# ASP .NET实验报告01

## **实验一:(可以直接访问<http://bingqiangzhou.cn:8080/Exp0301/> 查看效果)**

### （1）实验内容

设计并实现一个网上食品商店，其中包括商品的浏览、商品选购，查看购物车

结账，生成订单，查看订单等功能。在网站中需要使用的技术有：

1. 使用用户控件和母版页保证网站一致性；
2. 使用CSS完成页面布局
3. 数据表的显示和编辑；
4. 数据表的同步；
5. 使用内存表和Session创建购物车
6. 将内存中的数据表存入订单表等。
7. 使用模板列改善数据表的显示，使用户界面友好。

### （2）程序源码

#### （2.1）Default.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/Default.Master" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="WebApplication1.Default" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<script src="JS/Default.js" type="text/javascript"></script>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container">

<div class="row">

<asp:Repeater runat="server" DataSourceID="SqlDataSource1">

<ItemTemplate>

<div class="card col-md-6 col-lg-3"> <img class="card-img-top" src='<%# "images/"+Eval("Image") %>' alt="Card image cap">

<div class="card-body">

<h5 class="card-title">￥<%#Eval("Price")%></h5>

<p class="card-text"><%#Eval("Name")%></p>

<button type="button" class="btn btn-primary" title="Added Successfully!"

data-container="body" data-toggle="popover" data-placement="top"

data-content='<%#Eval("Name")%>(￥<%#Eval("Price")%>)' value="<%#Eval("CommodityID")%>">Add to shopping cart</button> </div>

</div>

</ItemTemplate>

</asp:Repeater>

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionString="<%$ ConnectionStrings:RemoteConnectionString %>" SelectCommand="SELECT \* FROM [Commodity]"></asp:SqlDataSource>

</div>

</div>

<div id="myDiv"></div>

</asp:Content>

#### （2.2）Default.Master

<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Default.master.cs" Inherits="WebApplication1.Default1" %>

<!DOCTYPE html>

<html>

<head runat="server">

<title></title>

<link type="text/css" rel="stylesheet" href="Content/bootstrap.css" />

<script src="Scripts/jquery-3.0.0.js" type="text/javascript"></script>

<script src="Scripts/bootstrap.js" type="text/javascript"></script>

<script src="Scripts/bootstrap.bundle.js" type="text/javascript"></script>

<asp:ContentPlaceHolder ID="head" runat="server">

</asp:ContentPlaceHolder>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container">

<a class="navbar-brand" href="Default.aspx">Not A Mall</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent1" aria-controls="navbarSupportedContent1" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent1">

<ul class="navbar-nav mr-auto">

<li class="nav-item active"> <a class="nav-link" href="ShoppingCart.aspx">Shopping Cart</a> </li>

<li class="nav-item active"> <a class="nav-link" href="Order.aspx">Order</a> </li>

</ul>

</div>

</div>

</nav>

<form id="form1" runat="server">

<div>

<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

</asp:ContentPlaceHolder>

</div>

</form>

</body>

</html>

#### （2.3）ShoppingCart.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/Default.Master" AutoEventWireup="true" CodeBehind="ShoppingCart.aspx.cs" Inherits="WebApplication1.ShoppingCart" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<title>Exp03--Shopping Cart</title>

<link rel="stylesheet" type="text/css" href="CSS/main.css" />

<script type="text/javascript" src="JS/ShoppingCart.js"></script>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="container col-sm-12">

<table class="table">

<thead class="thead-light">

<tr>

<th scope="col"><%--<div class="custom-control custom-checkbox">

<input type="checkbox" class="custom-control-input" id="customCheck1">

<label class="custom-control-label" for="customCheck1">Select All</label>

</div>--%></th>

<th scope="col">Commodity</th>

<th scope="col">Name</th>

<th scope="col">Price</th>

<th scope="col">Quantity</th>

<th scope="col">Operator</th>

</tr>

</thead>

<tbody>

<asp:Repeater ID="Repeater1" DataSourceID="SqlDataSource1" runat="server">

<ItemTemplate>

<tr>

<th scope="row"><div class="custom-control custom-checkbox">

<input type="checkbox" class="custom-control-input" id='Select\_<%# Eval("CommodityID")%>'>

<label class="font-weight-bold custom-control-label" for='Select\_<%# Eval("CommodityID")%>'></label>

</div></th>

<td><img src='images/<%# Eval("Image")%>' width="60" alt="..." class="img-thumbnail"></td>

<td><%# Eval("Name")%></td>

<td>￥<span id='Price\_<%# Eval("CommodityID")%>' class="price"><%# Eval("Price")%></span></td>

<td><div class="btn-group" role="group" aria-label="First group">

<input type="text" class="form-control" style="width:80px;" readonly id='Input\_<%# Eval("CommodityID")%>' value='<%# Eval("Amount")%>' />

<button type="button" class="btn btn-toolbar" id='Plus\_<%# Eval("CommodityID")%>'>+</button>

<button type="button" class="btn btn-light" id='Minus\_<%# Eval("CommodityID")%>'>-</button>

</div></td>

<td><button type="button" class="btn btn-light" id='Delete\_<%# Eval("CommodityID")%>'>Delete</button></td>

</tr>

</ItemTemplate>

</asp:Repeater>

</tbody>

<tfoot class="bg-light">

<tr>

<td><div class="custom-control custom-checkbox">

<input type="checkbox" class="custom-control-input" id="Select\_All">

<label class="custom-control-label" for="Select\_All">Select All</label></div></td>

<td></td>

<td></td>

<td></td>

<td><label class="text-danger font-weight-bold">Altogether: &nbsp;&nbsp;￥<span id="totalPrice">10000</span></label></td>

<td><button type="button" class="btn" id="Purchase" disabled>Purchase</button></td>

</tr>

</tfoot>

</table>

</div>

<asp:SqlDataSource ID="SqlDataSource1" runat="server"

ConnectionString="<%$ ConnectionStrings:RemoteConnectionString %>"

SelectCommand="SELECT Commodity.CommodityID, Commodity.Name, Commodity.Price, Commodity.Image, ShoppingCart.Amount FROM Commodity INNER JOIN ShoppingCart ON Commodity.CommodityID = ShoppingCart.CommodityID"></asp:SqlDataSource>

</asp:Content>

#### （2.4）Order.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/Default.Master" AutoEventWireup="true" CodeBehind="Order.aspx.cs" Inherits="WebApplication1.Order" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

<title>Exp03--Order</title>

<link rel="stylesheet" type="text/css" href="CSS/main.css" />

<script src="JS/Order.js" type="text/javascript"></script>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<% foreach (var orderList in OrderInfoList)

{

double totalPrice = 0; %>

<div class="container col-sm-12" style="margin:2% 0">

<table class="table">

<thead class="thead-light">

<tr>

<th scope="row"></th>

<th scope="col">Commodity</th>

<th scope="col">Name</th>

<th scope="col">Price</th>

<th scope="col">Quantity</th>

<th scope="col">Amount</th>

</tr>

</thead>

<tbody>

<%foreach (var info in orderList.Value.CommodityList)

{%>

<tr>

<td></td>

<td><img src='images/<%=info.CommodityImage %>' width="60" alt="..." class="img-thumbnail"></td>

<td><%=info.CommodityName%></td>

<td>￥<span><%=info.CommodityPrice%></span></td>

<td><%=info.CommodityAmount%></td>

<td>￥<span><%=info.CommodityAmount\*info.CommodityPrice%></span></td>

</tr>

<% totalPrice += info.CommodityAmount \* info.CommodityPrice;

} %>

</tbody>

<tfoot class="bg-light">

<tr>

<td></td>

<td class="font-weight-bold">OrderNo:&nbsp;<%=orderList.Key %></td>

<td></td>

<td></td>

<td><label class="text-danger font-weight-bold">Altogether: &nbsp;&nbsp;￥<span><%=totalPrice %></span></label></td>

<td><button type="button" class="btn btn-outline-primary" id='DeleteOrder\_<%=orderList.Key %>'>Delete Order</button></td>

</tr>

</tfoot>

</table>

</div>

<% }%>

</asp:Content>

#### （2.5）Order.aspx.cs

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication1

{

public partial class Order : System.Web.UI.Page

{

private string ConnectionString

= System.Web.Configuration.WebConfigurationManager.ConnectionStrings["RemoteConnectionString"].ConnectionString;

public Dictionary<string, OrderList> OrderInfoList;

protected void Page\_Load(object sender, EventArgs e)

{

this.Title = "Exp03--Order";

BuildOrderInfoList(GetOrderInfo());

}

public DataSet GetOrderInfo()

{

SqlConnection sqlConnection = new SqlConnection(ConnectionString);

sqlConnection.Open();

SqlDataAdapter sqlDataAdapter = new SqlDataAdapter("EXEC Proc\_SelectOrderInfo", sqlConnection);

DataSet dataSet = new DataSet();

sqlDataAdapter.Fill(dataSet);

sqlDataAdapter.Dispose();

sqlConnection.Close();

return dataSet;

}

public void BuildOrderInfoList(DataSet dataSet)

{

OrderInfoList = new Dictionary<string, OrderList>();

DataTable dataTable = dataSet.Tables[0];

DataRowCollection dataRow= dataTable.Rows;

foreach(DataRow row in dataRow)

{

string commodityID = (string)row["CommodityID"];

string commodityName = (string)row["Name"];

string commodityImage = (string)row["Image"];

double commodityPrice = (double)row["Price"];

int commodityAmount = (int)row["Amount"];

CommodityInfo commodity = new CommodityInfo(commodityID, commodityName, commodityImage, commodityPrice, commodityAmount);

if (OrderInfoList.ContainsKey((string)row["OrderID"]) == false)

{

OrderList orderList = new OrderList();

orderList.CommodityList.Add(commodity);

OrderInfoList.Add((string)row["OrderID"], orderList);

}

else

{

OrderList orderList = OrderInfoList[(string)row["OrderID"]];

orderList.CommodityList.Add(commodity);

OrderInfoList[(string)row["OrderID"]] = orderList;

}

}

}

public class OrderList

{

private List<CommodityInfo> Commodities = new List<CommodityInfo>();

public List<CommodityInfo> CommodityList { get => Commodities; set => Commodities = value; }

}

public class CommodityInfo

{

private string commodityID;

private string commodityName;

private string commodityImage;

private double commodityPrice;

private int commodityAmount;

public string CommodityID { get => commodityID; set => commodityID = value; }

public string CommodityName { get => commodityName; set => commodityName = value; }

public string CommodityImage { get => commodityImage; set => commodityImage = value; }

public double CommodityPrice { get => commodityPrice; set => commodityPrice = value; }

public int CommodityAmount { get => commodityAmount; set => commodityAmount = value; }

public CommodityInfo(string commodityID, string commodityName, string commodityImage, double commodityPrice, int commodityAmount)

{

CommodityID = commodityID;

CommodityName = commodityName;

CommodityImage = commodityImage;

CommodityPrice = commodityPrice;

CommodityAmount = commodityAmount;

}

}

}

}

#### （2.6）Default.js

$(document).ready(

function () {

//alert("hello");

$('[data-toggle="popover"]').popover();

$("button").click(

function () {

$(this).popover('show');

var that = this;

$.ajax({

type: "post",

url: "/Ashx/ShoppingCartFunction.ashx",

data: "Operator=Add&CommodityID=" + $(that).val(),

dataType:"text",

async: true,

success: function (data) {

setTimeout(function () {

$(that).popover('hide');

}, 1000);

}

});

}

)

}

)

#### （2.7）Order.js

$(document).ready(

function () {

//alert("hello");

$("button").click(

function () {

var str = this.id;

var strs = str.split('\_');

var id = strs[1];

if(strs[0] != "Delete")

{

return;

}

$.ajax({

type: "post",

url: "/Ashx/OrderFunction.ashx",

data: "OrderID=" + id,

dataType: "text",

async: true,

success: function () {

location.reload();

}

});

}

)

}

)

#### （2.8）ShoppingCart.js

$(document).ready(

function () {

//alert("hello");

CalculateTotalPrice();

if ($("img").length > 0) {

$("#Purchase").addClass("btn-outline-primary");

$("#Purchase").removeAttr("disabled");

}

$("#Select\_All").click(

function () {

// alert($("#Select\_All").prop('checked'));

$(".custom-control-input").each(

function () {

$(this).prop('checked', $("#Select\_All").prop('checked'));

}

)

}

);

$("button").click(

function () {

var id = this.id;

var str = id.split('\_');

var Control = str[0];

var CommodityID = str[1];

//alert(Control + " " + CommodityID);

if (Control == "Delete") {

DeleteCommodity(CommodityID);

} else if (Control == "Plus") {

AddCommodity(CommodityID);

} else if (Control == "Minus") {

MinusCommodity(CommodityID);

} else if (Control == "Purchase") {

PurchaseCommodity();

}

}

);

function DeleteCommodity(CommodityID) {

$.ajax({

type: "post",

url: "/Ashx/ShoppingCartFunction.ashx",

data: "Operator=Delete&CommodityID=" + CommodityID,

dataType: "text",

async: true,

success: function (data) {

location.reload();

}

});

}

function MinusCommodity(CommodityID) {

var InputID = "#Input\_" + CommodityID;

var number = parseInt($(InputID).val());

if (number <= 1) {

return;

}

$.ajax({

type: "post",

url: "/Ashx/ShoppingCartFunction.ashx",

data: "Operator=Minus&CommodityID=" + CommodityID,

dataType: "text",

async: true,

success: function (data) {

$(InputID).val(number - 1);

CalculateTotalPrice();

}

});

}

function AddCommodity(CommodityID) {

$.ajax({

type: "post",

url: "/Ashx/ShoppingCartFunction.ashx",

data: "Operator=Add&CommodityID=" + CommodityID,

dataType: "text",

async: true,

success: function (data) {

var inputID = "#Input\_" + CommodityID;

var number = parseInt($(inputID).val()) + 1;

//alert(inputID + " " + number);

$(inputID).val(number);

CalculateTotalPrice();

}

});

}

function PurchaseCommodity() {

var CommodityList = "";

$(".custom-control-input").each(

function () {

if ($(this).prop('checked') && this.id != "Select\_All") {

var id = this.id;

var str = id.split('\_');

var CommodityID = str[1];

var InputID = "#Input\_" + CommodityID;

CommodityList += CommodityID + "-"+$(InputID).val() + "\_";

}

}

)

CommodityList = CommodityList.substring(0, CommodityList.length - 1);

if (CommodityList.length <= 2) {

return;

}

//alert(CommodityList);

$.ajax({

type: "post",

url: "/Ashx/ShoppingCartFunction.ashx",

data: "Operator=Purchase&CommodityList=" + CommodityList,

dataType: "text",

async: true,

success: function (data) {

window.location.reload();

}

});

}

function CalculateTotalPrice() {

var totalPrice = 0;

$(".price").each(

function () {

var id = this.id;

var str = id.split('\_');

var CommodityID = str[1];

var price = parseInt($(this).html());

var InputID = "#Input\_" + CommodityID;

var number = parseInt($(InputID).val());

totalPrice += price \* number;

}

);

$("#totalPrice").html(totalPrice);

}

}

)

#### （2.9）OrderFunction.ashx

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

namespace WebApplication1.Ashx

{

/// <summary>

/// Summary description for OrderFunction

/// </summary>

public class OrderFunction : IHttpHandler

{

string ConnectionString

= System.Web.Configuration.WebConfigurationManager.ConnectionStrings["RemoteConnectionString"].ConnectionString;

public void ProcessRequest(HttpContext context)

{

//context.Response.ContentType = "text/plain";

//context.Response.Write("Hello World");

DeleteOrder(context.Request.Params["OrderID"]);

}

private void DeleteOrder(string id)

{

SqlConnection sqlConnection = new SqlConnection(ConnectionString);

sqlConnection.Open();

SqlCommand sqlCommand = new SqlCommand();

sqlCommand.Connection = sqlConnection;

sqlCommand.CommandType = System.Data.CommandType.StoredProcedure;

sqlCommand.CommandText = "Proc\_DeleteOrderInfo";

SqlParameter sqlParameter = new SqlParameter("@ID", id);

sqlCommand.Parameters.Add(sqlParameter);

sqlCommand.ExecuteNonQuery();

sqlConnection.Close();

}

public bool IsReusable

{

get

{

return false;

}

}

}

}

#### （2.10）ShoppingCartFunction.ashx

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

namespace WebApplication1.Ashx

{

/// <summary>

/// Summary description for AddCommodityToShoppingCart

/// </summary>

public class ShoppingCartFunction : IHttpHandler

{

private string ConnectionString

= System.Web.Configuration.WebConfigurationManager.ConnectionStrings["RemoteConnectionString"].ConnectionString;

public void ProcessRequest(HttpContext context)

{

context.Response.ContentType = "text/plain";

//context.Response.Write("Hello World");

//System.Diagnostics.Debug.Write(context.Request.Params["CommodityID"]);

//context.Response.Write(context.Request.Params["CommodityID"]);

context.Response.Write(context.Request.Params["Operator"]);

//Add Commodity

if (context.Request.Params["Operator"].Equals("Add"))

{

AddCommodity(context.Request.Params["CommodityID"]);

}

//Minus Commodity

else if (context.Request.Params["Operator"].Equals("Minus"))

{

MinusCommodity(context.Request.Params["CommodityID"]);

}

//Delete Commodity

else if (context.Request.Params["Operator"].Equals("Delete"))

{

DeleteCommodity(context.Request.Params["CommodityID"]);

}

//Purchase Commodity

else if (context.Request.Params["Operator"].Equals("Purchase"))

{

PurchaseCommodity(context.Request.Params["CommodityList"]);

}

}

public void AddCommodity(string ID)

{

SqlConnection sqlConnection = new SqlConnection(ConnectionString);

sqlConnection.Open();

SqlCommand command = new SqlCommand();

command.CommandType = CommandType.StoredProcedure;

command.CommandText = "Proc\_AddCommodityToShoppingCart";

command.Connection = sqlConnection;

command.Parameters.Add(new SqlParameter("@ID", ID));

command.ExecuteNonQuery();

sqlConnection.Close();

}

public void MinusCommodity(string ID)

{

SqlConnection sqlConnection = new SqlConnection(ConnectionString);

sqlConnection.Open();

SqlCommand command = new SqlCommand();

command.CommandType = CommandType.StoredProcedure;

command.CommandText = "Proc\_MinusCommodityFromShoppingCart";

command.Connection = sqlConnection;

command.Parameters.Add(new SqlParameter("@ID", ID));

command.ExecuteNonQuery();

sqlConnection.Close();

}

public void DeleteCommodity(string ID)

{

SqlConnection sqlConnection = new SqlConnection(ConnectionString);

sqlConnection.Open();

SqlCommand command = new SqlCommand();

command.CommandType = CommandType.StoredProcedure;

command.CommandText = "Proc\_DeleteCommodityFromShoppingCart";

command.Connection = sqlConnection;

command.Parameters.Add(new SqlParameter("@ID", ID));

command.ExecuteNonQuery();

sqlConnection.Close();

}

public void PurchaseCommodity(string CommodityList)

{

string OrderNo = "O"+DateTime.Now.ToFileTimeUtc().ToString();//Generate the order number

string[] CommodityInfo = CommodityList.Split('\_');//Split single commodity from list

SqlConnection sqlConnection = new SqlConnection(ConnectionString);

sqlConnection.Open();

SqlTransaction sqlTransaction = sqlConnection.BeginTransaction();

SqlCommand command = new SqlCommand();

command.Connection = sqlConnection;

command.Transaction = sqlTransaction;

foreach(string commodity in CommodityInfo)

{

string[] str = commodity.Split('-');

string ID = str[0];

string Number = str[1];

command.CommandType = CommandType.StoredProcedure;

command.CommandText = "Proc\_AddCommodityToOrder";

command.Parameters.Add(new SqlParameter("@OrderNO", OrderNo));

command.Parameters.Add(new SqlParameter("@ID", ID));

command.Parameters.Add(new SqlParameter("@Number",int.Parse(Number)));

command.ExecuteNonQuery();

command.Parameters.Clear();

}

sqlTransaction.Commit();

sqlConnection.Close();

}

public bool IsReusable

{

get

{

return false;

}

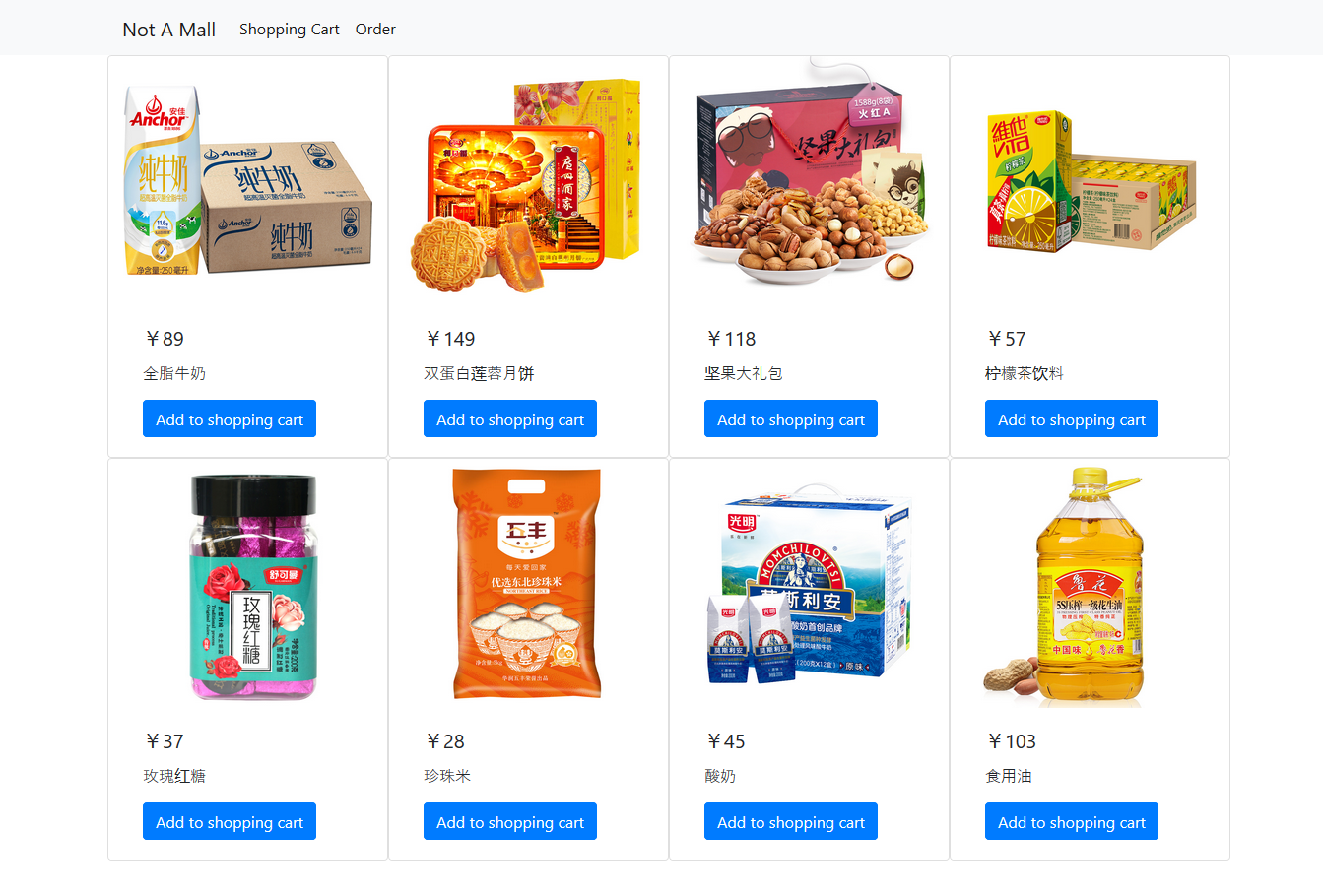
}

}

}

### （3）效果截图

#### （3.1）主页：商品展示

图1、主页：商品展示效果截图

#### （3.2）购物车页面

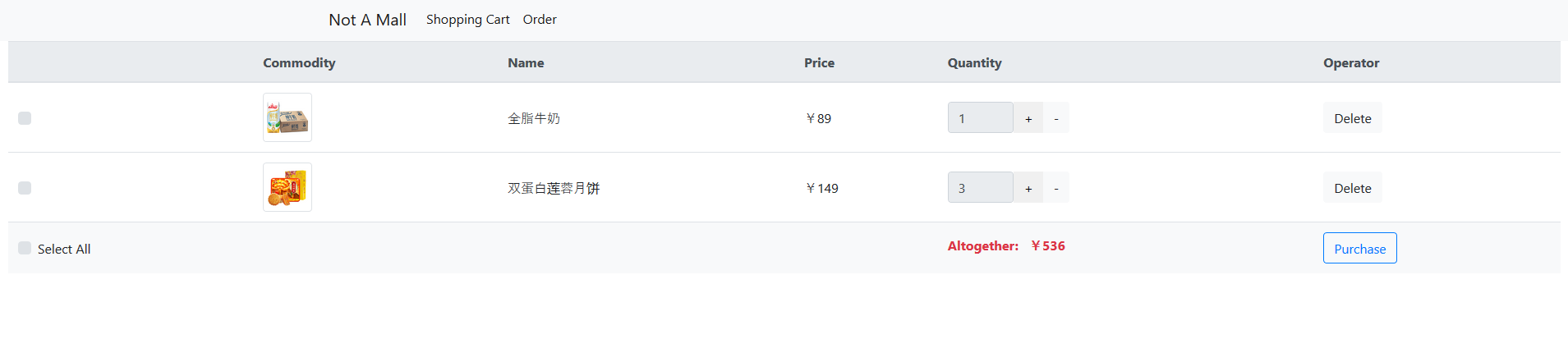


图2、购物车页面效果截图

#### （3.3）订单页面



图2、订单页面效果截图