)
       {
           DataTable dt = new DataTable();
           String Conn =
ConfigurationManager.ConnectionStrings["MyCon"].ConnectionString;
           SqlConnection con = new SqlConnection(Conn);
           con.Open();
           SqlDataAdapter da = new SqlDataAdapter(query, con);
           da.Fill(dt);
           con.Close();
           return dt;
       }
       private string PopulateOrderEmailBody(string CustomerName, string
TransactionNo)
       {
           string body = string.Empty;
           using (StreamReader reader = new
StreamReader(Server.MapPath("~/OrderTemplate.html")))
           {
               body = reader.ReadToEnd();
           }
           body = body.Replace("{CustomerName}", CustomerName);
           body = body.Replace("{OrderNo}", TransactionNo);
           body = body.Replace("{TransactionUrl}",
"http://OldBooksShoppingCart.in/TrackYourOrder.aspx?Id="+ TransactionNo);
           return body;
       }

       private void SendOrderPlacedAlert(string CustomerName,string
CustomerEmailID,string TransactionNo)
       {
           string body = this.PopulateOrderEmailBody(CustomerName, TransactionNo);
           EmailEngine.SendEmail(CustomerEmailID, "Shopping Cart----Your Email
Details", body);
       }

       protected void dlProducts_SelectedIndexChanged(object sender, EventArgs e)
       {

       }

       protected void dlCartProducts_SelectedIndexChanged(object sender, EventArgs e)
       {

       }
    }
}
```

TrackYourOrder.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace ShoppingCart
{
    public partial class TrackYourOrder : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!string.IsNullOrEmpty(Request.QueryString["Id"]))
            {
                string TransactionNo = Request.QueryString["Id"];
                usCustomerOrder1.TransactionNoText = TransactionNo;
            }
            if (!string.IsNullOrEmpty(Convert.ToString(Session["ShoppingCartAdmin"])))
            {
                usCustomerOrder1.IsAuthorisedToAddStatus = true;
            }
            else
```

```
                usCustomerOrder1.IsAuthorisedToAddStatus = false;
        }
    }
}
```

usCustomerOrder.ascx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using ShoppingCart.BusinessLayer;
using System.Data;

namespace ShoppingCart
{
    public partial class usCustomerOrder : System.Web.UI.UserControl
    {
        public bool CanIUpdateStatus;
        public string TransactionNoText
        {
            get { return txtTransactionNo.Text; }
            set { txtTransactionNo.Text = value; }
        }
        public bool IsAuthorisedToAddStatus
        {
            set{CanIUpdateStatus=value;}
        }
        protected void Page_Load(object sender, EventArgs e)
        {
            if(!IsPostBack)
            {
                if (txtTransactionNo.Text != string.Empty)
                {
                    ShowOrderDetails(rblProductDetails.SelectedValue,
Convert.ToInt32(txtTransactionNo.Text));
                }
                else
                {
                    rblProductDetails.Visible = false;
                    Panel1.Visible = false;
                    Panel2.Visible = false;
                    Panel3.Visible = false;
                    Panel4.Visible = false;
                }
            }

        }
        private void ShowOrderDetails(string PanelId, int OrderNo)
        {
            Panel1.Visible = false;
            Panel2.Visible = false;
            Panel3.Visible = false;
            Panel4.Visible = false;
            rblProductDetails.Visible = false;
            if (IsOrderNoValid(OrderNo))
            {
                rblProductDetails.Visible = true;
                if (PanelId == "1")
                {
                    ShoppingCartclass k = new ShoppingCartclass
                    {
                        Flag = OrderNo
                    };
                    DataTable dtCustomerDetails = k.GetOrdersList();
                    if (dtCustomerDetails.Rows.Count > 0)
                    {
                        Panel1.Visible = true;
                        lblCustomerName.Text = 
Convert.ToString(dtCustomerDetails.Rows[0]["CustomerName"]);
                        lblCustomerPhoneNo.Text = 
Convert.ToString(dtCustomerDetails.Rows[0]["CustomerPhoneNo"]);
                        lblCustomerEmailId.Text = 
Convert.ToString(dtCustomerDetails.Rows[0]["CustomerEmailId"]);
                        lblPaymentMethod.Text = 
Convert.ToString(dtCustomerDetails.Rows[0]["PaymentMethod"]);
                        lblTotalPrice.Text = 
Convert.ToString(dtCustomerDetails.Rows[0]["TotalPrice"]);
                        lblTotalProducts.Text = 
Convert.ToString(dtCustomerDetails.Rows[0]["TotalProducts"]);
                        txtCustomerAddress.Text = 
Convert.ToString(dtCustomerDetails.Rows[0]["CustomerAddress"]);

                    }
                }
                if (PanelId == "2")
                {
                    Panel2.Visible = true;
                    ShoppingCartclass k = new ShoppingCartclass
                    {
                        Flag = OrderNo
                    };
                    dlProducts.DataSource = k.GetTransactionDetails();
                    dlProducts.DataBind();
                }
                if (PanelId == "3")
                {
                    Panel3.Visible = true;
                    txtStatus.Visible = CanIUpdateStatus;
                    btnAdd.Visible = CanIUpdateStatus;
                    GetSetOrderStatus(0);
                }
            }
            else
                Panel4.Visible = true;
        }
        private void GetSetOrderStatus(int Flag)
        {
            ShoppingCartclass k = new ShoppingCartclass
            {
                OrderStatus=txtStatus.Text,
                OrderNo=txtTransactionNo.Text,
                Flag = Flag
            };
            DataTable dt = k.GetSetOrderStatus();
            gvOrderStatus.DataSource = dt;
            gvOrderStatus.DataBind();
            txtStatus.Text = string.Empty;
```

```
        }
        private bool IsOrderNoValid(int OrderNo)
        {
            ShoppingCartclass k = new ShoppingCartclass
            {
                Flag = OrderNo
            };
            DataTable dtCustomerDetails = k.GetOrdersList();
            if (dtCustomerDetails.Rows.Count > 0)
                return true;
            else
                return false;
        }
        protected void btnGo_Click(object sender, EventArgs e)
        {
            if(txtTransactionNo.Text!=string .Empty)
            {
                rblProductDetails.Visible = true;
                ShowOrderDetails(rblProductDetails.SelectedValue,
Convert.ToInt32(txtTransactionNo.Text));
            }
            else
            {
                rblProductDetails.Visible = false;
                Panel1.Visible = false;
                Panel2.Visible = false;
                Panel3.Visible = false;
                Panel4.Visible = false;
            }
        }

        protected void rblProductDetails_SelectedIndexChanged(object sender, EventArgs
e)
        {
            if (txtTransactionNo.Text != string.Empty)
            {
                rblProductDetails.Visible = true;
                ShowOrderDetails(rblProductDetails.SelectedValue,
Convert.ToInt32(txtTransactionNo.Text.Trim()));
            }
        }

        protected void btnAdd_Click(object sender, EventArgs e)
        {
            GetSetOrderStatus(1);
        }
    }
}
```

OrderTemplate.html

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title></title>
</head>
<body>
    <label style="font-size:30px">IGIT Shopping Cart</label>
    <hr />
    <span>Hello <b>{CustomerName}<b></b>,
        <br />
        You Have Recieved Your Order.<br /><br />
        Your Order No is: <b>{OrderNo}</b><br /><br />
        To Check Your Order Status, Click on the below link<br /><br />
        <a href="{TransactionUrl}">Click Here to check your order status</a>
        <br />
        <br />
        Thanks<br />
        IGIT Shopping Cart Team
    </span>
</body>
</html>
```

EmailEngine.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web;
using System.Net.Mail;
using System.Configuration;

namespace ShoppingCart
{
    public static class EmailEngine
    {
        public static void SendEmail(string recepientEmail,string subject, string
body)
        {
            using(MailMessage mailMessage=new MailMessage())
            {
                mailMessage.From = new
MailAddress(ConfigurationManager.AppSettings["UserName"]);
                mailMessage.Subject = subject;
                mailMessage.Body = body;
                mailMessage.IsBodyHtml = true;
                mailMessage.To.Add(new MailAddress(recepientEmail));
                SmtpClient smtp = new SmtpClient();
                smtp.Host = ConfigurationManager.AppSettings["Host"];
                System.Net.NetworkCredential NetWorkCred = new
System.Net.NetworkCredential();
                NetWorkCred.UserName = ConfigurationManager.AppSettings["UserName"];
                NetWorkCred.Password = ConfigurationManager.AppSettings["Password"];
                smtp.UseDefaultCredentials = true;
                smtp.Credentials = NetWorkCred;
                smtp.Port = int.Parse(ConfigurationManager.AppSettings["Port"]);
                smtp.Send(mailMessage);
            }
        }
    }
}
```

ShoppingCartClass.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data.SqlClient;
using System.Data;
```

```
namespace ShoppingCart.BusinessLayer
{
    public class ShoppingCartclass
    {
        public string CategoryName;
        public int CategoryID;
        public int StockType;
        public string ProductName;
        public string ProductImage;
        public string ProductPrice;
        public string ProductDescription;
        public int TotalProducts;
        public int TotalPrice;
        public string CustomerName;
        public string CustomerEmailID;
        public string CustomerAddress;
        public string CustomerPhoneNo;
        public string ProductList;
        public string PaymentMethod;
        public int CustomerID;
        public string OrderStatus;
        public string OrderNo;
        public int ProductID;
        public int Flag;
        public void AddNewCategory()
        {
            SqlParameter[] parameters=new SqlParameter[1];
            parameters[0] =
DataLayer.DataAcess.AddParameter("@CategoryName",CategoryName,System.Data.SqlDbType.Va
rChar,200);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_AddNewCategory", parameters);
         }
        public void AddNewProduct()
        {
            SqlParameter[] parameters = new SqlParameter[6];
            parameters[0] = DataLayer.DataAcess.AddParameter("@ProductName",
ProductName, System.Data.SqlDbType.VarChar, 300);
            parameters[1] = DataLayer.DataAcess.AddParameter("@ProductPrice",
ProductPrice, System.Data.SqlDbType.Int, 100);
            parameters[2] = DataLayer.DataAcess.AddParameter("@ProductImage",
ProductImage, System.Data.SqlDbType.VarChar, 500);
            parameters[3] = DataLayer.DataAcess.AddParameter("@ProductDescription",
ProductDescription, System.Data.SqlDbType.VarChar, 1000);
            parameters[4] = DataLayer.DataAcess.AddParameter("@CategoryID",
CategoryID, System.Data.SqlDbType.Int, 100);
            parameters[5] =
DataLayer.DataAcess.AddParameter("@ProductQuantity",TotalProducts,System.Data.SqlDbTyp
e.Int,100);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_AddNewProduct", parameters);
        }
        public DataTable GetCategories()
        {
            SqlParameter[] parameters = new SqlParameter[0];
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_GetAllCategories", parameters);
            return dt;
        }
        public DataTable GetAllProducts()
        {
            SqlParameter[] parameters = new SqlParameter[1];
            parameters[0] = DataLayer.DataAcess.AddParameter("@CategoryID",CategoryID,
System.Data.SqlDbType.Int, 20);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_GetAllProducts", parameters);
            return dt;
        }
        internal DataTable GetAvailableStock()
        {
            SqlParameter[] parameters = new SqlParameter[2];
            parameters[0] = DataLayer.DataAcess.AddParameter("@StockType", StockType,
System.Data.SqlDbType.Int, 10);
            parameters[1] = DataLayer.DataAcess.AddParameter("@CategoryID",
CategoryID, System.Data.SqlDbType.Int, 10);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_GetAvailableStock", parameters);
            return dt;
        }
        internal DataTable SaveCustomerDetails()
        {
            SqlParameter[] parameters = new SqlParameter[7];
            parameters[0] = DataLayer.DataAcess.AddParameter("@CustomerName",
CustomerName, System.Data.SqlDbType.VarChar, 100);
            parameters[1] = DataLayer.DataAcess.AddParameter("@CustomerEmailID",
CustomerEmailID, System.Data.SqlDbType.VarChar, 100);
            parameters[2] = DataLayer.DataAcess.AddParameter("@CustomerPhoneNo",
CustomerPhoneNo, System.Data.SqlDbType.VarChar, 10);
            parameters[3] = DataLayer.DataAcess.AddParameter("@CustomerAddress",
CustomerAddress, System.Data.SqlDbType.VarChar, 500);
            parameters[4] =
DataLayer.DataAcess.AddParameter("@TotalProducts",TotalProducts,
System.Data.SqlDbType.Int, 100);
            parameters[5] =
DataLayer.DataAcess.AddParameter("@TotalPrice",TotalPrice,System.Data.SqlDbType.Int,10
0);
            parameters[6] = DataLayer.DataAcess.AddParameter("@PaymentMethod",
PaymentMethod, System.Data.SqlDbType.VarChar, 100);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_SaveCustomerDetails", parameters);
            return dt;
        }
        internal void SaveCustomerProducts()
        {
            SqlParameter[] parameters = new SqlParameter[3];
            parameters[0] = DataLayer.DataAcess.AddParameter("@CustomerID",
CustomerID, System.Data.SqlDbType.Int, 20);
            parameters[1] = DataLayer.DataAcess.AddParameter("@ProductID", ProductID,
System.Data.SqlDbType.Int, 20);
            parameters[2] =
DataLayer.DataAcess.AddParameter("@TotalProduct",TotalProducts,
System.Data.SqlDbType.Int, 10);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_SaveCustomerProducts", parameters);
            return;
        }
        internal DataTable GetOrdersList()
        {
            SqlParameter[] parameters = new SqlParameter[1];
            parameters[0] = DataLayer.DataAcess.AddParameter("@Flag", Flag,
System.Data.SqlDbType.Int, 20);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_GetOrdersList", parameters);
            return dt;
        }
        internal DataTable GetTransactionDetails()
        {
            SqlParameter[] parameters = new SqlParameter[1];
            parameters[0] = DataLayer.DataAcess.AddParameter("@TransactionNo", Flag,
System.Data.SqlDbType.Int, 20);
            DataTable dt =
DataLayer.DataAcess.ExecuteDTbyProcedure("SP_GetTransactionDetails", parameters);
            return dt;
        }
        internal DataTable GetSetOrderStatus()
        {
            SqlParameter[] parameters = new SqlParameter[3];
            parameters[0] = DataLayer.DataAcess.AddParameter("@TransactionNo",
Convert.ToInt32(OrderNo), System.Data.SqlDbType.Int, 20);
            parameters[1] = DataLayer.DataAcess.AddParameter("@OrderStatus",
OrderStatus, System.Data.SqlDbType.VarChar, 300);
            parameters[2] = DataLayer.DataAcess.AddParameter("@Flag", Flag,
System.Data.SqlDbType.Int, 10);
            DataTable dt = DataLayer.DataAcess.ExecuteDTbyProcedure("SP_OrderStatus",
parameters);
            return dt;
        }
    }
}
```

DataAccess.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Configuration;
using System.Data.SqlClient;
using System.Data;

namespace ShoppingCart.DataLayer
{
    public class DataAcess
    {
        public static string ConnectionString
        {
            get
            {
                return
ConfigurationManager.ConnectionStrings["MyCon"].ConnectionString.ToString();
            }
        }
        public static SqlParameter AddParameter(String parameterName, object value,
SqlDbType DbType, int size)
        {
            SqlParameter param = new SqlParameter();
            param.ParameterName = parameterName;
            param.SqlDbType = DbType;
            param.Size = size;
            param.Direction = ParameterDirection.Input;
            param.Value = value;
            return param;
        }
        public static DataTable ExecuteDTbyProcedure(string ProcedureName,
SqlParameter[] Params)
        {
            SqlConnection conn = new SqlConnection(ConnectionString);
            SqlCommand cmd = new SqlCommand();
            cmd.Connection = conn;
            cmd.CommandText = ProcedureName;
            cmd.Parameters.AddRange(Params);
            cmd.CommandType = CommandType.StoredProcedure;
            SqlDataAdapter adopter = new SqlDataAdapter(cmd);
            DataTable dTable = new DataTable();
            try
            {
                adopter.Fill(dTable);

            }
            catch(Exception e)
            {
                Console.WriteLine(e.Message);
            }
            finally
            {
                //Disposing objects
                adopter.Dispose();
                cmd.Parameters.Clear();
                cmd.Dispose();
                conn.Dispose();
            }
            return dTable;
        }
    }
}
```

Web.config

```
<?xml version="1.0" encoding="utf-8"?>
<!--
  For more information on how to configure your ASP.NET application, please visit
  http://go.microsoft.com/fwlink/?LinkId=169433
  -->
<configuration>
  <system.web>
    <compilation debug="true" targetFramework="4.0" />
    <!--<httpRuntime targetFramework="4.5" />-->
  </system.web>
  <appSettings>
    <add key="AdminLoginID" value="admin"/>
    <add key="AdminPassword" value="123"/>
    <add key="Host" value="smtp.gmail.com"/>
    <add key="EnableSsl" value="true"/>
    <add key="UserName" value="brpbob26@gmail.com"/>
    <add key="Password" value="10121994b@b"/>
    <add key="Port" value="587"/>

  </appSettings>
  <system.web>
```

```
    <trace enabled="true"/>
  </system.web>
  <connectionStrings>
    <add name="MyCon" connectionString="Data Source=UNDERTAKER;Initial
Catalog=ShoppingCartDB;Integrated Security=True" providerName="System.Data.SqlClient"
/>
  </connectionStrings>
</configuration>
```

6.2 STORED PROCEDURE

SP_AddNewCategory

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_AddNewCategory]    Script Date: 06-05-2016
01:46:39 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- Batch submitted through debugger: AddNewCategory.sql|0|0|G:\Asp.net
Hub\Project\OldBooks ShoppingCart\ShoppingCart\ShoppingCart\AddNewCategory.sql
ALTER PROCEDURE [dbo].[SP_AddNewCategory]
(
       -- Add the parameters for the stored procedure here
       @CategoryName varchar(200)
)
AS
BEGIN
       -- SET NOCOUNT ON added to prevent extra result sets from
       -- interfering with SELECT statements.
       BEGIN TRY
              insert into Category values(@CategoryName)
       END TRY
       BEGIN CATCH
              PRINT(' ERROR OCCURED')
       END CATCH


END
```

SP_GetAllCategories

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_GetAllCategories]    Script Date: 06-05-
2016 01:48:07 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [dbo].[SP_GetAllCategories]

AS
BEGIN
       BEGIN TRY
```

```
              select * from Category
       END TRY
       BEGIN CATCH
              PRINT('ERROR OCCURED')
       END CATCH
END
```

SP_GetAllProducts

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_GetAllProducts]    Script Date: 06-05-2016
01:48:56 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- ============================================
ALTER PROCEDURE [dbo].[SP_GetAllProducts] (@CategoryID INT)
AS
       BEGIN
              BEGIN TRY
              IF(@CategoryID <> 0)
              BEGIN
                     select * from
                     (select P.CategoryID,
                                   P.ProductID,
                                   P.Name,
                                   P.Price,
                                   P.ImageUrl,
                                   C.CategoryName,
                                   P.ProductQuantity,
                                   IsNull(Sum(CP.TotalProduct),0)
as ProductSold,
                                   (P.ProductQuantity -
IsNull(Sum(CP.TotalProduct),0)) as AvailableStock
                            from Products P
                                   inner join Category C
                                                       on C.CategoryID= P.CategoryID
                                   left join CustomerProducts CP
                                                       on CP.ProductID =P.ProductID
                                   group by P.ProductID,
                                                 P.Name,
                                                 P.Price,
                                              P.ImageUrl,
                                                 C.CategoryName,
                                                 P.ProductQuantity,
                                                 P.CategoryID) StockTable
                                   where AvailableStock>0
                                          and CategoryID=@CategoryID
                     END
              ELSE
              BEGIN
```

```
                     select * from
                     (select P.CategoryID,
                                   P.ProductID,
                                   P.Name,
                                   P.Price,
                                   P.ImageUrl,
                                   C.CategoryName,
                                   P.ProductQuantity,
                                   IsNull(Sum(CP.TotalProduct),0)
as ProductSold,
                                   (P.ProductQuantity -
IsNull(Sum(CP.TotalProduct),0)) as AvailableStock
                            from Products P
                                   inner join Category C
                                                       on C.CategoryID= P.CategoryID
                                   left join CustomerProducts CP
                                                       on CP.ProductID =P.ProductID
                                   group by P.ProductID,
                                                  P.Name,
                                                  P.Price,
                                               P.ImageUrl,
                                                  C.CategoryName,
                                                  P.ProductQuantity,
                                                  P.CategoryID) StockTable
                                   where AvailableStock>0


                     END
              END TRY

                            BEGIN CATCH
                                   PRINT('ERROR OCCUR')
                            END CATCH
       END
```

SP_GetAvailableStock

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_GetAvailableStock]    Script Date: 06-05-
2016 01:49:37 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- Batch submitted through debugger: AddNewCategory.sql|0|0|G:\Asp.net
Hub\Project\OldBooks ShoppingCart\ShoppingCart\ShoppingCart\AddNewCategory.sql
ALTER PROCEDURE [dbo].[SP_GetAvailableStock](@StockType int,@CategoryID int)
AS
BEGIN
       BEGIN TRY
              declare @StartRange int
              declare @EndRange int

              if(@StockType =0)
                     begin
```

```
                            set @StartRange=0
                            set @EndRange=0
                     end
              if(@StockType =1)
                     begin
                            set @StartRange=1
                            set @EndRange=10
                     end
              if(@StockType =2)
                     begin
                            set @StartRange=11
                            set @EndRange=10000
                     end
              IF(@CategoryID <> 0)
              BEGIN
                     select * from
                     (select P.CategoryID,
                                   P.ProductID,
                                   P.Name,
                                   P.Price,
                                   P.ImageUrl,
                                   C.CategoryName,
                                   P.ProductQuantity,
                                   IsNull(Sum(CP.TotalProduct),0)
as ProductSold,
                                   (P.ProductQuantity -
IsNull(Sum(CP.TotalProduct),0)) as AvailableStock
                            from Products P
                                   inner join Category C
                                                       on C.CategoryID= P.CategoryID
                                   left join CustomerProducts CP
                                                       on CP.ProductID =P.ProductID
                                   where C.CategoryID=@CategoryID
                                   group by P.ProductID,
                                                 P.Name,
                                                 P.Price,
                                              P.ImageUrl,
                                                 C.CategoryName,
                                                 P.ProductQuantity,
                                                 P.CategoryID) StockTable
                                   where AvailableStock between @StartRange and
@EndRange

                     END
              ELSE
              BEGIN

                     select * from
                     (select P.CategoryID,
                                   P.ProductID,
                                   P.Name,
                                   P.Price,
                                   P.ImageUrl,
                                   C.CategoryName,
                                   P.ProductQuantity,
                                   IsNull(Sum(CP.TotalProduct),0)
as ProductSold,
                                   (P.ProductQuantity -
IsNull(Sum(CP.TotalProduct),0)) as AvailableStock
```

```
                                        from Products P
                                               inner join Category C
                                                                     on C.CategoryID= P.CategoryID
                                               left join CustomerProducts CP
                                                                     on CP.ProductID =P.ProductID
                                               group by P.ProductID,
                                                              P.Name,
                                                              P.Price,
                                                           P.ImageUrl,
                                                              C.CategoryName,
                                                              P.ProductQuantity,
                                                              P.CategoryID) StockTable
                                               where AvailableStock between @StartRange and
@EndRange
                     END
              END TRY

                                        BEGIN CATCH
                                               PRINT('ERROR OCCUR')
                                        END CATCH
END
```

SP_GetOrdersList

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_GetOrdersList]    Script Date: 06-05-2016
01:50:52 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- Batch submitted through debugger: AddNewCategory.sql|0|0|G:\Asp.net
Hub\Project\OldBooks ShoppingCart\ShoppingCart\ShoppingCart\AddNewCategory.sql
ALTER PROCEDURE [dbo].[SP_GetOrdersList]
(
       -- Add the parameters for the stored procedure here
       @Flag int
)
AS
BEGIN
       -- SET NOCOUNT ON added to prevent extra result sets from
       -- interfering with SELECT statements.
       BEGIN TRY
              if(@Flag <> 0)
                     begin
                            select * from CustomerDetails where Id=@Flag
                     end
              else
                     begin
                            select * from CustomerDetails
                     end
       END TRY
       BEGIN CATCH
```

```
              PRINT(' ERROR OCCURED')
       END CATCH


END
```

SP_GetTransactionDetails

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_GetTransactionDetails]    Script Date: 06-
05-2016 01:51:29 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- Batch submitted through debugger: AddNewCategory.sql|0|0|G:\Asp.net
Hub\Project\OldBooks ShoppingCart\ShoppingCart\ShoppingCart\AddNewCategory.sql
ALTER PROCEDURE [dbo].[SP_GetTransactionDetails]
(
       -- Add the parameters for the stored procedure here
       @TransactionNo int
)
AS
BEGIN
       -- SET NOCOUNT ON added to prevent extra result sets from
       -- interfering with SELECT statements.
       BEGIN TRY
              select P.Name,P.ImageUrl,P.Price,CP.TotalProduct as ProductQuantity from
Products P
              inner join CustomerProducts CP
               on CP.ProductID=P.ProductID
                where CP.CustomerID=@TransactionNo    --CustomerID is same as
TransactionNO
       END TRY
       BEGIN CATCH
              PRINT(' ERROR OCCURED')
       END CATCH


END
```

SP_OrderStatus

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_OrderStatus]    Script Date: 06-05-2016
01:52:21 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[SP_OrderStatus]
```

```
(
       @TransactionNo int,
       @OrderStatus varchar(300),
       @Flag int
)

AS
BEGIN
       begin try
       if(@Flag=1)
       begin
       insert into DeliveryStatus(TransactionNo,StatusMessage) values
       (
              @TransactionNo,
              @OrderStatus
       )
       end
       select StatusMessage as ShipmentStatus,DataEntry as UpdatedOn from
DeliveryStatus where TransactionNo=@TransactionNo
       end try
       begin catch
              print('Error Occurred')
       end catch
END
```

SP_SaveCustomerDetails

```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_SaveCustomerDetails]    Script Date: 06-05-
2016 01:53:04 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- Batch submitted through debugger: AddNewCategory.sql|0|0|G:\Asp.net
Hub\Project\OldBooks ShoppingCart\ShoppingCart\ShoppingCart\AddNewCategory.sql
ALTER PROCEDURE [dbo].[SP_SaveCustomerDetails]
(
       @CustomerName varchar(100),
       @CustomerEmailID varchar(100),
       @CustomerPhoneNo varchar(10),
       @CustomerAddress varchar(500),
       @TotalProducts int,
       @TotalPrice int,
       @PaymentMethod varchar(100)
)
AS
BEGIN
       BEGIN TRY
              insert into
CustomerDetails(CustomerName,CustomerEmailID,CustomerPhoneNo,CustomerAddress,TotalProd
ucts,TotalPrice,PaymentMethod)

       values(@CustomerName,@CustomerEmailID,@CustomerPhoneNo,@CustomerAddress,@TotalP
roducts,@TotalPrice,@PaymentMethod)
```

```
                select @@IDENTITY AS CustomerID
        END TRY
        BEGIN CATCH
                PRINT(' ERROR OCCURED')
        END CATCH


END
```

SP_SaveCustomerProducts

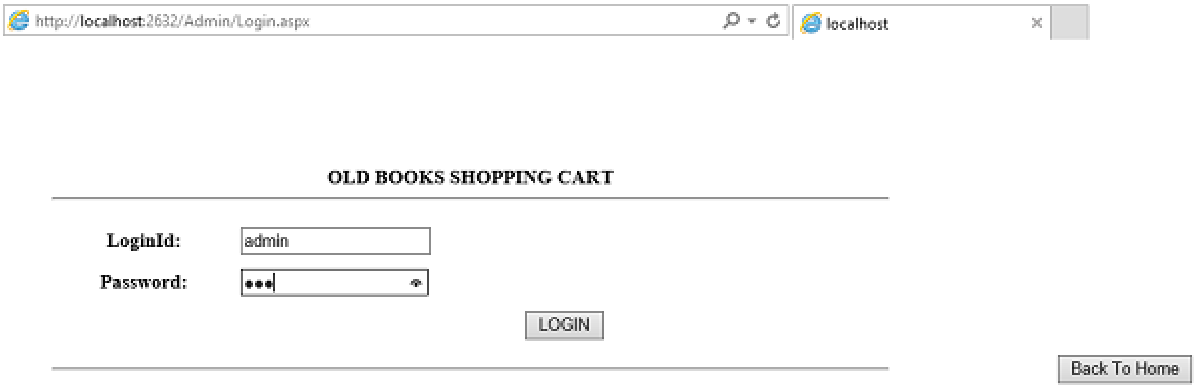
```
USE [ShoppingCartDB]
GO
/****** Object:  StoredProcedure [dbo].[SP_SaveCustomerProducts]    Script Date: 06-
05-2016 01:53:55 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [dbo].[SP_SaveCustomerProducts]
(
        @CustomerID int,
        @ProductID int,
        @TotalProduct int
)

AS
BEGIN
        begin try
        insert into CustomerProducts values(@CustomerID,@ProductID,@TotalProduct)
        end try
        begin catch
                print('Error Occurred')
        end catch
END
```

CHAPTER-7

REPORTS

The reports are the output screen of the application. Screen shots of each interface of the application are given below, which gives the clear idea about the project.

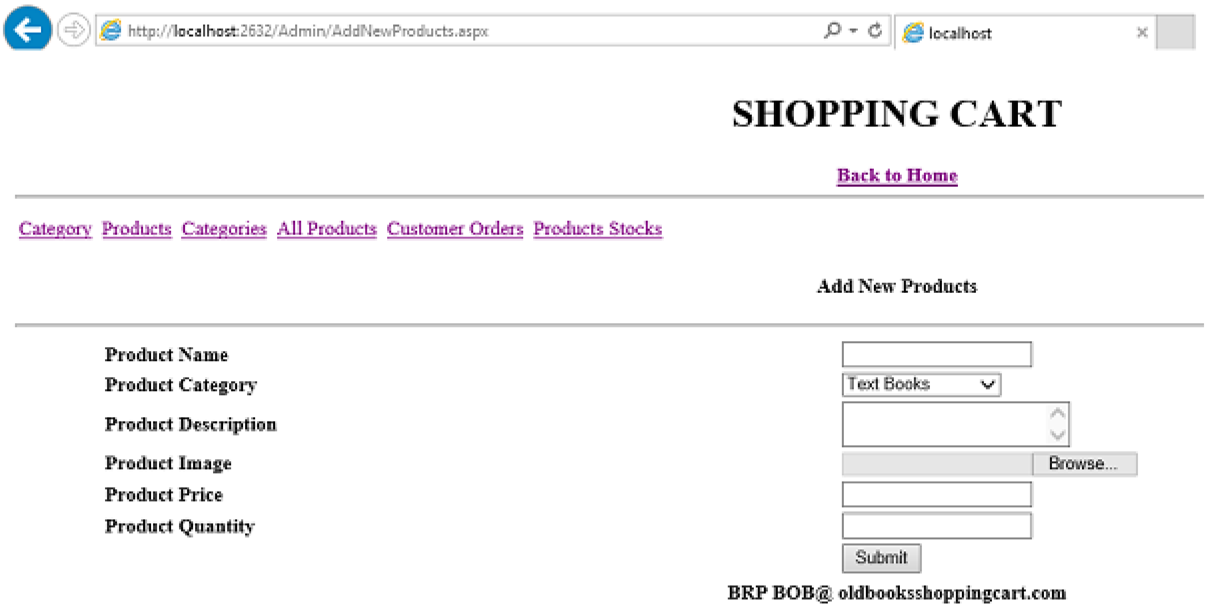
7.1. ADMIN CONTROLS AND FUNCTIONS:



LoginId: admin

Password: •••

(Admin Login Page)



http://localhost:2632/Admin/AddNewProducts.aspx

SHOPPING CART

Back to Home

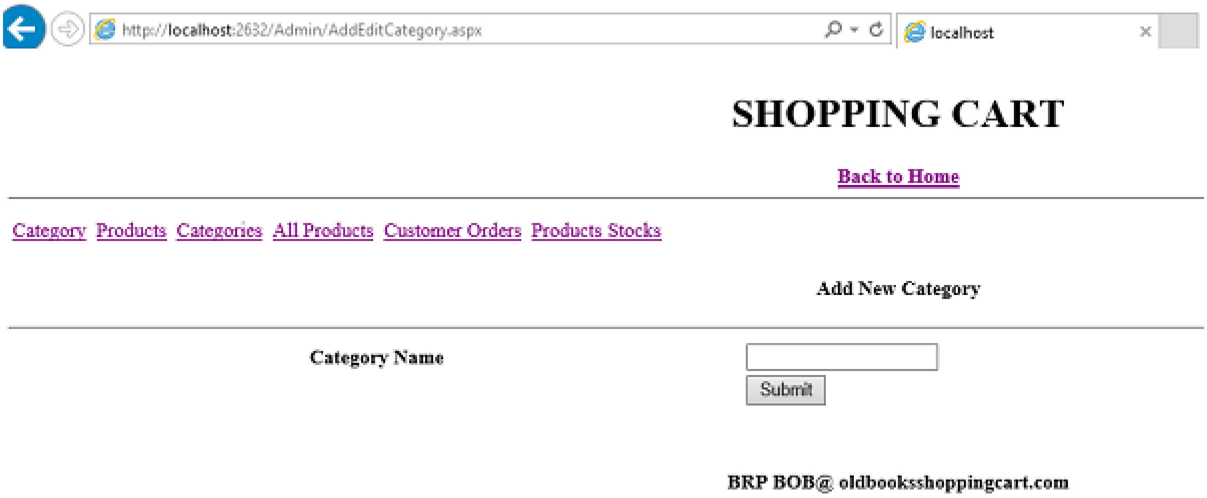
Category Products Categories All Products Customer Orders Products Stocks

Add New Products

Product Name	
Product Category	Text Books
Product Description	
Product Image	Browse...
Product Price	
Product Quantity	

BRP BOB@ oldbooksshoppingcart.com

(AddNew Product)



http://localhost:2632/Admin/AddEditCategory.aspx

SHOPPING CART

Back to Home

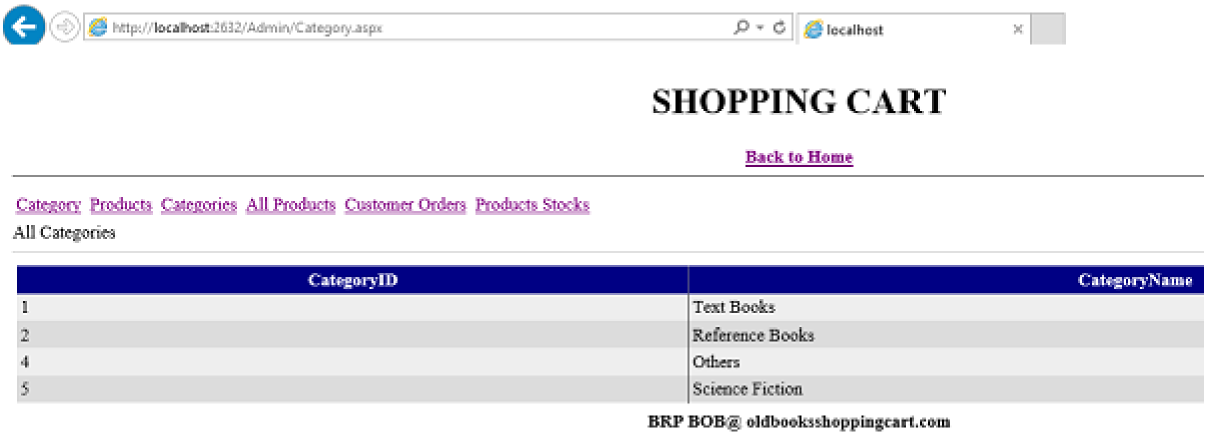
Category Products Categories All Products Customer Orders Products Stocks

Add New Category

Category Name

BRP BOB@ oldbooksshoppingcart.com

(Add New Category)



http://localhost:2632/Admin/Category.aspx

SHOPPING CART

Back to Home

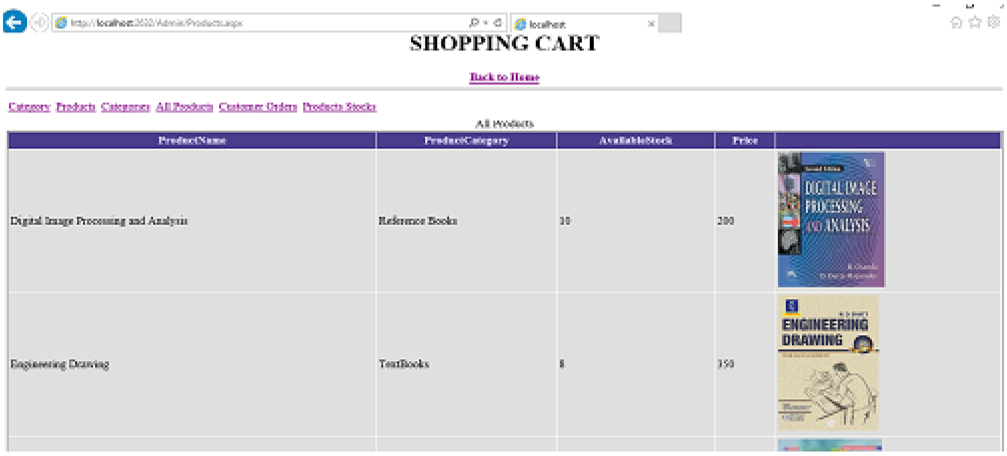
Category Products Categories All Products Customer Orders Products Stocks

All Categories

CategoryID	CategoryName
1	Text Books
2	Reference Books
4	Others
5	Science Fiction

BRP BOB@ oldbooksshoppingcart.com

(view all Categories)



http://localhost:2632/Admin/Products.aspx

SHOPPING CART

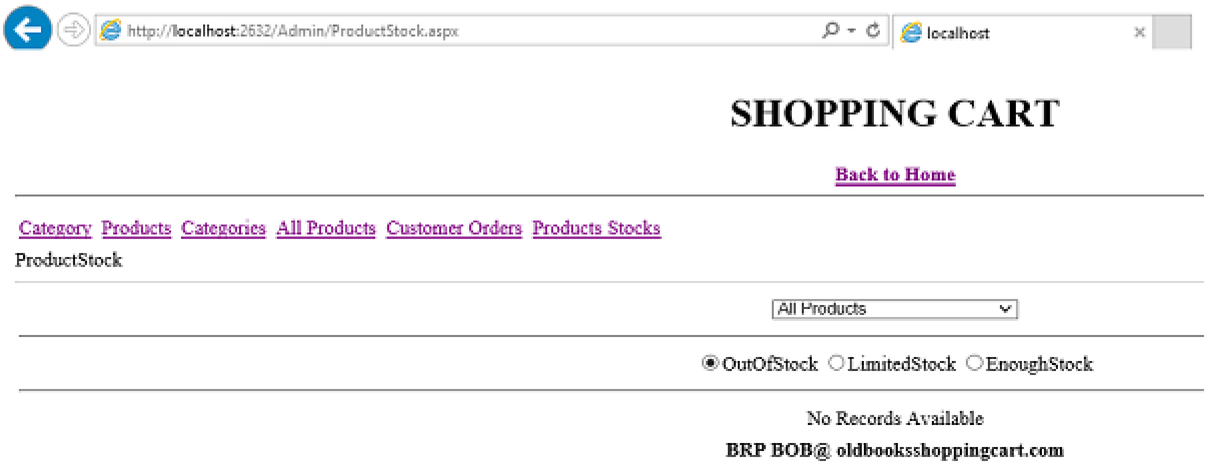
Back to Home

Category Products Categories All Products Customer Orders Products Stocks

All Products

ProductName	ProductCategory	AvailableStock	Price	
Digital Image Processing and Analysis	Reference Books	10	200	
Engineering Drawing	TextBooks	8	350	

(view all Products)



http://localhost:2632/Admin/ProductStock.aspx

SHOPPING CART

Back to Home

Category Products Categories All Products Customer Orders Products Stocks

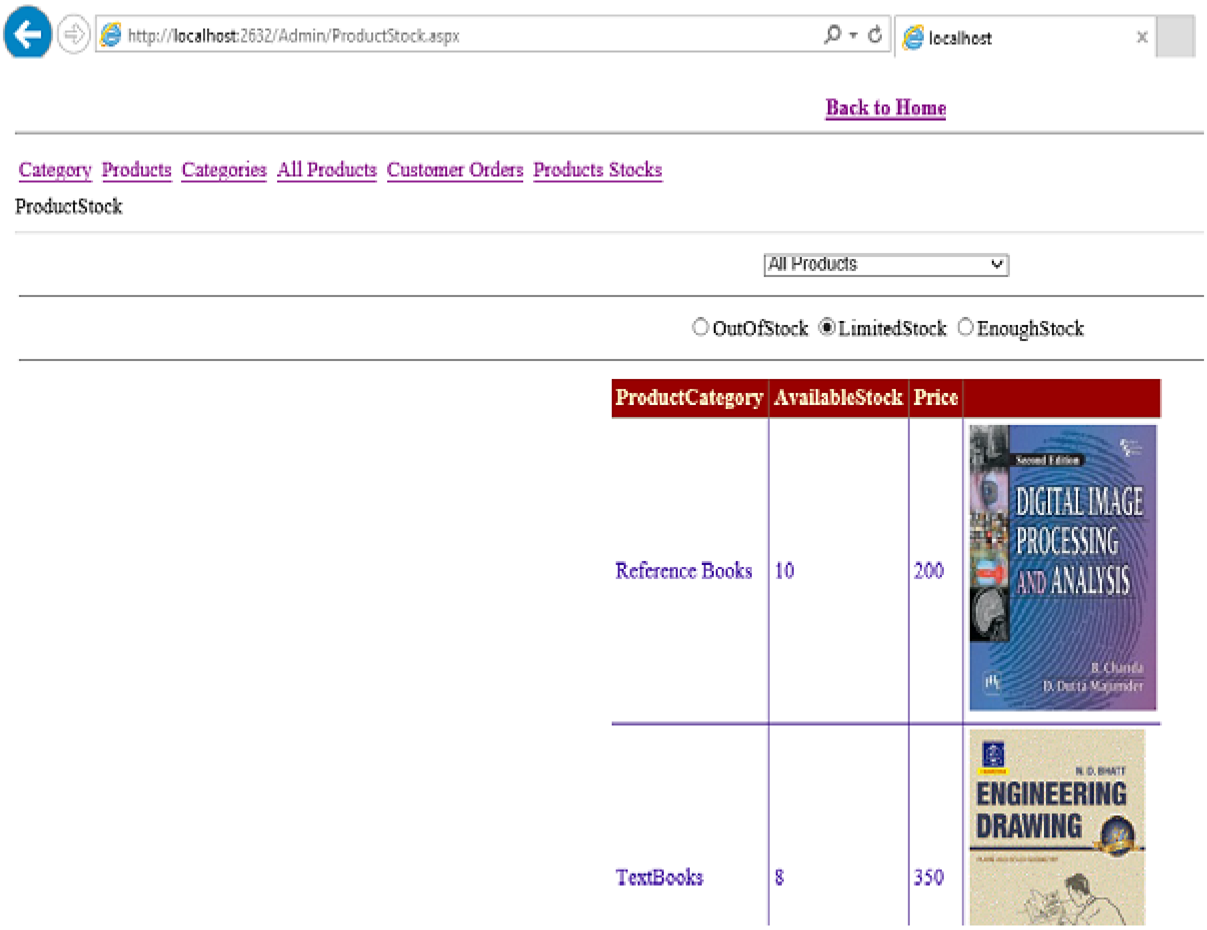
ProductStock

All Products

● OutOfStock ○ LimitedStock ○ EnoughStock

No Records Available

BRP BOB@ oldbooksshoppingcart.com



http://localhost:2632/Admin/ProductStock.aspx
localhost
Second Edition
DIGITAL IMAGE PROCESSING AND ANALYSIS
B. Chanda
D. Dutta Majumder
N. D. BHATT
ENGINEERING DRAWING

Back to Home

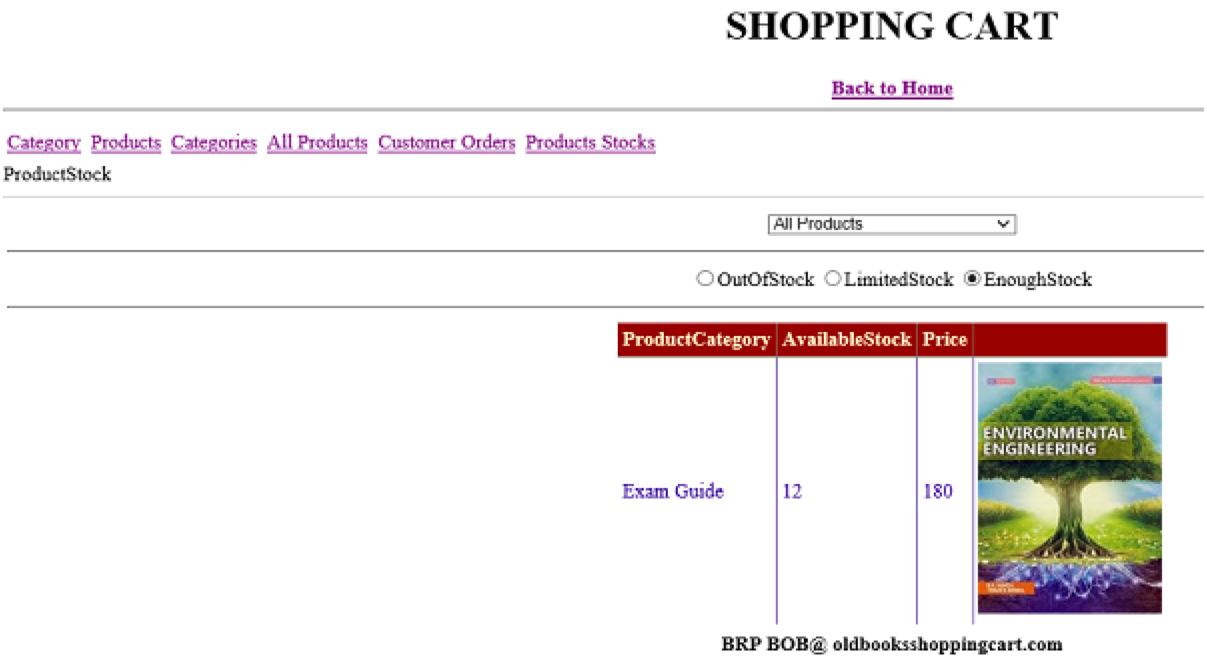
Category Products Categories All Products Customer Orders Products Stocks

ProductStock

All Products

○ OutOfStock ● LimitedStock ○ EnoughStock

ProductCategory	AvailableStock	Price	
Reference Books	10	200	
TextBooks	8	350	



SHOPPING CART

Back to Home

Category Products Categories All Products Customer Orders Products Stocks

ProductStock

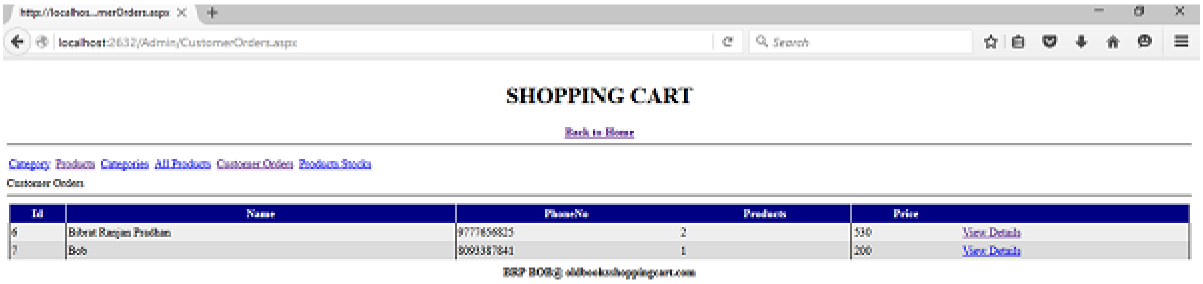
All Products

○ OutOfStock ○ LimitedStock ● EnoughStock

ProductCategory	AvailableStock	Price	
Exam Guide	12	180	

BRP BOB@ oldbooksshoppingcart.com

(The Above Three Images are for viewing The Product Stock)



http://localhos...merOrders.aspx
localhost:2632/Admin/CustomerOrders.aspx
Search

SHOPPING CART

Back to Home

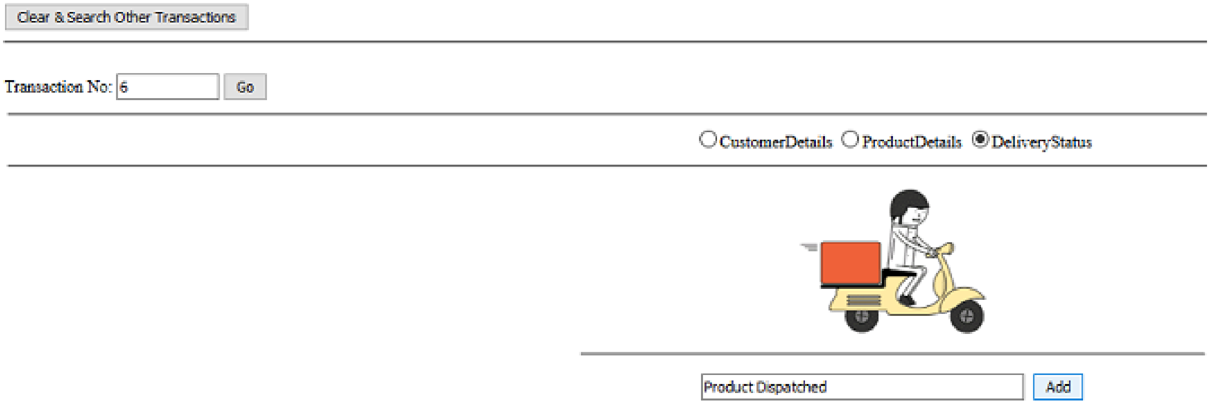
Category Products Categories All Products Customer Orders Products Stocks

Customer Orders

Id	Name	PhoneNo	Products	Price	
6	Bibrat Ranjan Pradhan	9777656825	2	530	View Details
7	Bob	8093387841	1	200	View Details

BRP BOB@ oldbooksshoppingcart.com

(View Customer Details)



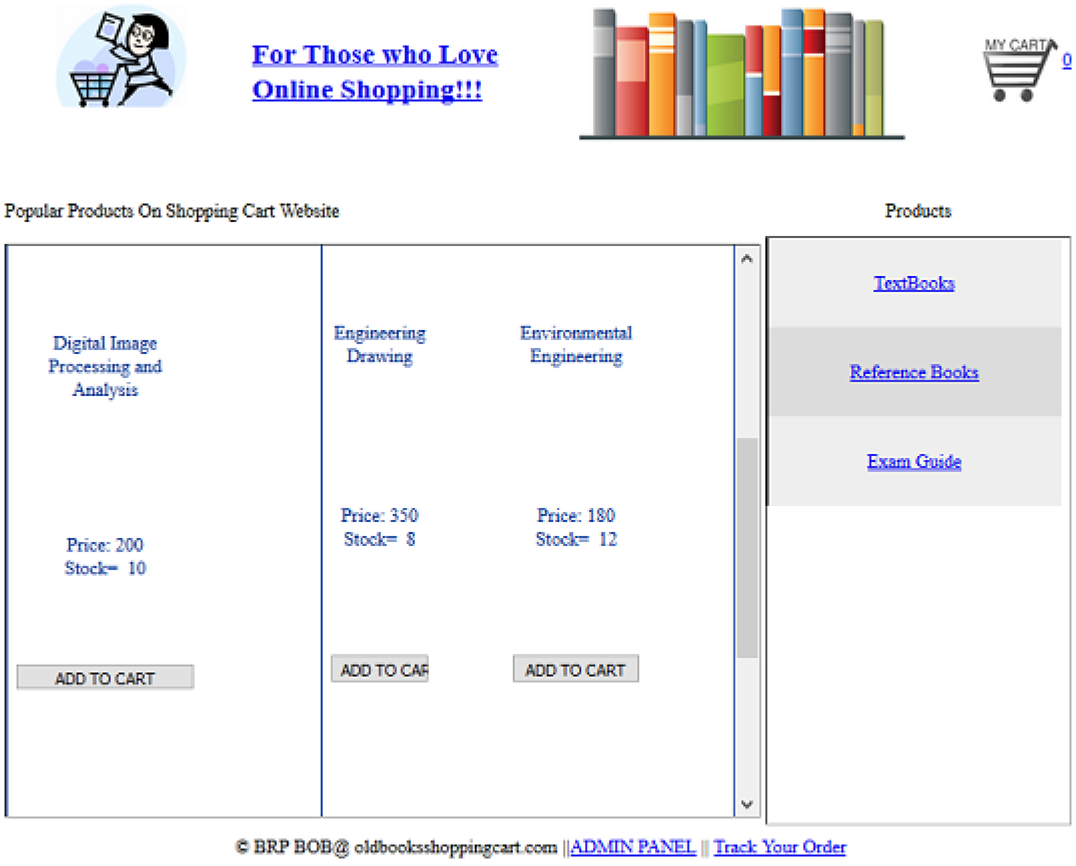
Transaction No: 6

○ CustomerDetails ○ ProductDetails ● DeliveryStatus

Product Dispatched

(Update Delivery Status)

7.2. CUSTOMER:



For Those who Love Online Shopping!!!

Popular Products On Shopping Cart Website

Products

Digital Image Processing and Analysis

Price: 200
Stock= 10

Engineering Drawing

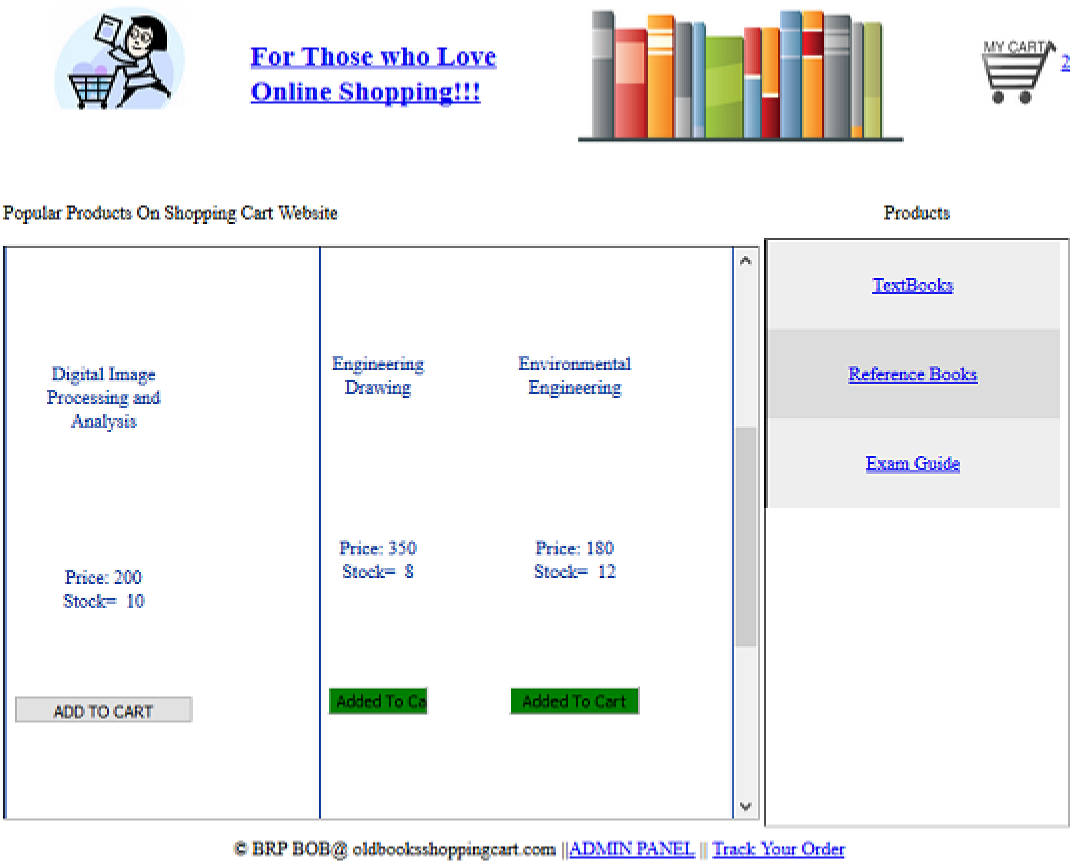
Price: 350
Stock= 8

Environmental Engineering

Price: 180
Stock= 12

© BRP BOB@ oldbooksshoppingcart.com ||ADMIN PANEL || Track Your Order

(Shopping Cart Home)



For Those who Love Online Shopping!!!

Popular Products On Shopping Cart Website

Products

Digital Image Processing and Analysis

Price: 200
Stock= 10

Engineering Drawing

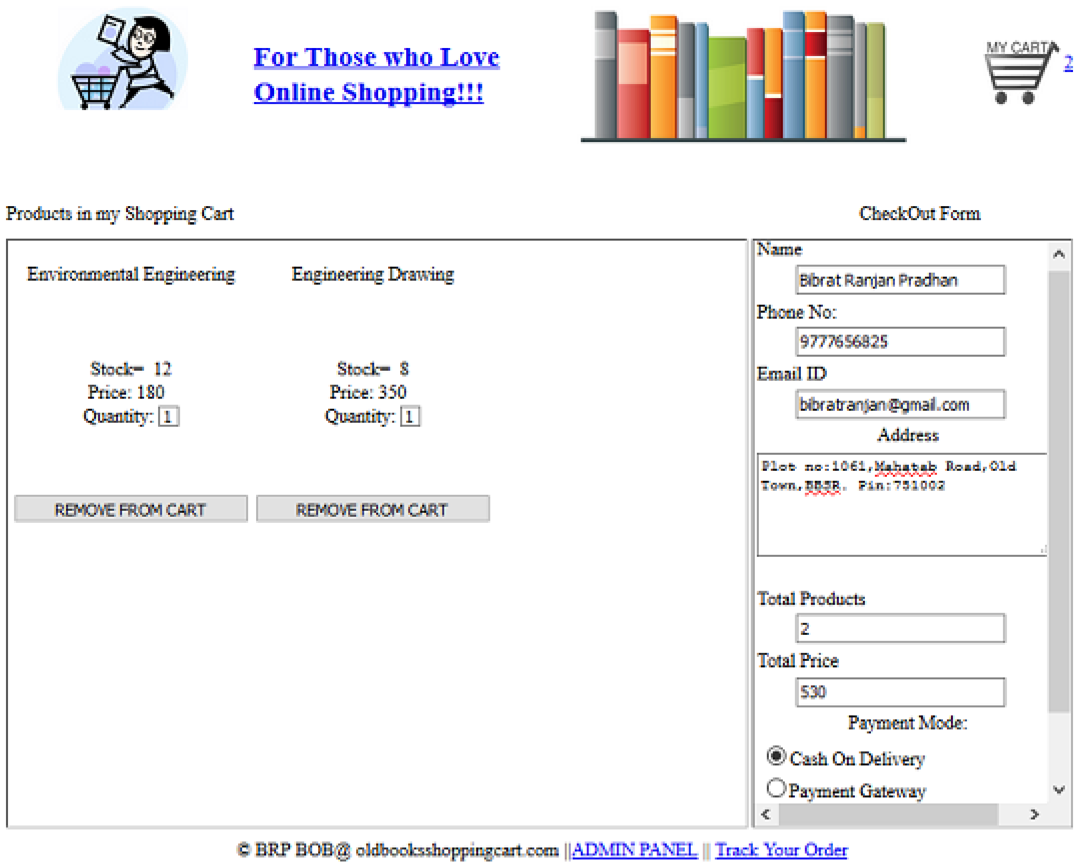
Price: 350
Stock= 8

Environmental Engineering

Price: 180
Stock= 12

© BRP BOB@ oldbooksshoppingcart.com ||ADMIN PANEL || Track Your Order

(Added To Cart)



Products in my Shopping Cart

Environmental Engineering	Engineering Drawing
Stock= 12 Price: 180 Quantity: 1	Stock= 8 Price: 350 Quantity: 1

CheckOut Form

Name: Bibrat Ranjan Pradhan

Phone No: 9777656825

Email ID: bibratranjan@gmail.com

Address: Plot no:1061,Mahatab Road,Old Town,BBSR. Pin:751002

Total Products: 2

Total Price: 530

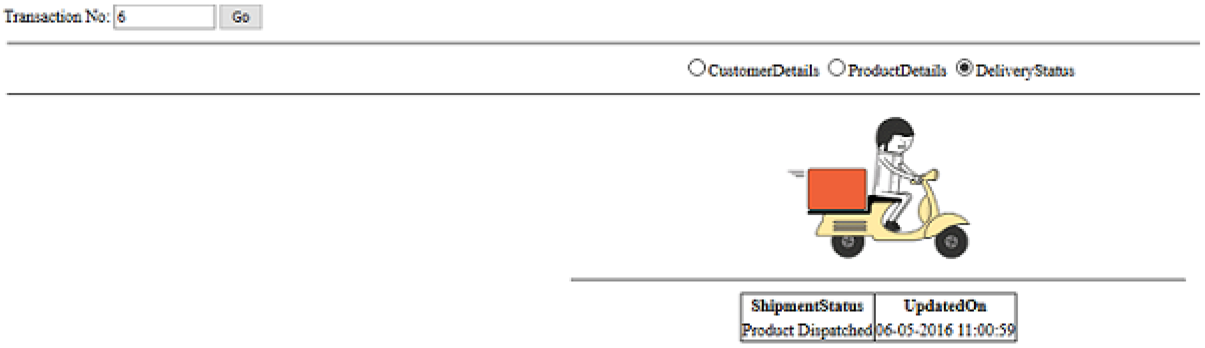
Payment Mode:

● Cash On Delivery

○ Payment Gateway

© BRP BOB@ oldbooksshoppingcart.com ||ADMIN PANEL || Track Your Order

(Cart Panel)



Transaction No: 6

○ CustomerDetails ○ ProductDetails ● DeliveryStatus

ShipmentStatus	UpdatedOn
Product Dispatched	06-05-2016 11:00:59

(Track Order)

CHAPTER-8

SYSTEM TESTING AND IMPLIMENTATION

8.1 INTRODUCTION:

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well-planned series of step that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply to both large and small-scale systems.

8.2. UNIT TESTING:

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have white box oriented and some subjects the steps are conducted in parallel.

8.3. WHITEBOX TESTING

This type of testing ensures that all independent paths have been exercised at least once. All logical decisions have been exercised on their true and false sides. All loops are executed at their boundaries and within their operational bounds. All internal data structures have been exercised to assure their validity.

CHAPTER-9

SYSTEM SECURITY

9.1. INTRODUCTION:

The protection of computer based resources that include hardware, software, data, procedures and people against unauthorized use or natural disaster is known as system security.

System security can be divided into four related issues:

- Security
- Integrity
- Privacy
- Confidentiality

SYETEM SECURITY refers to the technical innovations and applied to the hardware and operation systems to protect against deliberate or accidental damage from a defined threat.

DATA SECURITY is the protection of data from loss, disclosure, modification and destruction.

SYSTEM INTIGRITY refers to the power functioning of hardware and programs, appropriate physical security and safety against external threats such as eavesdropping and wiretapping.

PRIVACY defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination about it.

CONFIDENTALITY is a special status given to sensitive information in a database to minimize the possible invasion of privacy. It is an attribute of information that characterizes its need for protection.

9.2 ADMIN VALIDATION:

Admin validation is done using info in web.config file

CHAPTER-10

CONCLUSION

The Online Shopping system (OSS) application enables vendors to set up online shops, customers to browse through the shops, and a system administrator to approve and reject requests for new shops and maintain lists of shop categories. Also the developer is designing an online shopping site to manage the items in the shop and also help customers purchase them online without having to visit the shop physically.The online shopping system will use the internet as the sole method for selling goods to its consumers.

BIBLOGRAPHY:

[1] Rajib Mall, “Fundamentals of Software engineering”, PHI publications.

[2] W3Schools.com

[3] www.tuitorialspoint.com/C#/

[4] www.github.com/trending/C#/

[5] www.github.com/trending/Asp.Net/

[6] YouTube.com