Movie Cornucopia: An On-Line Video Order System

by

Yong Keyes

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

University of Cincinnati
College of Applied Science

March 2002

Movie Cornucopia: An On-Line Video Order System

by

Yong Keyes

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology

_____	_____
Yong Keyes	Date
_____	_____
Faculty Advisor: Robert E. Schlemmer	Date
_____	_____
Department Head: Lawrence G. Gilligan	Date

Acknowledgements

I would thank all who have helped me throughout this project: Aaron Sherling, Robert Brown, John Stullenberger, and Erma Fritsche for their expert assistances on the technical elements of my project; Nancy Hunter, for her honest feedback; my advisor, Professor Robert Schlemmer for his diligence and patience throughout the development of my project.

Dedication

To my husband, Hugh Keyes. Thank you for your immeasurable support and endurance throughout my pursuit of this degree.

Table of Contents

List of Figures

Abstract

This document discusses the creation and attributes of Yong's Video Collection, an online movie store created as my Senior Design project. Yong's Video Collection has all the functionality of an e-commerce site. It consists of three-tier architecture: a data server, a web server and a client browser.

The project is created using Active Server Pages with Microsoft SQL server 2000 database as the back-end. Any customer can browse through the movie genres, search a particular movie on the site by either actor or title, put the selected items in a shopping cart and then go through a checkout process. A registered customer can update the information and a new customer can create an account of his or her own. Collection maintenance and controls are also implemented into the project so that the database administrator can manage movie and purchase-related data.

1. Statement of the Need

I am going to develop an online Video Store. To prove my competency as a professional in Information Technology, I chose to implement a fully functional, interactive, intuitive Web site, through which a user can purchase videos on line. It is going to involve the knowledge of Database Management, Active Server Pages, Visual Basic Scripting Edition, and HTML. While keeping a user-friendly interface in mind, I will incorporate search functions by Title or Actor to demonstrate the unparalleled convenience of the online store.

With research firms projecting E-commerce revenues of more than $100 billion in 2003, businesses large and small are considering jumping into the electronic commerce marketplace (1). The Internet is quickly becoming a crucial factor in many small companies' growth strategies (1). The wholehearted embrace of E-business by General Electric that happened earlier in the year of 2000, accompanied by dramatic moves in the automotive, aerospace and energy sectors sent a signal that E-business was no longer a novelty (13). As an Information Technology professional, in order to keep pace with the cutting-edge of the industry, it is necessary to be familiar with all the turnkey elements involved in this booming technology field.

2. Review of the Literature

2.1 Introduction to E-commerce

E-commerce refers to the process of buying or selling a product or service over an electronic network. The most popular medium in which E-commerce is conducted is the Internet (16).

E-commerce encompasses three types of business transactions. First, a transaction can occur between a business and consumer. A primary example is the amazon.com. As a company that started a scant eight years ago, amazon.com is selling more than five times as many books online than the long established bookseller Barnes and Noble (16).

A second general form of E-commerce involves transactions between one business and another. A business that engages in this type of E-commerce is typically less visible to consumers and, therefore, to the general public. A good example of a company that engages in business-to-business E-commerce is Cisco Systems. Cisco Systems creates much of the physical infrastructure of the Internet that allows businesses to communicate (16).

Finally, a form of E-commerce that has become popular involves consumer-to-consumer transactions. A well-known example is eBay. It enables its customers to auction items to other customers and collects a fee from every transaction (16).

2.2 Five Basic Steps in Setting up Shop in Cyberspace

There are five steps to complete before transacting business on the Web site (1).

2.2.1 Domain Name Registration.

It refers to the e-tailer's address in cyberspace. It has to be registered with InterNIC, the agency that registers and maintains a database of domain names. Domain name can be obtained directly from NetworkSolution.com and Register.com. However, most of the Internet Service Providers will perform this task for e-tailers (1). Considering the cost of domain name registration, I will not include this step in my senior design project.

2.2.2 Web Store Design. It refers to the initial task of construction and the ongoing responsibility of making modifications to the site (1). Since I am selling videos online, a product catalogue needs to be developed to provide information about the videos, such as title, actors, and genres a particular movie belongs to. The Web store design should also take the following factors into consideration: the look and feel of the store logo and the overall site, the navigation tools, the forms of payment in business transactions and the calculation of tax and shipping charges.

2.2.3 Server Hosting. Using a hosting service can take less than an hour to set up. It will also speed the time it takes for customers to download pages on the Web site, thus improving customers' experience with the Web site (1). In my senior design project, I will utilize Windows 2000 Server with IIS provided by the university to implement this step.

2.2.4 Payment Solutions. In order to become truly E-commerce enabled, e-tailer must have the following elements: Payment software, a merchant account, payment processing services and a gateway to connect all these elements of the payment process (1). The e-tailer will need cash register software to help easily calculate sales tax as well as shipping charges, and may want to include a shopping cart function as well (1). I will not be able to encompass all the factors mentioned above in my senior design project. I will emphasize on building the operational databases and the shopping cart.

2.2.5 Traffic Coverage. This involves registering the e-tailer's Web site with search engines and establishing relationships with sites that reach a similar

demographic group and offer premiums in exchange for links, referrals and demographic information (1). This step is beyond the scope of my senior design project. I will not implement this step.

3. Description of the Solution

3.1 User Profile

The targeted users of this online vide store will be moderately experienced computer users and Web surfers who are comfortable with a logically structured web site. The web site should have minimum complexity with easy use and navigation.

3.2 Design Protocols

My Web store design should take the following factors into consideration:

- A good page layout and graphical design for products' display
- An efficient user interface
- Clear and easy navigation
- An effective user registration and verification using username and password for membership verification
- Fully functional search capabilities, such as searching by Title or Actor
- Extensive use of ASP and HTML
- Smooth transitions from page to page
- Effective database design that can handle a large number of simultaneous transactions
- Payment Solutions and Credit Card Validation
- Calculation of tax and shipping charges

In order to develop such an E-commerce Web site, I need to incorporate a great deal of Information Technology knowledge.

- Creating the Store Database
  - Creating the Products table, Customers table, Actors table, Orders table, OrdersDetails table, and ActorDetails table

- Connecting to the Database

- Displaying Products
    - Creating the Main Store Page
    - Displaying Product Details
    - Paging Through Record Sets
- Searching for Products
    - Creating a Search Page
- Building the Transaction Databases
    - Registering Users
- Building the Shopping Cart
    - Using Native ADO Methods
- Checking Out
    - ASP Page Transactions
    - ADO Transactions
    - Database Transactions
    - Completing the Order
    - Retrieving Address and Payment Information
    - Updating Address and Payment Information
    - Processing Orders
- Working with Credit Cards
- Functionality for Store Maintenance and Controls

4. Objectives of the Project ("Deliverables")

In order to give the website a consistent look, I implemented a header file and a footer file to my project. By doing so, all pages of the web site will have a consistent look.

Header Page (header.inc):

The header.inc file has two tables.

Home Page (default.asp): Figure 3 on page 17

This orientation screen welcomes the users to the online video store web site. It has navigation links for home, search, browse, cart, account, help and check out. This screen will be the "home" page for the site.

Browse Page (browse.asp): Figure 4 on page 19

This page allows the user to browse through the store's different movie genres and their corresponding contents.

Search Page (search.asp): Figure 5 on page 21

From here, users will be able to search movies by Title or Actor via a drop down list box. A textbox is provided to accept the user's input data. A "Continue" submit button is provided on this page. After the user enters data in the textbox, he/she has to click on the "Continue" button to execute the query. A "New Search" command button resets the input data on this page.

Search Result Page (search_results.asp): Figure 6 on page 23

Based on the data entered in the search page's textbox, the Products table in the database residing on the web server will be accessed and the matching record(s) will be retrieved and displayed. An "Add to Cart" button is implemented next to each matching item.

Movie Detail Page (prod_det.asp): Figure 7 on page 27

With the click on the image or the hyperlink of the returned search results, the relevant detailed information about a specific movie will be displayed. Such movie details include: Name, Synopsis, Star, and Price. An "Add to Cart" button is implemented on this page as well.

Once the user clicks on the "Add to Cart" button, the cart.asp is triggered.

Actor's related Movie Page (asearch.asp): Figure 8 on page 30

From the prod_det.asp, with the click on the hyperlink of the actor, the user will be taken to the page of asearch.asp. On the page of asearch.asp, all movie(s) starring with the specified actor will be displayed.

Display cart items Page (cart.asp): Figure 9 and Fiugre 10 on page 34

This is the Shopping Cart Information screen. The newly added movie is displayed in the list of selected movies. If the user has already added movie(s) to the shopping cart, these movies are also displayed. The user may enter the desired quantity for each selected movie. The price for the selected item is displayed. A "Shop more" button is implemented for further shopping need within the session. A "Check out" button is implemented on this page for the continuation of the shopping process within the session. By clicking on the "Check out" button, the account_login.asp page will be triggered.

Login Page (account_login.asp): Figure 11 on page 36

For a registered user, once he/she types in the correct username, password and clicks on the "Continue" submit button, he/she continues with the check out process.

Error checking functionality is implemented on this page.

For a new visitor, he/she is able to register by clicking on the "create a new account" link. This functionality is further illustrated in the paragraph below.

Review Page (account_form.asp): Figure 12 on page 37, Figure 13 on page 40

This page allows a registered user to update the information and/or a new visitor to register. On this screen, there are fields for First Name, Last Name, Street Address, City, State, Zip/Postal Code, Country, Phone, Email, Username and Password. For a registered user, data are retrieved from the database and will be displayed in the corresponding fields. A registered user may make changes to the information.

A new user needs to enter the required data.

After the required data entry process is finished, the user may proceed by clicking on the "Continue" submit button.

Error checking functionality is implemented on this page.

Checkout Page (confirmation.asp): Figure 14 on page 41

On this screen, the selected items are displayed together with the customer's Credit Card Information, Shipping Information and Shopping Cart Information. Total cost for the order is calculated and displayed. A "Continue" button is implemented on the page. The user can make last-minute changes at this screen before he/she makes the final purchase. The user may click on the "Change" button under "Shipping Information" for the update of shipping information. The user may click on the "Change" button under "Shopping Cart Information" for the update of the shopping cart information, such as to modify the desired quantity, shop more within the session, and/or cancel the shopping order. Credit card validation functionality is implemented. With the click of the "Continue" button on this page, the on-line shopping process will be completed. A sample "order complete" page is shown in Figure 15 on page 42.

How-To (help.asp): Figure 16 on page 43

This screen will provide tips on how to perform movie searches, browse through genres, and manage the account.

Database Maintenance and Controls (admin/default.asp): Figure 17 on page 44

This allows the authorized administrator of the database to log in and manage movie and purchase-related data.

5. Design and Development

5.1 Hardware and Software Requirements

Hardware requirements are:

- *Web Server*: Intel x86 based machine with a Pentium class processor 500mhz plus, 128MB (minimum) of RAM, 2-2GB hard disk (minimum), network adapter with IP address.

- *Data Server*: Intel x86 based machine with a Pentium class processor 350mhz plus, 128MB (minimum) of RAM, 2-8GB hard disk (minimum), network adapter with IP address, tape backup drive.

A variety of software was used to develop this project:

- *Windows NT/2000 Server with IIS*: This will be used to host the Web site.

- *MS SQL Server 2000*: This will provide the data storage for the information that will be presented on the site.

- *Visual InterDev*: This will be the main tool for creating the Web pages both static and dynamic (ASP) pages.

- *Visual Basic Scripting Edition*: The subset of MS Visual Basic used on ASP for access to the database ADO object on the server.

- *Jasc Paint Shop Pro 7*: This will be used as a secondary editing tool for the graphical interfaces design of the Web pages.

- *Visio*: This will allow me to create an overall site diagram for presentation and publication of my Senior Design project.

The University's facilities will provide me with a Windows 2000 Server as my Web Server. The data server is a Microsoft SQL Server 2000. The internet connectivity is accomplished via Local Area Network. In addition, the University's facilities will provide me with Visual Studio, which includes Visual InterDev and Visual Basic. I plan to use Jasc Paint Shop Pro 7 for graphic design, which is also available through University's facilities. I will acquire Visio on my own to complete this project.

5.2 Budget

My proposed budget is based on the needed resources quoted from section 5.1 and the estimated cost of the project. A store budget table is shown in Appendix B..

5.3 Timeline

Senior Design I

- Research and information
- Develop initial idea
- Preliminary site research
- Preliminary database schema using MS Access 2000
- Design and populate database with data

Senior Design II

Week 1

- Review site layout and overall functionality
- Review table schema and make any necessary changes

Week 2

- Upsize database and tables to SQL Server
- Begin initial site creation

Week 3

- Research ASP information to develop necessary functionality
- Continue with site creation

Week 4

- Continue research on ASP and any other necessary Web technologies
- Continue with site creation

Week 5

- Work on site's graphical/visual elements
- Continue with site creation

Week 6

- Continue work on graphical/visual elements
- Continue with site creation

Week 7

- Finalize any graphical/visual elements for working prototype stage
- Finalize and debug site. Prepare site for working prototype stage

Week 8

- Work on any additional issues that arise with the project's development

Week 9

- Prepare for presentation in week 10

Week 10

- Demonstrate working prototype

Fall Quarter 2002 - Co-oping

- Implement the interactive feature of the website
- Implement the shopping cart and check out

Senior Design III

January 21st

- Troubleshooting

February 4th

- Preliminary documentation

March 6th

- Demonstrate final project
- Submission of final project, including all documentations

March 13th

- Final Presentation

Throughout the project

- Research and gather information

6. Proof of Design

The project consists of three-tier architecture: a database server, a web server and a client browser. The basic architecture of the project is shown as in Figure 1.

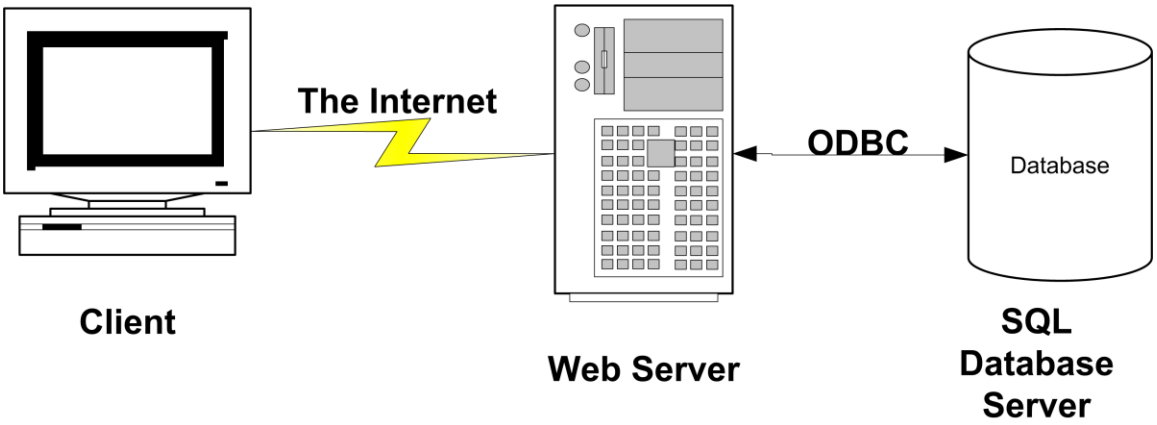


Figure 1. Architecture of the project

The project is best viewed using the Microsoft Internet Explorer (4.x or above) web browser on the client side. The web site and the database are hosted on the Web server. The web site consists of various Active Server Pages. When a user enters the URL of the project in the browser, a request is sent to the server which is then processed on the server side and sent back to the client.

For this project, I chose the Microsoft SQL Server 2000 as the database server because of its performance and the simultaneous multiple data accessing capability. To access the database from any client workstation, a system Data Source Name (DSN) needs to be established on the server side. The system DSN can be set up using the Open Database Connectivity tool in the Administrative Tools of the Control panel in Windows 2000 Professional and Windows 2000 Server.

The database named “ykeyes” is created on the server with 15 tables’ setup for the operational purpose of the store. Primary keys and foreign keys have been set up in each

table to enforce referential integrity. The database relationship diagram is shown as in Figure 2.

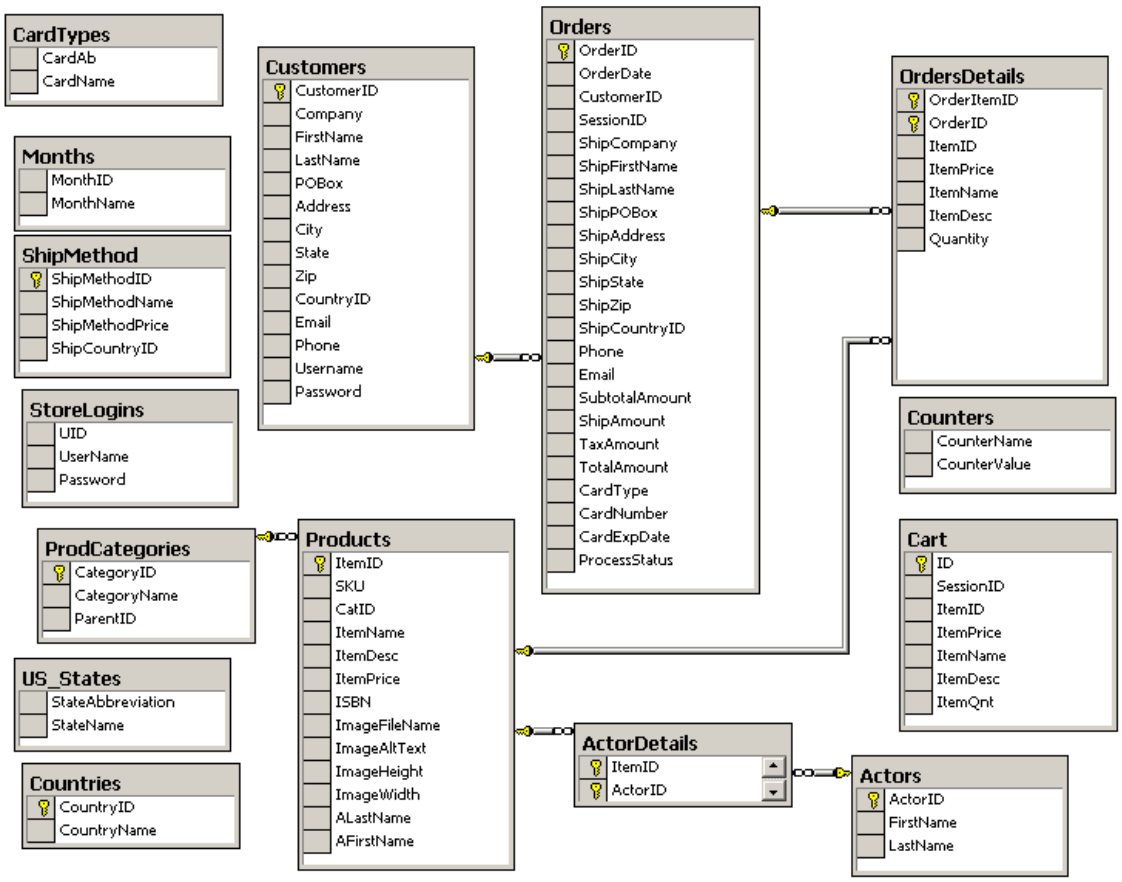


Figure 2. Database Relationship Diagram

The following code is an example of how to use SQL scripts to create the Customers table:

```
CREATE TABLE [dbo].[Customers] (
[CustomerID] [int] IDENTITY (1, 1) NOT NULL ,
[Company] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[FirstName] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[LastName] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[POBox] [int] NULL ,
[Address] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[City] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[State] [varchar] (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Zip] [varchar] (20) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
```

```
[CountryID] [int] NULL ,
[Email] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Phone] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Username] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL ,
[Password] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO
```

Following are the screenshots of how the online movie order system functions:

Figure 3. default.asp

This is the orientation screen that welcomes the visitors to the website. It is the "home" page of the website as well. The page is designed as an Active Server Page since I have footer.inc and header.inc files associated with the page. In doing so, a consistent look exists throughout the website. The image links are also available throughout the website since they are part of the header file. This feature is helpful for the users to navigate through the website. Every link is explicitly indicated by a symbolic image. The links are home, search, browse, cart, account, help and check out.

The header file consists of two tables: The first table is of one row and three columns, with one column for each image. The code for the first table is:

```
<table cellSpacing="0" cellPadding="5" width="100%" border="0">
<tr>
   <td style="WIDTH: 10%" vAlign="center" align="middle" width="25%"><IMG
height=214 src="images/Tvhappy.gif" width=224></td>
   <td style="WIDTH: 10%" vAlign="center" align="middle" width="25%"><IMG
height=150 src="images/logo.gif" width=200></td>
```

```
   <td style="WIDTH: 10%" vAlign="center" align="middle" width="25%"><IMG
style="WIDTH: 137px; HEIGHT: 150px" height=150 src="images/Slate.gif" width=150
></td>
</tr>
</table>
```

The second table is designed for the image links. The code for the second table in the header file is:

```
<table border="0" cellpadding="5" cellspacing="0" width="640">
<tr>
   <td align="center"><a href="default.asp"><img border="0"
src="images/home.gif"></a><a href="search.asp"><img border="0"
src="images/search.gif"></a><a href="browse.asp"><img border="0"
src="images/browse.gif"></a><a href="cart.asp"><img border="0"
src="images/cart.gif"></a><a href="account_login.asp"><img border="0"
src="images/account.gif"></a><a href="help.asp"><img border="0"
src="images/help.gif"></a><a href="checkout.asp"><img border="0"
src="images/checkout.gif"></a></td>
</tr>
<tr>
   <td colspan="2"><br>
</tr>
</table>
```

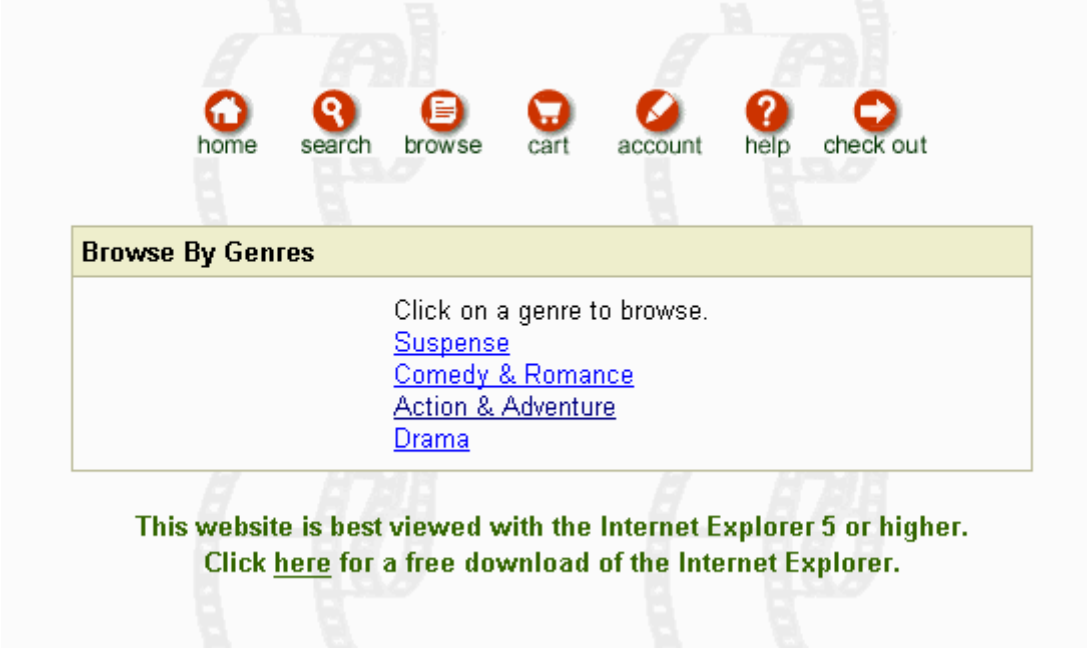


Figure 4. browse.asp

This page allows the user to browse through the store's different movie genres and their corresponding contents. The code for this page is:

```
<%Response.Buffer = true%>
<!-- #INCLUDE file="header.inc" -->
<%
        Dim objCategories, arrRecord, intNumOfRows, intParentID
        Set objCategories = New EStore_ProdCategories
        if Request("ParentID") <> "" then
                intParentID = Request("ParentID")
        else
                intParentID = 0
        end if
arrRecord = objCategories.GetCategories(intParentID, intNumOfRows)
set objCategories = Nothing
%>

<div align="center">
  <center>
<table bgColor="#bbbb99" border="0"  width="480" cellspacing="1"
cellpadding="4">
  <tr>
    <td width="472" bgcolor="#EEEECC"><font face="arial, helvetica, sans-serif"
size="-1"><b>Browse
      By Genres</b></font></td>
```

```
  </tr>
  <tr>
    <td width="472" bgcolor="#FFFFFF">
          <!--Start Inner Table-->
<TABLE border=0 cellPadding=4 cellSpacing=0 align="center">
<TR>
 <TD class=regular><font face="arial, helvetica, sans-serif" size="-1">Click on a
genre to browse.<br>
<%
If intNumOfRows > 0 then
          for i = 0 to intNumOfRows-1%>
          <a
          href="browse.asp?ParentID=<%=arrRecord(0,i)%>"><%=arrRecord(1,i)
          %>
          </a>
          <br>
<%next
else
          Response.Redirect"Search_results.asp?CatID="&intParentID%>
<%end if%>
      </font>
    </TD>
  </TR>
</TABLE>
<!--End Inner Table-->
    </td>
  </tr>
</table>
<!-- #INCLUDE file="footer.inc" -->
```

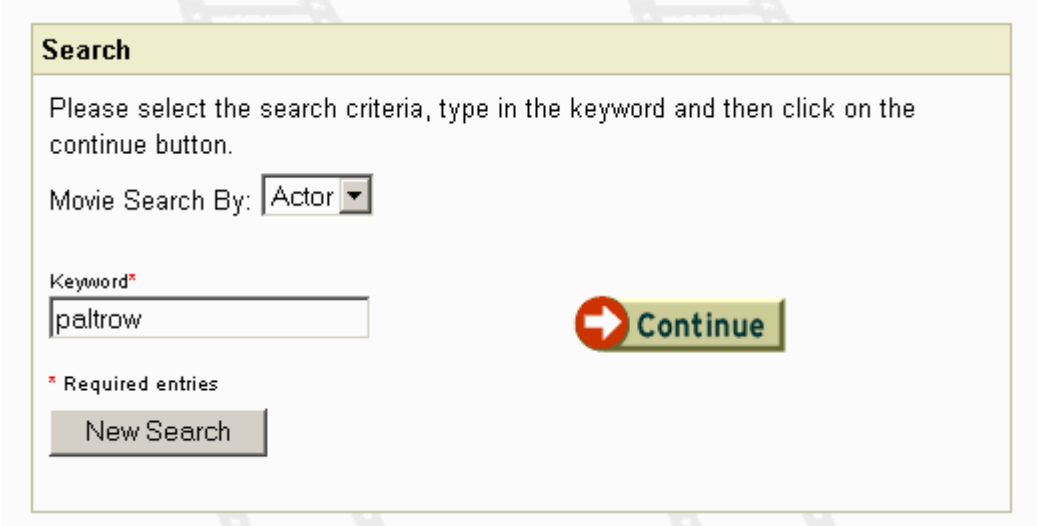


Figure 5. search.asp

This is the screen of search by actors. The user may choose movie search by Title or Actor. The code for the layout and the functionality of this page is:

```
<!--form action="search_results.asp" method="post"-->
<form action="search_results.asp" method="post" name=form>
<table bgColor="#bbbb99" border="0" width="480" cellspacing="1" cellpadding="4"
align="center">
<tr>
   <td width="472" bgcolor="#EEEECC"><font face="arial, helvetica, sans-serif"
   size="-1"><b>Search</b></font>
   </td>
</tr>
<tr>
    <td width="472" bgcolor="#FFFFFF">
        <table border="0" cellpadding="2" align="center" width="480">
           <tr>
                <td colspan="2"><font face="arial, helvetica, sans-serif" size="-1">Please
        select the search criteria, type in the keyword and then click on the continue
        button.</font>
                </td>
           </tr>
<tr>
    <td colspan="2"><font face="arial, helvetica, sans-serif" size="-1">Movie Search
By:</font>
        <SELECT id=select1 name=select1>
                <OPTION value="Title">Title</OPTION>
                <OPTION value="Actor">Actor</OPTION>
     </td>
</tr>
```

```
 <tr>
      <td width="239"><br><font face="arial, helvetica, sans-serif" size="-
2">Keyword<font color="#ff0000">*</font><br></font>
          <input size="23" name="ProdName" value="<%=ProdName%>"
maxLength="50" tabindex="1"><sub> </sub>
      </td>
      <td width="209" valign="bottom" align="left"><input type="image" alt
src="images/continue.gif" align="left" value border="0" name="Continue"
WIDTH="106" HEIGHT="28">
      </td>
</tr>
<tr>
      <td width="239">
      </td>
      <td width="209">
      </td>
 </tr>
 <tr>
      <td width="239"><font face="arial, helvetica, sans-serif" size="-2"><font
color="#ff0000">*</font>Required entries</font>
      </td>
      <td align="left" width="209">
      </td>
 </tr>
 <tr>
      <!--td colspan="2" width="456"></td-->
      <td width="209" valign="bottom" align="left">
         <P align=left><INPUT name=reset1 type=reset value="New Search" ></P>
</tr>
</table>
 
      </td>
                  </tr>
         </table>
</form>
<!-- #INCLUDE file="footer.inc" -->
```

Figure 6. search_results.asp

This is the screen for the search results. The code for this page is:

```
<%
Response.Buffer = true
Dim objProducts,CatID, Rs, RecordArray, PageSize, nPages, i, CurrentPage,
nRecords, QueryElement, nRows
PageSize = 10
If Request("Page") = "" then
        CurrentPage = 1
Else
        CurrentPage = Cint(Request("Page"))
End If
        Set objProducts = New EStore_Products
        CatID = Request("CatID")
        ProdName = Request("ProdName")
If CatID <> "" then
        RecordArray =
objProducts.GetProdByCategory(cint(CatID),CurrentPage,PageSize, nRows, nRecords,
nPages, False)
        set objProducts = Nothing
ElseIf ProdName <> "" then
```

```
        RecordArray = objProducts.GetProdByName(ProdName,CurrentPage,PageSize,
nRows, nRecords, nPages, False)
        set objProducts = Nothing
Else
        Response.Redirect("no_results.asp")
        End If
Set objProducts = Nothing
Dim MaxPages
%>
<!-- #INCLUDE file="header.inc" -->
<%If nRecords = 0 then Response.Redirect("no_results.asp")%>
<div align="center">
<center>
<table bgColor="#BBBB99" border="0"  width="480" cellspacing="1" cellpadding="4">
<tr>
    <td width="239" bgcolor="#EEEECC"><font face="arial, helvetica, sans-serif"
size="2">
<b>Search Results:<%=nRecords%> items found</b></font>
    </td>
    <td width="239" bgcolor="#EEEECC"><font face="arial, helvetica, sans-serif"
size="-1">
<!--Start Tool bar table--></font>
<table align="center" cellpadding="4" cellspacing="1" border="0"
bordercolor="#000000">
        <tr>
                <td><a href="search_results.asp?CatID=<%=CatID & "&Page=" &
(CurrentPage - 1)&"&ProdName="&ProdName%>"><font size="2" face="arial,
helvetica, sans-serif"><b>&lt;&lt;
       Prev</b></font></a></td>
                <td><font size="2" face="arial, helvetica, sans-serif"><b>Page
<%=CurrentPage%> of <%=nPages%>
       </b>
                </font></td>
                <td><a href="search_results.asp?CatID=<%=CatID & "&Page=" &
(CurrentPage + 1)&"&ProdName="&ProdName%>"><font size="2" face="arial,
helvetica, sans-serif"><b>Next
       &gt;&gt;</b>
                </font></a>
      </td>
        </tr>
</table>
      <font face="arial, helvetica, sans-serif" size="-1">
<!--End Tool bar table-->
    </font></td>
  </tr>
  <tr>
```

```
    <td width="472" bgcolor="#FFFFFF" colspan="2">
<!--Start search result table-->
<%
        For i = 0 to nRows - 1
        dim imageFileName, ImageAltText, ImageHeight, ImageWidth
        imageFileName = RecordArray(7,i)
        ImageAltText = RecordArray(8,i)
        ImageHeight = RecordArray(9,i)
        ImageWidth = RecordArray(10,i)
%>
<table align="center" width="90%" cellpadding="4" cellspacing="1" border="0"
bordercolor="#000000">
        <tr>
                <th align="left" rowspan="3">
      <table border="0" width="100%">
        <tr>
          <td width="100%">
          </td>
        </tr>
        <tr>
          <td width="100%" align="center">
      <a href="cart.asp?ProdID=<%=RecordArray(0,i)%>"><font face="arial, helvetica,
sans-serif" size="2"><img border="0" src="images/addtocart.gif" WIDTH="80"
HEIGHT="20"></font></a>
          </td>
        </tr>
      </table>
                </th>
                <th align="left" valign="bottom"><font size="2" face="arial, helvetica,
sans-serif">Name</font>
                </th>
                <th align="left" valign="bottom"><font size="2" face="arial, helvetica,
sans-serif">Synopsis</font>
                </th>
        </tr>
<tr>
                <td><font face="arial, helvetica, sans-serif" size="2"><a
href="prod_det.asp?ProdID=<%=RecordArray(0,i)%>"><%=RecordArray(3,i)%></a></
font>
                </td>
                <td><font size="2"><font face="arial, helvetica, sans-
serif"><%=RecordArray(4,i)%></font></font>
                </td>
</tr>
        <tr>
```

```
                <td><font size="2" face="arial, helvetica, sans-serif"><b><font
color="#000000">Your price:</font></b><font color="#ff0009"><b>
<%=FormatCurrency(RecordArray(5,i))%></b></font>
                </font></td>
                <td>
      <%if imageFileName <> "" then%>
         <a href="prod_det.asp?ProdID=<%=RecordArray(0,i)%>"><img
src="<%="admin/images/"&imageFileName%>"ALT="<%=
RecordArray(8,i)%>"height="30" width="30" border=0></a>
      <%end if%>
                </td>
         </tr>
         <tr>
                <td colspan="3" height="12">
      <hr width="95%" size="1" color="#EEEECC">
                </td>
         </tr>
<%next%>
<!--End search result table-->
  </table>
  </center>
  </table>
<!-- #INCLUDE file="footer.inc" -->
```

Figure 7. prod_det.asp

This screen is invoked once you click on the image or the hyperlink from the table on the page of search_results.asp. The image on the search_results.asp acts as a hotspot.

The code for this page is:

```
<!-- #INCLUDE file="header.inc" -->
<%
        Dim objProducts
        Set Rs = Server.CreateObject("ADODB.RECORDSET")
        Set RSAct = Server.CreateObject("ADODB.Recordset")
        Set Conn = Server.CreateObject("ADODB.Connection")
        conn.open GetConnectionString()

        SQLText = "select * from products where itemID = "&request("prodID")
        Rs.open SQLText,conn, adOpenStatic,adLockReadOnly

        'Get a list of actors/actressess in the movie.
        sSQL = "select actors.*, actordetails.itemid from actors, actordetails where
        actors.actorid = actordetails.actorid and itemid=" &
        Request.QueryString("prodID")
        RSAct.Open sSQL, conn, adOpenForwardOnly, adLockReadOnly
        'Response.Write "rsact = " & rsact.State
%>

<table bgColor="#bbbb99" border="0" width="480" cellspacing="1" cellpadding="4"
align="center">
  <tr>
    <td width="472" bgcolor="#EEEECC" colspan="2"><font face="arial, helvetica, sans-
serif" size="-1"><b>Movie
      Detail</b></font></td>
  </tr>
```

```
  <tr>
    <td width="86" bgcolor="#FFFFFF" valign="top" rowspan="2">
        <%if (rs("imageFileName") & "") <> "" then%>
                        <img src="<%="admin/images/"&rs("imageFileName")%>" <%if
rs("imageHeight") <> "" and rs("imageWidth") <> "" then%>
height="<%=rs("imageHeight")%>" width="<%=rs("imageWidth")%>" <%end if%>
border=0>
       <%end if%>
          </td>
    <td width="386" bgcolor="#FFFFFF" valign="top" align="left"><font face="arial,
helvetica, sans-serif" size="2">
    <% if trim(rs("SKU")) <> "" then%>
    <b>SKU</b>: <%=rs("SKU")%><br><%end if%>
    <b>Name: <%=rs("ItemName")%></b><br>
    <b>Synopsis:</b> <%=rs("ItemDesc")%><br>
    <b>Starring:</b>
    <%
    iCounter = 0

    While Not RSAct.EOF
                If iCounter>0 Then
                        Response.Write ", "
                End If
                Response.Write "<a href=""asearch.asp?actorid=" & RSAct("actorid") &
        Chr(34) & ">"
                Response.Write  rsact("firstname") & " " & rsact("lastname")
                Response.Write "</a>"
                iCounter = iCounter + 1
                RSAct.MoveNext
    Wend
    %>
    <br>
    <b><font color="#000000">Our Price</font>:
<%=FormatCurrency(rs("ItemPrice"))%></b><br>
    <% if trim(rs("ISBN")) <> "" then%>
    <b>ISBN</b>: <%=rs("ISBN")%>
     <%end if%>
     </font>
          </td>
  </tr>
  <tr>
    <td width="386" bgcolor="#FFFFFF" valign="top" align="left">
          <a href="cart.asp?ProdID=<%=rs("itemid")%>"><font face="arial, helvetica,
        sans-serif" size="2"><img border="0" src="images/addtocart.gif" WIDTH="80"
        HEIGHT="20"></font></a></td>
  </tr>
```

```
</table>
<%rs.Close
	Set rs = nothing

If rsAct.State = adStateOpen Then
	rsAct.Close
End If
Set rsAct = Nothing
%>
<!-- #INCLUDE file="footer.inc" -->
```

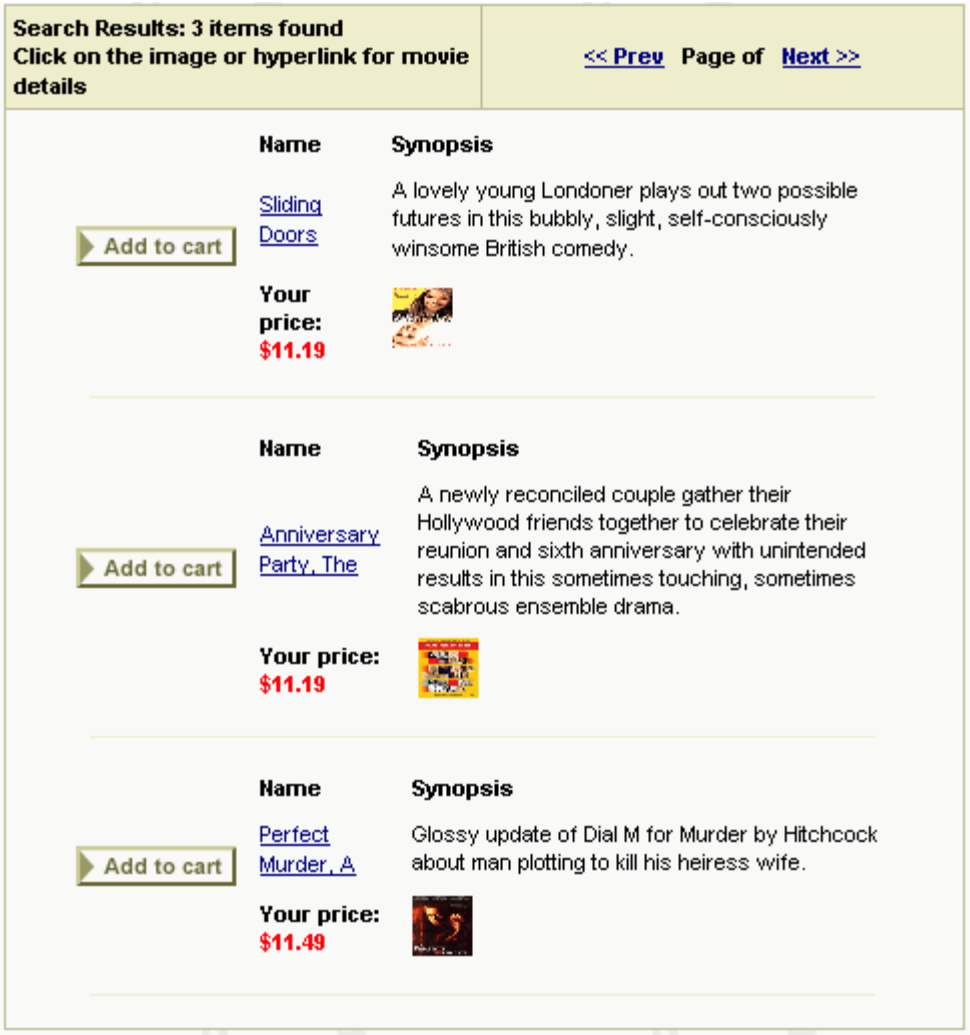


Figure 8. asearch.asp

From the prod_det.asp, with the click on the hyperlink of the actor, the user will be taken to the page of asearch.asp. On the page of asearch.asp, all movie(s) starring with the specified actor will be displayed. The code for this page is listed as below:

```
<%

    Set oConn = server.CreateObject("ADODB.Connection")
    Set oARS = Server.CreateObject("ADODB.Recordset")
    oConn.Open GetConnectionString()
sASQL = "select products.*, actordetails.itemid from products, actordetails where
actordetails.itemid = products.itemid and actordetails.actorid=" &
Request.QueryString("actorid")
    oARS.Open sASQL, oConn, adOpenKeyset, adLockOptimistic
```

```
        If oARS.BOF and oARS.EOF Then       'No records in recordset
                Response.Redirect("no_results.asp")
                Response.End
           Else
                iItemNumber = oARS("itemid")
                nRecs = oARS.RecordCount
%>
<!-- #INCLUDE file="header.inc" -->
<div align="center">
  <center>
  <table bgColor="#BBBB99" border="0"  width="480" cellspacing="1"
cellpadding="4">
    <tr>
      <td width="239" bgcolor="#EEEECC"><font face="arial, helvetica, sans-serif"
size="2"><b>Search
       Results:
 <%=nRecs%> items found</b></font></br>
 <font face="arial, helvetica, sans-serif" size="2"><b>Click on the image or hyperlink for
movie details</b></font>
 </td>
      <td width="239" bgcolor="#EEEECC"><font face="arial, helvetica, sans-serif"
size="-1">
<!--Start Tool bar table-->
      </font>
<table align="center" cellpadding="4" cellspacing="1" border="0"
bordercolor="#000000">
          <tr>
                <td><a href="search_results.asp?CatID=<%=CatID & "&Page=" &
(CurrentPage - 1)&"&ProdName="&ProdName%>"><font size="2" face="arial,
helvetica, sans-serif"><b>&lt;&lt;
        Prev</b></font></a></td>
                <td><font size="2" face="arial, helvetica, sans-serif"><b>Page
<%=CurrentPage%> of <%=nPages%>
        </b>
                </font></td>
                <td><a href="search_results.asp?CatID=<%=CatID & "&Page=" &
(CurrentPage + 1)&"&ProdName="&ProdName%>"><font size="2" face="arial,
helvetica, sans-serif"><b>Next
        &gt;&gt;</b>
                </font></a>
       </td>
          </tr>
</table>
       <font face="arial, helvetica, sans-serif" size="-1">
<!--End Tool bar table-->
      </font></td>
```

```
  </tr>
  <tr>
    <td width="472" bgcolor="#FFFFFF" colspan="2">

<!--Start search result table-->
<%
        While Not oARS.EOF
%>
<table align="center" width="90%" cellpadding="4" cellspacing="1" border="0"
bordercolor="#000000">
        <tr>
                <th align="left" rowspan="3">
      <table border="0" width="100%">
        <tr>
          <td width="100%">
          </td>
        </tr>
        <tr>
          <td width="100%" align="center">
      <a href="cart.asp?ProdID=<%=oARS("itemid")%>"><font face="arial, helvetica,
sans-serif" size="2"><img border="0" src="images/addtocart.gif" WIDTH="80"
HEIGHT="20"></font></a>
          </td>
        </tr>
      </table>
                </th>
                <th align="left" valign="bottom"><font size="2" face="arial, helvetica,
sans-serif">Name</font>
                </th>
                <th align="left" valign="bottom"><font size="2" face="arial, helvetica,
sans-serif">Synopsis</font>
                </th>
        </tr>
<tr>
                <td><font face="arial, helvetica, sans-serif" size="2"><a
href="prod_det.asp?ProdID=<%=oARS("itemid")%>"><%=oARS("itemname")%></a>
</font>
                </td>
                <td><font size="2"><font face="arial, helvetica, sans-
serif"><%=oARS("itemdesc")%></font></font>
                </td>
</tr>
        <tr>
                <td><font size="2" face="arial, helvetica, sans-serif"><b><font
color="#000000">Your price:</font></b><font color="#ff0009"><b>
<%=FormatCurrency((oARS("itemprice")&""))%></b></font>
```

```
                </font></td>
                <td>
        <%if  oARS("imagefilename") <> "" then%>
           <a href="prod_det.asp?ProdID=<%=oARS("itemid")%>"><img
src="<%="admin/images/" & oARS("imagefilename")%>"
ALT="<%=oARS("imagealttext")%>"height="30" width="30" border=0></a>
        <%end if%>
                </td>
           </tr>
           <tr>
                <td colspan="3" height="12">
        <hr width="95%" size="1" color="#EEEECC">
                </td>
           </tr>
<%'next
                        oARS.MoveNext
                Wend
           End if

If oARS.State = adStateOpen Then
           oARS.Close
End If
Set oARS = Nothing
%>
<!--End search result table-->
  </table>
  </center>
  </table>
<!-- #INCLUDE file="footer.inc" -->
```

Figure 9. cart.asp

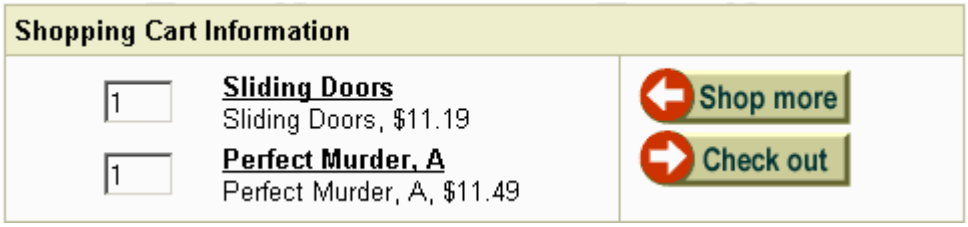


Figure 10. cart.asp

Once the "Add to Cart" button is clicked by the user, the cart.asp is triggered. On this screen, the user can enter the desired quantity of the selected movie(s). The user can either choose to shop more by clicking on the "Shop more" button, which takes him to the "home" page and repeat the shopping process or to click on the "Check out" button to continue and finish the shopping process. The code for this page is:

```
<%
Dim tmpQnt, i, numRows, ProdID, SessionID, NumItems, oCart, Cart
'Creat cart object and initiate session if not exist
Set oCart = New EStore_Cart
if session("SessionID") = "" then
        session("SessionID") = oCart.CreateSessionID()
end if
SessionID = session("SessionID")
'Response.Write "sessionid:"& sessionid
'assign value to prodID
If not IsEmpty(Request("ProdID")) then
        ProdID=clng(Request("ProdID"))
        oCart.AddToCart SessionID,ProdID
'       Response.Write "prod:"& Request("ProdID")
End if
Cart = oCart.GetCart(SessionID, NumItems)
'Response.Write "num:"&NumItems
'Response.End
If numitems = 0 then Response.Redirect("cart_empty.asp")
```

```
Redim QtyItems(50)
if Request("numRows") <> "" then
	numRows= clng(Request("numRows"))
	if numRows > 50 and numRows < 100 then  redim QtyItems(100)
	For i = 1 to numRows
		QtyItems(i-1) = Trim(Request(Request.Form.key(i)))
	Next
	oCart.UpdateCart SessionID, QtyItems
end if
Set oCart = Nothing
'If user pressed the continue, redirect to main page for more shopping
If request("continue.x") <> "" then Response.Redirect("default.asp")
'If user press the check out button, check if user has customer ID.
'If not redirect to user to get it, if exist redirect user to checkout.
if Request("checkout.x") <> "" then
	if Session("CustomerID") = "" then
		Response.Redirect "account_login.asp?checkout=yes"
	else
		Response.Redirect "checkout.asp"
	end if
end if
%>
<!-- #INCLUDE file="header.inc" -->
<!-- #INCLUDE file="html_templates/cart_frm.asp" -->
<!-- #INCLUDE file="footer.inc" -->
```

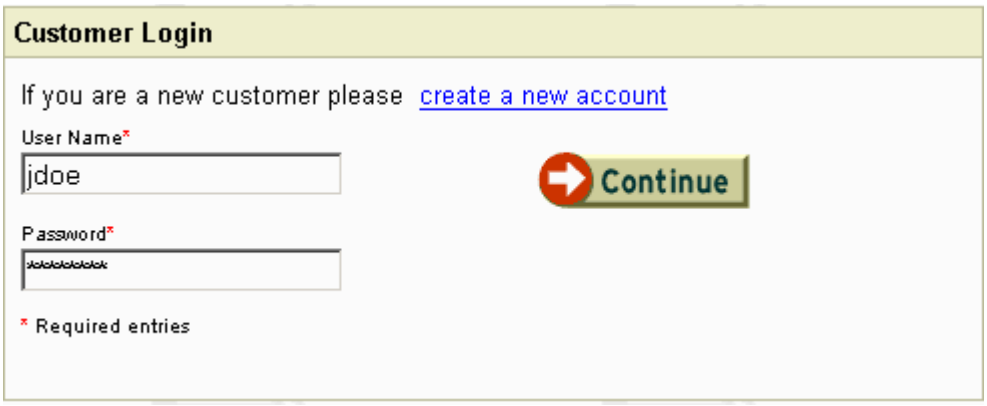


Figure 11. account_login.asp

Once the user clicks on the "Check out" button from the page of cart.asp, he/she will be directed to this page of account_login.asp.

For a registered user, once he/she types in the correct username, password and clicks on the "Continue" submit button, he/she continues with the check out process.

The code for this page is:

```
<%Response.Buffer = true%>
<!-- #INCLUDE file="header.inc" -->
<%checkout = request("checkout")
If Session("CustomerID") <> "" then
        Response.Redirect "account_form.asp?action=update"
End if
if Request.ServerVariables("REQUEST_METHOD") = "POST" then
        dim oForm, oAccount, MsgList(3), MsgCount, sUserID, sPassword, CustomerID
        UserName = Request("txtUserName")
        Password = Request("txtPassword")
        Set oForms = New EStore_Forms
    if oForms.VerifyUser(MsgList, MsgCount, UserName, Password, CustomerID) then
        Session("CustomerID") = CustomerID
        set oForms = nothing
        If request("checkout") = "yes" then
                Response.Redirect "checkout.asp"
        Else
                Response.Redirect "account_form.asp?action=update"
        End if
    Else
        oForms.WriteMsg MsgList
                Set oForms = Nothing
    End if      End if%>
```

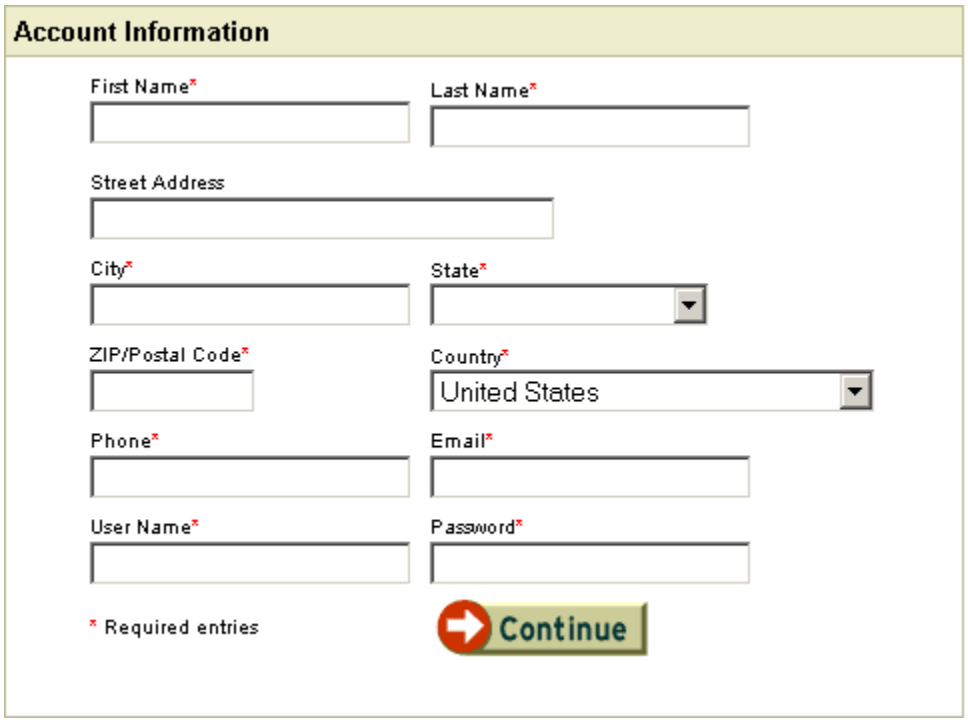


Figure 12. account_form.asp

For a new visitor, he/she is able to register by clicking on the "create a new account" link.

The code for the page is:

```
<%Response.Buffer = true%>
<!-- #INCLUDE file="header.inc" -->
<%
Dim MsgCount,MsgList(20), action, objSPM, oAccount, Cart, SessionID, CustomerID,
FirstName, LastName, Email, Phone, UserName, Password, CreditCard, ExpDate,
Address, City, State, Zip, CountryID, CountryName, NumItems
SessionID = Session("SessionID")
CustomerID = Session("CustomerID")
FirstName = cstr(Request.Form("txtFName"))
LastName = cstr(Request("txtLName"))
Address= cstr(Request("txtAddress"))
City =  cstr(Request("txtCity"))
State = cstr(Request("txtState"))
Zip = cstr(Request("txtZip"))
CountryID =  cstr(Request("txtCountry"))
Phone = cstr(Request("txtPhone"))
Email = cstr(Request("txtEmail"))
```

```
UserName =  cstr(Request("txtUserName"))
Password = cstr(Request("txtPassword"))
checkout = request("checkout")
If Request.QueryString("action") = "update" then
        Action = "update"
        'CustomerID = Session("CustomerID")
        'We are simply updating an existing account from within the site
        If IsEmpty(Request("txtFName")) then
        'First time to page, need to get current account information for this user
                Set oAccount = New EStore_Account
                Cart = oAccount.GetAccountInfo( CustomerID, FirstName,
                LastName, Address, City, State, Zip, CountryID, CountryName,
                Phone, Email, UserName, Password)
                Set oAccount = Nothing
        Else
                Set oForms = New EStore_Forms
                oForms.ValidateCustomerInfo  MsgList, FirstName, LastName,
                Address, City, State, Zip, CountryID, Phone, Email, UserName,
                Password, MsgCount,"update"
                        If MsgCount > 0 then
                                oForms.WriteMsg MsgList
                                Set oForms = Nothing
                                'action = "update"
                        else
                                'Update the Account Record
                                Set oAccount = New EStore_AccountTx
                                oAccount.UpdateAccount CustomerID, FirstName,
                                LastName, Address, City, State, Zip ,
                                CountryID, Phone, Email, UserName, Password
                                set oAccount = Nothing
                                        If request("checkout") = "yes" then
                                                Response.Redirect "checkout.asp"
                                        Else
                                                Response.Redirect "default.asp"
                                        End if
                        End if
        End if
Else
        If Request.QueryString("Action") = "insert" then
                Set oForms = New EStore_Forms
                oForms.ValidateCustomerInfo  MsgList, FirstName, LastName, Address,
                City, State, Zip , CountryID, Phone, Email, UserName,
                Password, MsgCount,"insert"
                        If MsgCount > 0 then
                                oForms.WriteMsg MsgList
                                Set oForms = Nothing
```

```
                    action = "insert"
                    Else
                        Set oAccount = New EStore_AccountTx
                        CustomerID = oAccount.CreateAccount(FirstName,
                        LastName, Address, City, State, Zip,
                        CountryID, Phone, Email, UserName, Password)
                        set oAccount = Nothing
                        Session("CustomerID") = CustomerID
                            If checkout = "yes" then
                                Response.Redirect "checkout.asp"
                            Else
                                Response.Redirect "default.asp"
                            End if
                    End If
                Else
                    action = "insert"
                End If
End If
%>
```

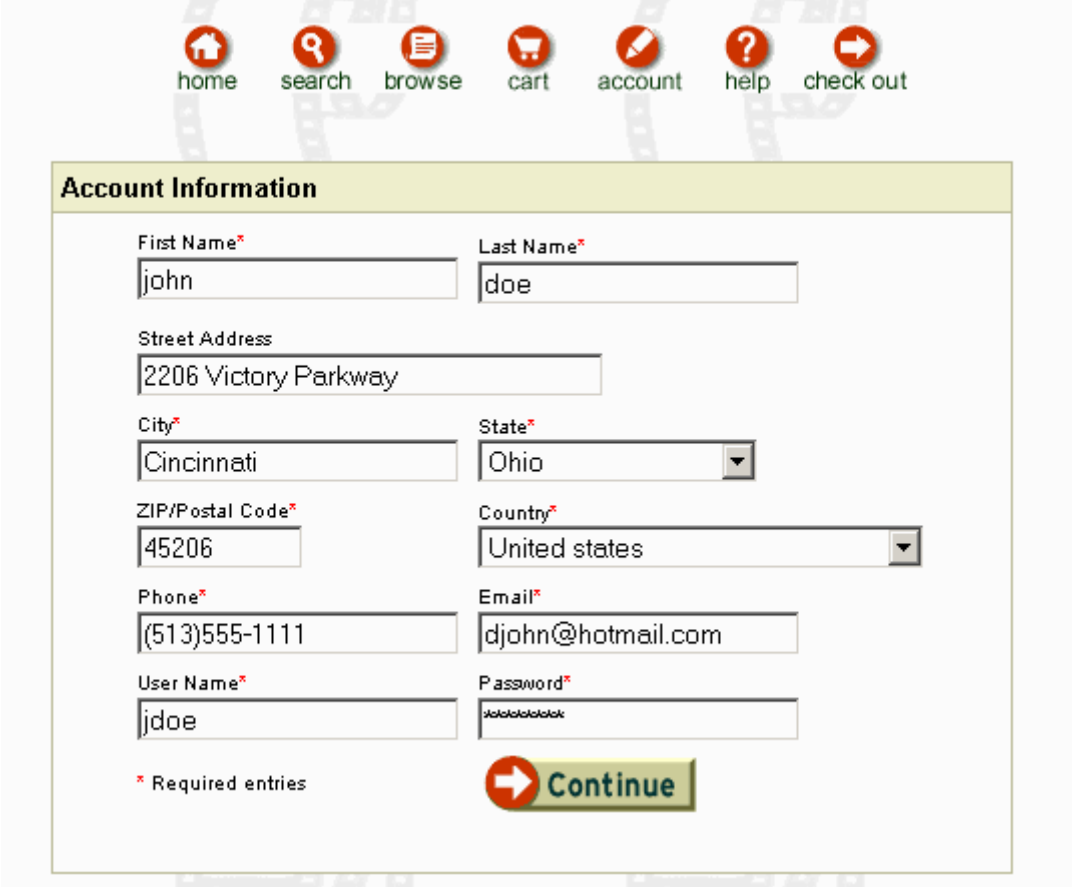


Figure 13. account_form.asp

The same code is also used for a registered user to update his/her information. After the change is made, the click on the "Continue" button validates the updated information.

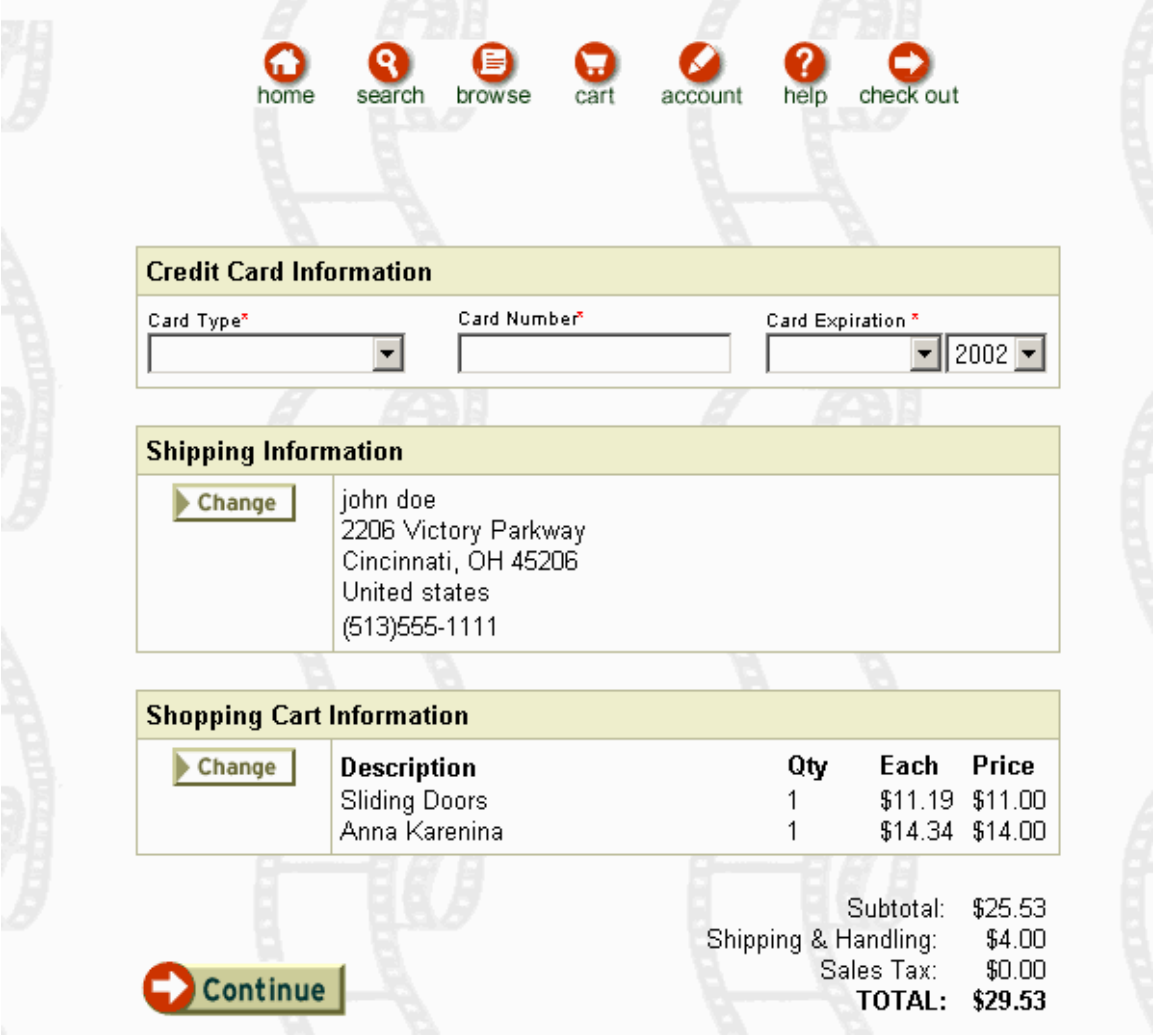


Figure 14. confirmation.asp

Before the final transaction is made, the customer has the option to change the Shipping Information, as well as the Shopping Cart Information.

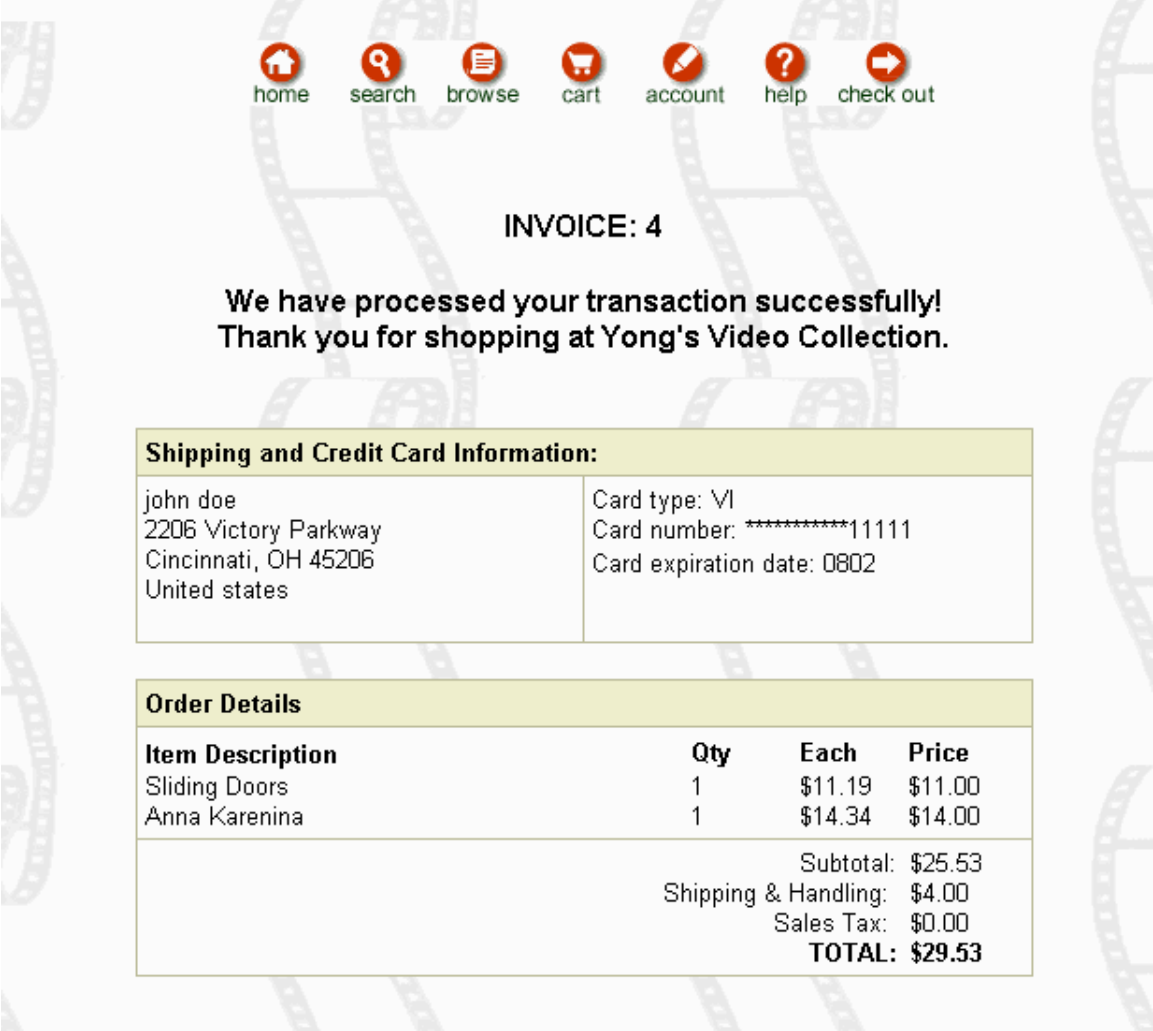


Figure 15. order complete

This is a screen showing how a completed order looks like.

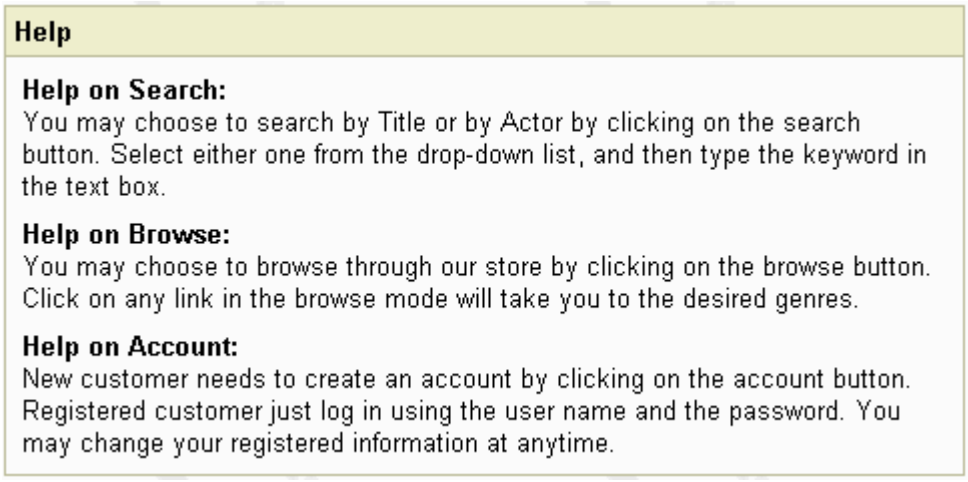


Figure 16. help.asp

This is a screen of the help.asp. This screen provides tips on how to perform movie searches, browse through genres, and manage the account.

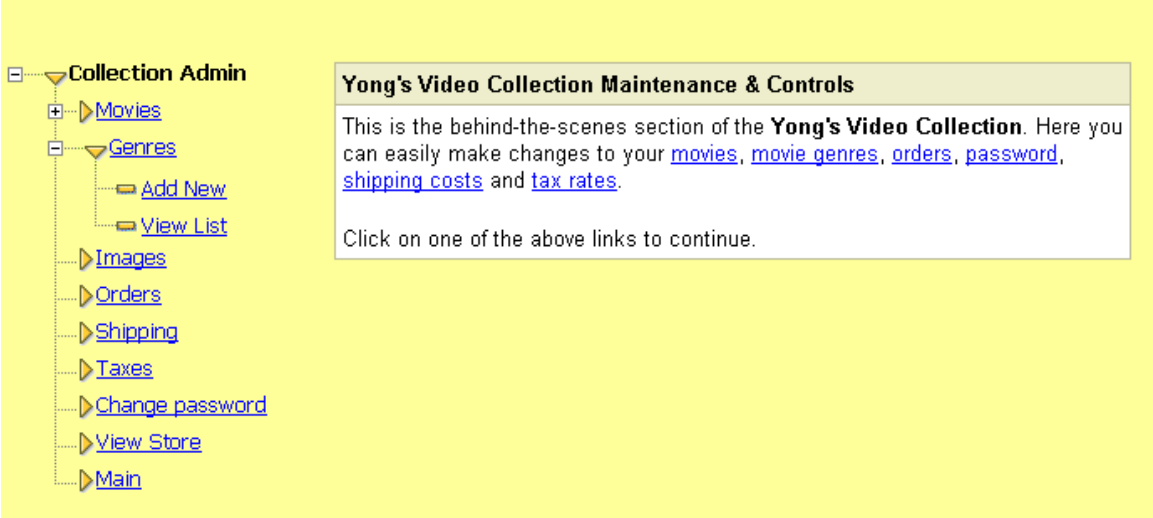


Figure 17. admin/default.asp

This page allows the authorized administrator of the database to log in and manage movie and purchase-related data. My project enables the store administrator to manage the movie data input, movie image upload, order tracking, shipping, tax rate and password modification in a rather different interface.

7. Conclusions and Recommendations

The development of my senior project was an ongoing learning process. I was able to apply what I have learned to my project. This includes the knowledge of Active Server Pages, Hyper Text Mark up Language, PaintShop Pro, Visio, and SQL.

It is of great importance that a leaner coordinates with the advisor throughout the development process of the senior design project. It is also very important for a learner to meet the deadline of the assignments at the different stages and don't wait till the last minute to do anything.

I seek help from all sources towards the third part of my senior design course in order to put my final project together.

As a learner, I would recommend others choosing the fields of interest and key strengths as the topics for their senior design projects. As a full time Information Engineering Technology student, it is not an easy task to concentrate in learning web programming intensively in just a few months.

Appendix A.

Special JAVA Scripts

This java script is used to create the tree structure on the administrator's page.

```
var Ccounter = 0;
var Mcounter = 1;
var firstNode = 0;
var lastNode = 0;
var emptyCat = new Array(); var ec = 1; //track empty categories
var currCat = 1, newCat;
function dopreload()
{
	var the_images = new
Array('../images/ftv2blank.gif','../images/ftv2lastnode.gif','../images/ftv2mlastnode
.gif','../images/ftv2mnode.gif','../images/ftv2node.gif','../images/ftv2plastnode.gif','
../images/ftv2pnode.gif','../images/ftv2vertline.gif','../images/ftv2pfirstnode.gif','../
images/ftv2mfirstnode.gif');
	preloadImages(the_images);
}

function preloadImages(the_images_array)
{
	for(loop = 0; loop < the_images_array.length; loop++)
	{
		var an_image = new Image();
		an_image.src = the_images_array[loop];
	}
}

function openTree(caption)
{
	document.write('<font face="Tahoma,Arial,Helvetica,Sans Serif"
	size="2">');
	document.write('<div style="margin-left:0; display:;">');
	document.write('<table cellpadding=0 cellspacing=0><tr>');
	document.write('<td valign=top><img src="../images/ftv2mfirstnode.gif"
	ID=MI-1 onClick="expandCollapseClick();">');
	document.write('<img ID=FI-1 src="../images/foldere2.gif"></td>');
	document.write('<td valign=top><span ID=MT-1><font size="-1"><b>');
	document.write(caption);
	document.write('</a></b></font></span></td>');
	document.writeln('</tr></table>');
}
```

```
function closeTree()
{
	document.writeln ('</DIV>\n</FONT>');
}

// This function handles to onClick event for expanding/closing
// nodes of the tree.
function expandCollapseClick ()
{
	var parentID;
	var child;
	var parentImage;
	var otherImage;

	parentID = window.event.srcElement.id;
	var imageNumber = parentID.substr(3);

	// used to detect which catagory is being clicked.
	if(imageNumber != '1')
		if(currCat != '1' && imageNumber != currCat){
			expandMenu("MI-" + currCat);
	}

	if(parentID.charAt (0) == 'M' && !emptyCat[parentID.substr(3)])
	{
		child = document.all ('C-' + imageNumber);
		parentImage = document.all ('MI-' + imageNumber);
		if (child.style.display == 'none') // hidden
		{
			otherImage = document.all ('FI-' + imageNumber);
			otherImage.src = '../images/foldere2.gif';
			child.style.display = '';

			// used only if tree root is not clicked.
			if(imageNumber != '1') currCat = imageNumber;

			/* use last minus sign when appropiate - determine which
			type of node */

			switch(parentID) {
				case 'MI-1' :
					parentImage.src =
'../images/ftv2mfirstnode.gif';
					break;
				case ('MI-' + Mcounter) :
```

```
                                        parentImage.src =
                                '../images/ftv2mlastnode.gif';
                                        break;
                                default :
                                        parentImage.src = '../images/ftv2mnode.gif';
                                        break;
                        }

                }
                else
                {
                        otherImage = document.all ('FI-' + imageNumber);
                        otherImage.src = '../images/folderc2.gif';

                        // used only if tree root is not clicked.
                        if(imageNumber != '1') currCat = '1';

                        child.style.display = 'none';
                        switch(parentID) {
                                case 'MI-1' :
                                        parentImage.src =
                                '../images/ftv2pfirstnode.gif';
                                        break;
                                case ('MI-' + Mcounter) :
                                        parentImage.src =
                                '../images/ftv2plastnode.gif';
                                        break;
                                default :
                                        parentImage.src = '../images/ftv2pnode.gif';
                                        break;
                        }
                }
        }
}

// This function gets called from a script tag at the bottom of
// your html to expand the menu upon page opening.
function expandMenu(cat)
{
        var parentID;
        var child;
        var parentImage;
        var otherImage;

        // Get First level desired to expand. You want to specify
        // Top level (MI-1) when calling this function.
```

```
parentID = cat.substr(0);

if (parentID.charAt(0) == 'M' && !emptyCat[parentID.substr(3)])
{
	child = document.all ('C' + parentID.substr (2));
	parentImage = document.all ('MI' + parentID.substr (2));
	if (child.style.display == 'none') // hidden
	{
		otherImage = document.all ('FI' + parentID.substr (2));
		otherImage.src = '../images/foldere2.gif';
		child.style.display = '';
		/* use last minus sign when appropiate - determine which
		type of node */
		switch(parentID) {
			case 'MI-1' :
				parentImage.src =
			'../images/ftv2mfirstnode.gif';
				break;
			case ('MI-' + Mcounter) :
				parentImage.src =
			'../images/ftv2mlastnode.gif';
				break;
			default :
				parentImage.src = '../images/ftv2mnode.gif';
				break;
		}
	}
	else
	{
		otherImage = document.all ('FI' + parentID.substr (2));
		otherImage.src = '../images/folderc2.gif';
		child.style.display = 'none';
		switch(parentID) {
			case 'MI-1' :
				parentImage.src =
			'../images/ftv2pfirstnode.gif';
				break;
			case ('MI-' + Mcounter) :
				parentImage.src =
			'../images/ftv2plastnode.gif';
				break;
			default :
				parentImage.src = '../images/ftv2pnode.gif';
				break;
		}
	}
```

```
        }
}

// This function creates the link.
function makeLink (category, urlDir, target)
{
        var categoryToDisplay, targetstr;

        if (category.length > 12)
        {
                categoryToDisplay = category.substring (0, 50);
        }
        else
        {
                categoryToDisplay = category;
        }

        HTMLstr='<font size="-1"><A HREF="' + urlDir + '" TITLE="Open
        &#145;' + category + '&#146;." class="Corp3"';
        targetstr = target + ""; // Explicitely change target to string

        if (targetstr != "") // Test to see if target is empty, if not empty add the
        target attribute to href tag
        {
                HTMLstr += 'target="' + target +'">' + categoryToDisplay +
        '</A></font>';
        }
        else // else leave tag alone.
        {
                HTMLstr += '>' + categoryToDisplay + '</A></font>';
        }
        return HTMLstr;
}

// This function creates the end nodes under any given category.
function makeNode (name, URLpath, target)
{
        ++lastNode; // increment to track last node of node division.
        document.writeln ('<table CELLPADDING=0
        CELLSPACING=0><tr><td valign=top><img ID=img1-' + lastNode + '
        src="../images/ftv2vertline.gif"><img ID=img2-' + lastNode + '
        src="../images/ftv2node.gif"><img src="../images/doc.gif"></td><td
        width="100" valign=middle>' + makeLink(name, URLpath, target) +
        '</td></tr></table>');
}
```

```
// This function creates the category.
function makeCategory (name, URLpath, target)
{
Mcounter++;
document.writeln ('<table CELLPADDING=0 CELLSPACING=0><tr><td
valign=top><img src="../images/ftv2pnode.gif" ID=MI-' + Mcounter + '
onClick=expandCollapseClick()><img ID=FI-' + Mcounter + '
src="../images/folderc2.gif"></td><td valign=top><SPAN ID=MT-' + Mcounter
+ '>' + makeLink (name, URLpath, target) + '</SPAN></td></tr></table>');
}

// This function creates the opening div tag for hiding and
// showing with the expandCollapseClick function.
function openDiv ()
{
        emptyCat[ec] = false;
        Ccounter++;
        //Changed margin from 15 to 24 for new images
        document.writeln ('<DIV ID=C-' + Ccounter + ' STYLE="margin-left:24;
        display:None;">');
}
function openNodeDiv ()
{
        firstNode = lastNode + 1; // remember first node of node division.
        ++Ccounter;
        document.writeln ('<DIV ID=C-' + Ccounter + ' STYLE="margin-left:0;
        display:None;">');
}

// This function closes the div tag.
function closeDiv ()
{
        var image;

        document.writeln ('</DIV>');

        // Choose correct image for last catagory
        image = document.all('MI-' + Mcounter);

        // if last catagory is empty no plus sign else plus sign
        if(emptyCat[ec])
                image.src = '../images/ftv2lastnode.gif';
        else
                image.src = '../images/ftv2plastnode.gif';

        // blank last catagory's node's first image(vertical line).
```

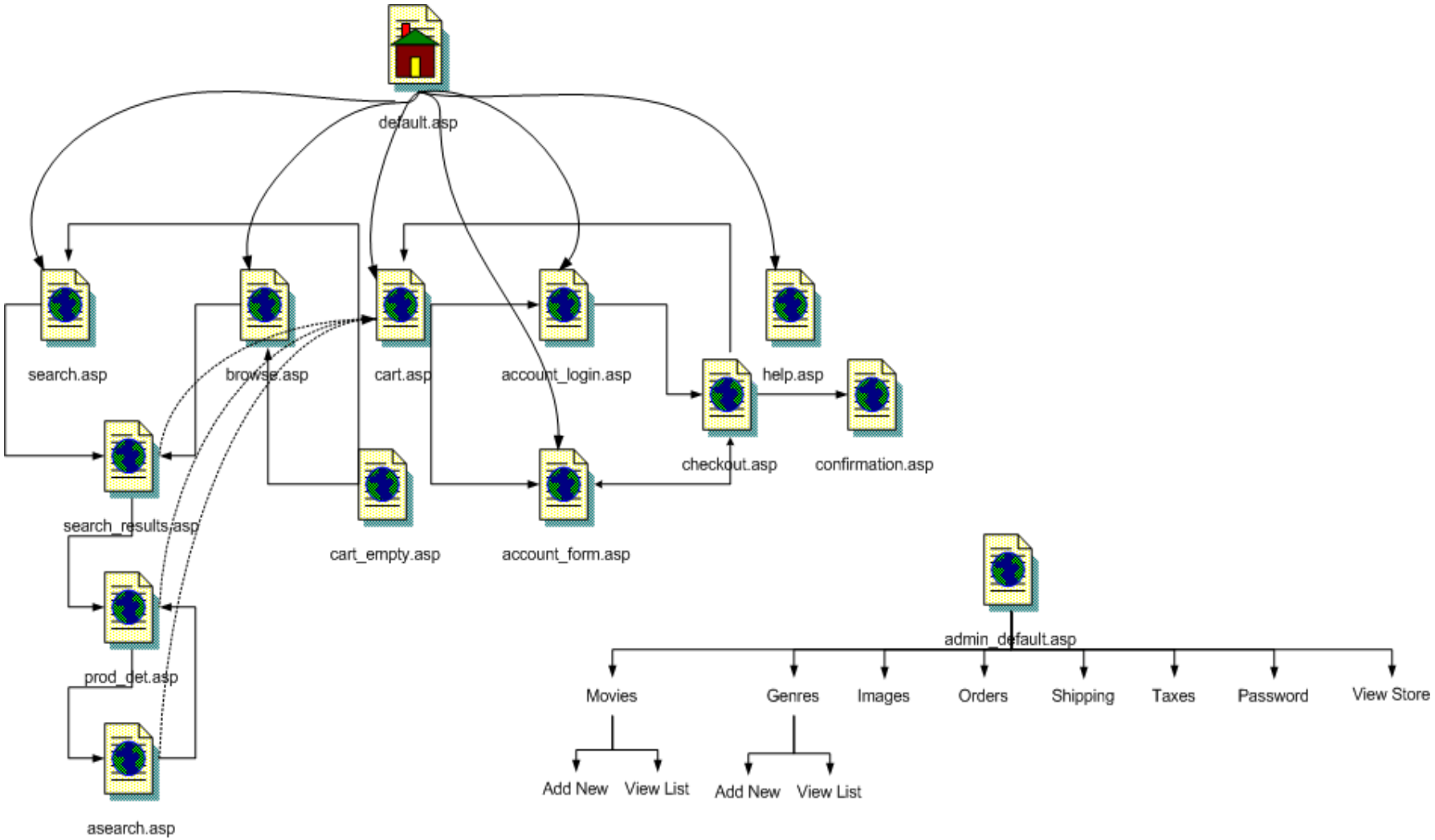
```
        for(i = firstNode; i <= lastNode; ++i)
        {
                image = document.all('img1-' + i);
                //Added width and height for proper alignment - should be OK
                image.width = '24';
                image.height = '0';
                image.src = '../images/ftv2blank.gif';
        }
}
        function closeNodeDiv ()
{
        var image;

        document.writeln('</DIV>');

        // if catagory empty(no nodes) then no plus sign else set last node image
        if(firstNode == lastNode + 1){
                emptyCat[++ec] = true;
                image = document.all('MI-' + ec);
                image.src = '../images/ftv2node.gif';
        }
        else{
                emptyCat[++ec] = false;
                image = document.all('img2-' + lastNode);
                image.src = '../images/ftv2lastnode.gif';
        }
}
```

Appendix B.

Hardware and Software Costs

Web Video Store Budget

Hardware:	
17" Monitor (15)	$116.00
Intel Pentium 4 Processor 1.4GHz (15)	350.00
80GB Ultra DMA HDD (15)	238.00
128MB PC 600 RDRAM (15)	118.00
Software:	
Windows 2000 Professional (10)	319.00
Windows Me (10)	209.00
Windows NT Server Enterprise Edition (10)	3999.00
SQL Server 2000 (10)	4999.00
Visual Studio (10)	1619.00
Jasc Paint Shop Pro 7 (9)	109.00
Visio Pro (10)	399.00
MS IE 5.5 (10)	Free
Total	**$12475.00**



Appendix C.

Site Diagram

References

1. Brackett, Carolyn. “Setting Up Shop in Cyberspace”. First Data Merchant Services Corporation. http://www.ecommercetimes.com/small_business/getting_started/firstdata.shtml

2. Caswell, Stephen. “Study: U.S. Manufacturers Not B2B E-commerce Ready”. http://www.ecommercetimes.com/perl/printer/2561. February 23,2000

3. Davis, Jessica. “Dot-Com Casualty List Mounts as Holiday Season Approaches”. *INFOWORLD*. November 20, 2000. 32.

4. Fonseca, Brian. “Biometrics Eye the Mainstream Markets”. *INFOWORLD*. January 15, 2001. 32.

5. Geller, David. “The Seven Deadly Sins of E-commerce”. E-commerce Times. http://www.ecommercetimes.com/small_business/getting_started/deadly-sins-a.shtml

6. Grygo, Eugene. “Easier Integration for B-to-B”. *INFOWORLD*. November 13, 2000. 8.

7. Grygo, Eugene. “Firms Push B-to-B Basics”. *INFOWORLD*. December 11, 2000. 8.

8. Hsu, Victor. “Web-based Commerce Auto Parts Store.” March 2000.

9. Jasc Software. http://www.jasc.com/product.asp?pf%5Fid=001 February 26, 2001.

10. Microsoft Corporation. “The Product and Technology Catalog.” http://www.microsoft.com/catalog/default.asp?subid=22. March 4, 2001.

11. Miller, Michael J. “E-Business Essentials”. *PC Magazine*. July 2000.132-136.

12. Neel, Dan. “Start-ups Benefit From High-Tech Support”. *INFOWORLD*. November 6, 2000. 41.

13. Rash, Wayne. “When E-Business Became Real Business.” *Internet Week*. December 11, 2000. 32.

14. Shah, Priya. “Online Bookstore.” March 2000.

15. Street Price Search Engine. http://www.pricewatch.com/. March 4, 2001.

16. Walther, Stephen and Levine, Jonathan. “Sams Teach Yourself E-commerce Programming with ASP in 21 Days”. Indianapolis: SAMS, A Division of Macmillan USA, 2000.

17. “E-commerce Revenue Forecasts.” *Internet Week*. Special Year-End Issue. December 18, 2000. 32.

18. “Planning for the Perfect Online Store”. http://www.ecommercetimes.com/small_business/strategy/story-2-a.shtml

19. “Privacy”. *INFOWORLD*. December 4, 2000. 29.

20. “Where the Web Development Budget Went”. *INFOWORLD*. December 25, 2000/January 1, 2001. 12.

21. “Which Products Sell Best Online?” E-commerce Times. http://www.ecommercetimes.com/small_business/strategy/story-1-a.shtml