

ONLINE SHOPPING

By

CHAITANYA REDDY MITTAPELLI

B.E., Osmania University, 2005

A REPORT

Submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computing and Information Sciences
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2008

Approved By:

Major Professor
DANIEL ANDRESEN Ph.D.

ABSTRACT

The Online Shopping is a web based application intended for online retailers. The main objective of this application is to make it interactive and its ease of use. It would make searching, viewing and selection of a product easier. It contains a sophisticated search engine for user's to search for products specific to their needs. The search engine provides an easy and convenient way to search for products where a user can Search for a product interactively and the search engine would refine the products available based on the user's input. The user can then view the complete specification of each product. They can also view the product reviews and also write their own reviews. The application also provides a drag and drop feature so that a user can add a product to the shopping cart by dragging the item in to the shopping cart. The main emphasis lies in providing a user-friendly search engine for effectively showing the desired results and its drag and drop behavior.

Table of Contents

Acknowledgements.....	v
1. Introduction.....	1
1.1 Goal.....	1
1.2 Need of the application.....	1
1.3 Scope.....	2
1.4 Platform Specifications – Deployment.....	2
1.4.1 Hardware Specification.....	2
1.4.2 Software Specification.....	3
2. System Requirement Analysis.....	3
2.1 Information Gathering.....	3
2.2 System Feasibility.....	4
2.2.1 Economic Feasibility.....	4
2.2.2 Technical Feasibility.....	4
2.2.3 Behavioral Feasibility.....	4
3. System Analysis.....	5
3.1 ER Diagram.....	5
3.2 Data Flow Diagram.....	6
3.3 Use case Diagram.....	10
3.4 Class Diagram.....	11
4. Design.....	11
4.1 Design Goals.....	11
4.2 Architectural Design.....	12
4.2.1 Architectural Context Diagram.....	12
4.2.2 Description of Architectural Design.....	12
4.3 Procedural/Modular Approach.....	13
4.3.1 Shop Products Module.....	13
4.3.2 Product Description Module.....	13
4.3.3 Cart Details Module.....	13
5. Implementation.....	14
5.1 Database Design and Implementation.....	14
5.2 User Interface Design and Implementation.....	15
5.3 Technical Discussions.....	19
6. Testing.....	19
6.1 Unit Testing.....	19
6.2 Integration Testing.....	20
6.3 Validation Testing.....	21
6.4 White Box Testing.....	22
6.5 Performance Testing.....	22
7. Results & Challenges.....	24
7.1 Challenges.....	25

8. Conclusions.....	25
8.1 Limitations	25
8.2 Scope for Future Work.....	26
9. References.....	27

Acknowledgements

I would like to thank my major professor Dr. Daniel Andresen for his constant guidance and help throughout the project. I would also like to thank Dr. Torben Amtoft and Dr. Mitchell Neilsen for graciously accepting to be on my committee.

Finally, I would like to thank my family and my friends for all the support and encouragement.

1. Introduction

1.1 Goal

Shopping has long been considered a recreational activity by many. Shopping online is no exception. The goal of this application is to develop a web based interface for online retailers. The system would be easy to use and hence make the shopping experience pleasant for the users. The goal of this application is

- To develop an easy to use web based interface where users can search for products, view a complete description of the products and order the products.
- A search engine that provides an easy and convenient way to search for products specific to their needs. The search engine would list a set of products based on the search term and the user can further filter the list based on various parameters.
- An AJAX enabled website with the latest AJAX controls giving attractive and interactive look to the web pages and prevents the annoying post backs.
- Drag and Drop feature which would allow the users to add a product to or remove a product from the shopping cart by dragging the product in to the shopping cart or out of the shopping cart.
- A user can view the complete specification of the product along with various images and also view the customer reviews of the product. They can also write their own reviews.

1.2 Need of the application

There are large numbers of commercial Online Shopping websites offering large number of products tailored to meet the shopping interests of large number of customers. These online marketplaces have thousands of products listed under various categories.

Problem:

- The basic problems with the existing systems are the non-interactive environment they provide to the users.
- The use of traditional user interfaces which make continuous post backs to the server; each post back makes a call to the server, gets the response and then refreshes the entire web form to display the result. This scenario adds an extra trade off causing a delay in displaying the results
- A search engine that would display the results without allowing the users to further filter the results based on various parameters.
- Use of traditional and non user friendly interfaces that are hard to use

Solution:

- The motive of this Online Shopping Web Application is to allow the user to play with the search tool and create different combinatorial search criterion to perform exhaustive search.
- Making the application AJAX enabled gets rid of these unnecessary delays letting the user to perform exhaustive search. The users of this application can easily feel the difference between the Ajax empowered user interfaces vs. traditional user interfaces.
- Provide Interactive interface through which a user can interact with different areas of application easily.
- A search engine that provides an easy and convenient way to search for products specific to their needs. The search engine would list a set of products based on the search term and the user can further filter the list based on various parameters.
- Provide Drag and Drop feature thereby allowing the user to add products to or remove products from the shopping cart by dragging the products in to or out of the shopping cart.

1.3 Scope

- The current system can be extended to allow the users to create accounts and save products in to wish list.
- The users could subscribe for price alerts which would enable them to receive messages when price for products fall below a particular level.
- The current system is confined only to the shopping cart process. It can be extended to have a easy to use check out process.
- Users can have multiple shipping and billing information saved. During checkout they can use the drag and drop feature to select shipping and billing information.

1.4 Platform Specifications – Deployment**1.4.1 Hardware Specification**

Processor P IV

RAM 250 MB

Minimum Space Required 100 MB

Display 16 bit color

1.4.2 Software Specification

Operating Environment Win 2000/XP

Platform .Net Framework & IIS Visual Studio 2008

Database SQL Server 2005

2. System Requirement Analysis

2.1 Information Gathering

As the goal of the application is ease of use and to provide an interactive interface, extensive research has been done to gain an insight into the needs and behaviors of various users. The working of the application is made convenient and easy to use for the end user. Dr Andresen, Associate Professor, CIS provided regular feedback on the project.

Users can be classified into two types based on their knowledge of the products that suit their needs. They can be classified as users who know about the product that would satisfy their needs and users who have to figure out the product that would satisfy their needs. Users who know about the product should be able to find the product easily with the click of a button. Such users can search for the product by using the product name as the search term. Users who have to figure out the product that would satisfy their needs could use a search term to find a list of products and then should be able to filter the results based on various parameters like product type, manufacturer, price range, platform supported etc.

The users should be able to view the complete specification of the product and various images at different Zoom levels. The user should be able to read the customer reviews for the product and the ratings provided. They should be able to write their own reviews. They should be able to print out the specifications for a product or email the product page to a friends etc.

To increase the ease of use the user should be able to add a product to the shopping cart by dragging a product and dropping it in the shopping cart. A user should be able to edit the contents of a shopping cart. They should be able to update the quantities of the products added to the cart and remove the products from the cart. The user should be able to remove the product from the shopping cart by dragging the product and dropping it outside the cart.

The application can be made interactive by pop up messages when a product has been dropped in to the shopping cart or out of the shopping cart. The user can be notified

if the cursor enters a drop area and the object that could be dropped. Also users are impatient making it important to load pages soon.

Other than this, I did a lot of research on various other methods of building this application which and was able to incorporate a few stronger features into the application. The tools and controls used in the application are recommended ASP.NET controls and AJAX Toolkit controls which improves the navigation and usability and interactivity.

2.2 System Feasibility

The system feasibility can be divided into the following sections:

2.2.1 Economic Feasibility

The project is economically feasible as the only cost involved is having a computer with the minimum requirements mentioned earlier. For the users to access the application, the only cost involved will be in getting access to the Internet.

2.2.2 Technical Feasibility

To deploy the application, the only technical aspects needed are mentioned below:

Operating Environment Win 2000/XP

Platform .Net Framework & IIS

Database SQL Server 2005

For Users:

Internet Browser

Internet Connection

2.2.3 Behavioral Feasibility

The application requires no special technical guidance and all the views available in the application are self explanatory. The users are well guided with warning and failure messages for all the actions taken.

3. System Analysis

After carefully analyzing the requirements and functionality of the web application, I had two important diagrams by the end of the analysis phase. They are the ER diagram and data flow diagram which were the basis for finding out entities and relationships between them, the flow of information.

3.1 ER Diagram

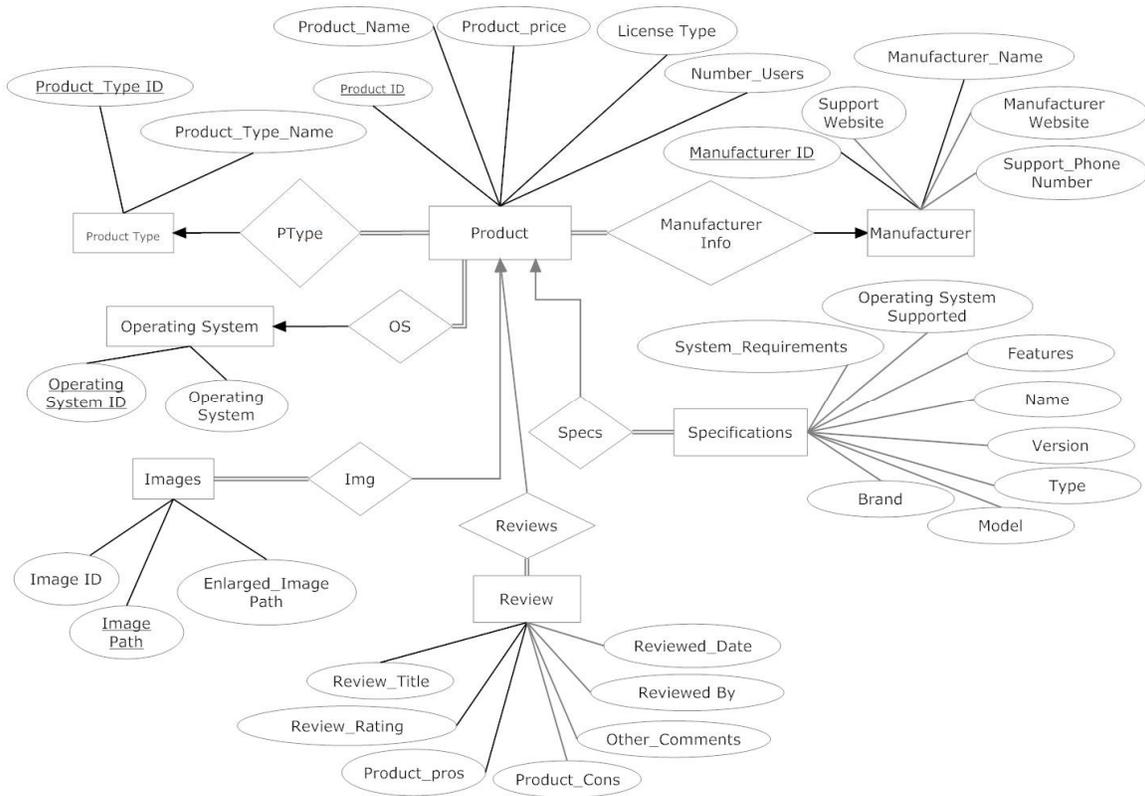


Figure 3.1 Entity Relation Ship Diagram

3.2 Data Flow Diagram

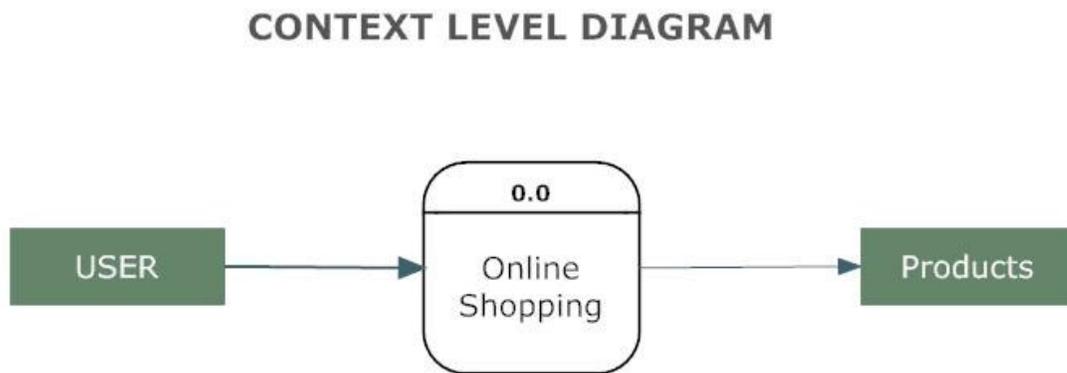


Figure 3.2: A Context Level Diagram

First Level DFD

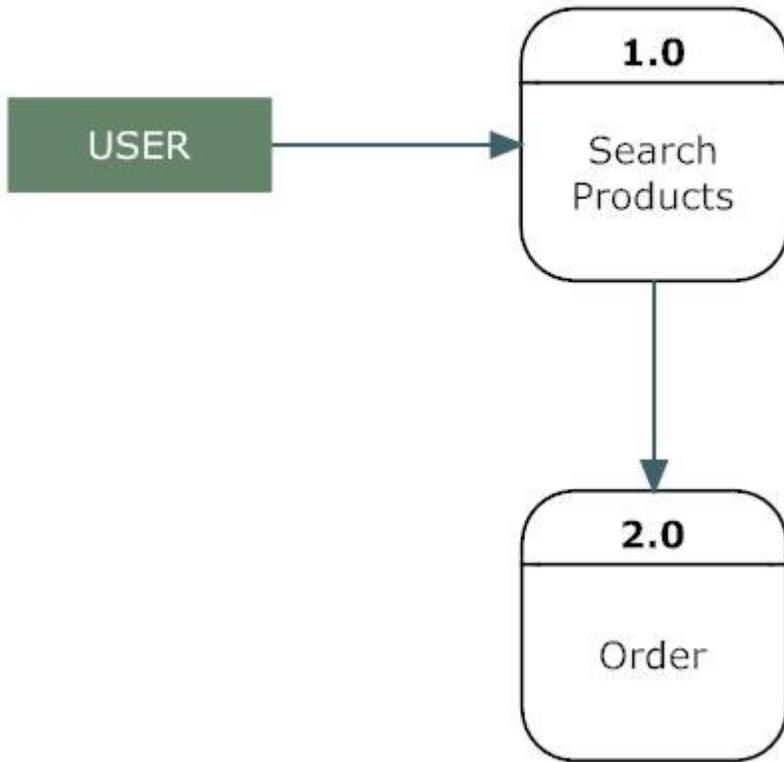


Figure 3.3: A First Level Diagram

SECOND LEVEL DFD

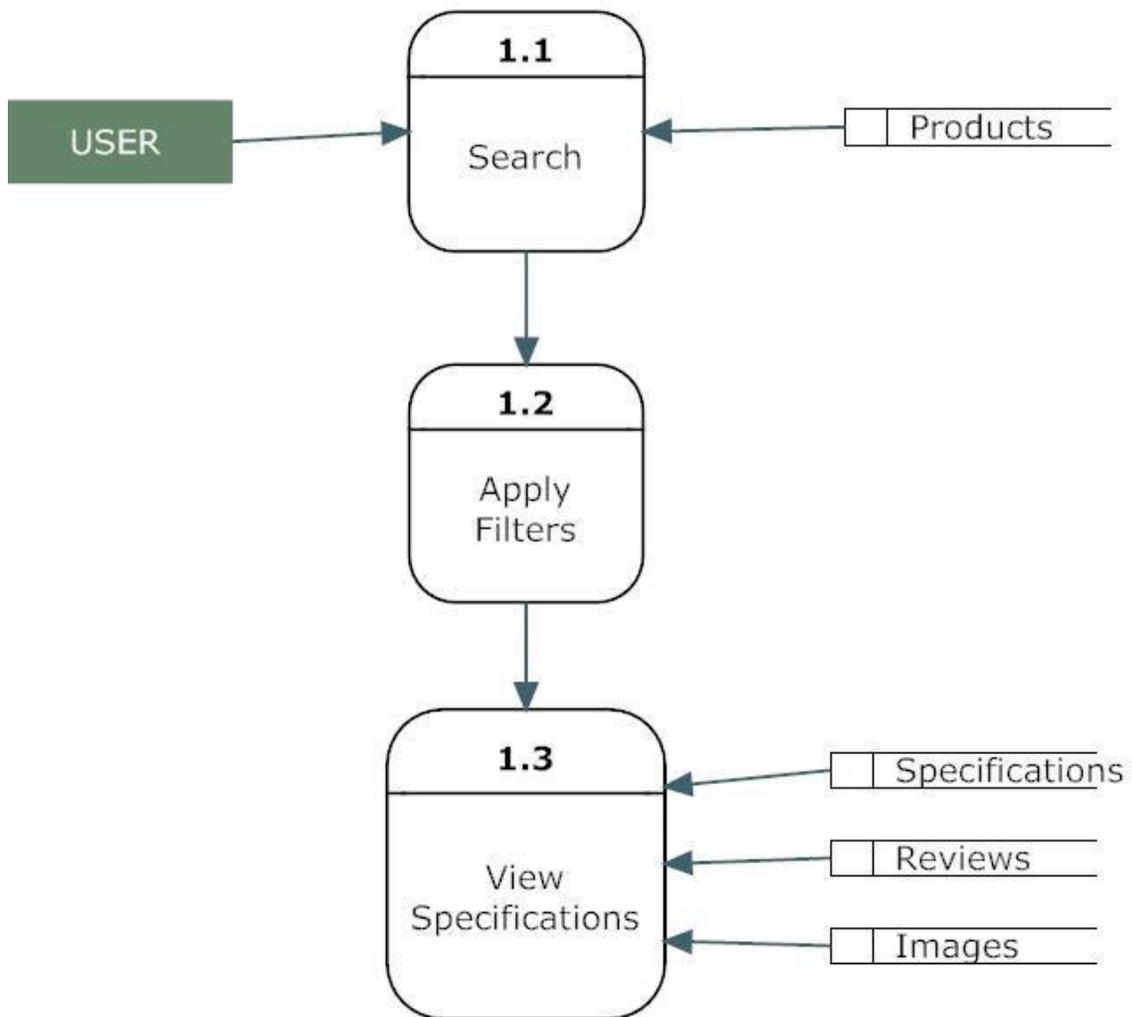


Figure 3.4: A Second Level Diagram

SECOND LEVEL DFD

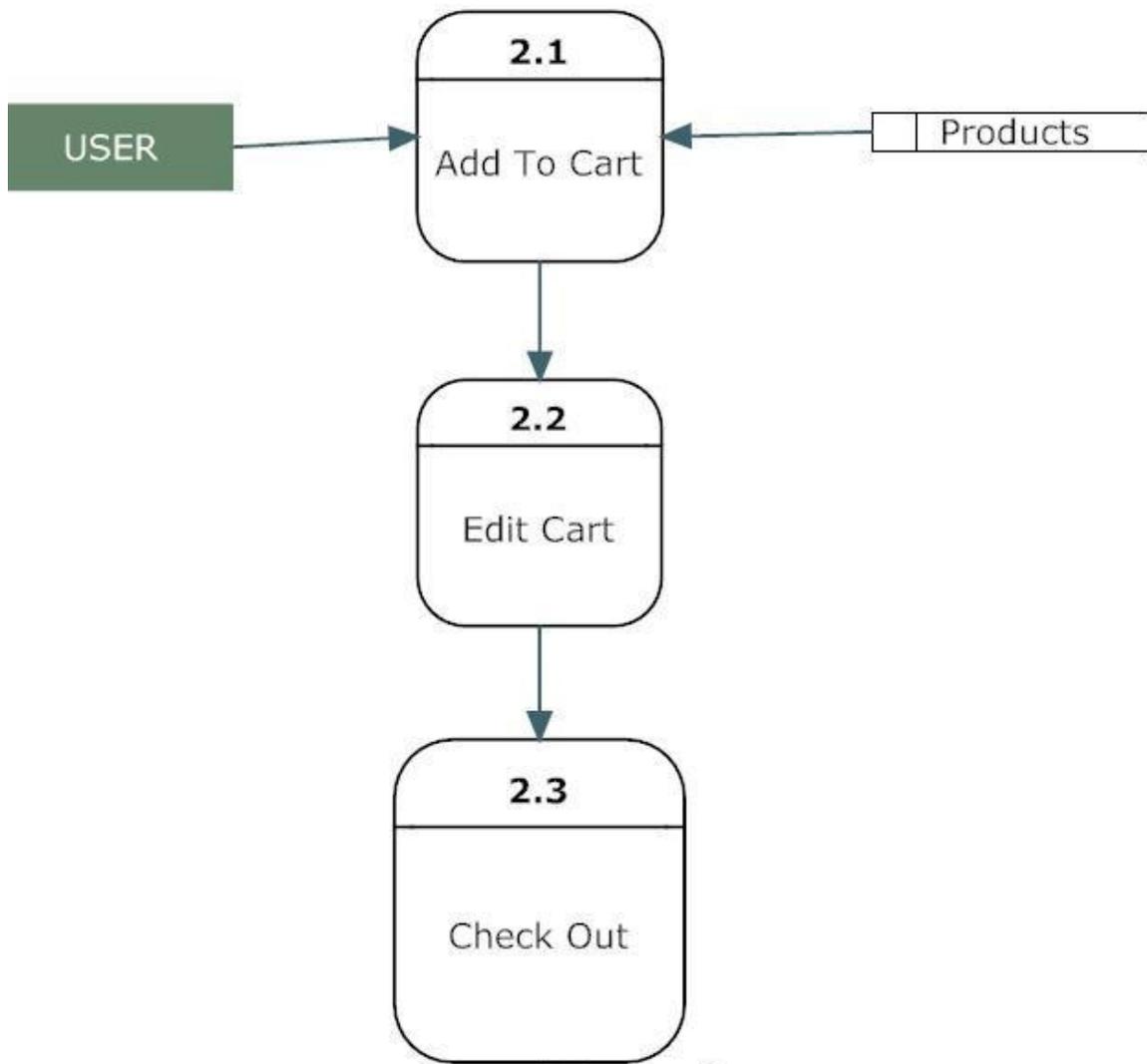


Figure 3.5: A Second Level Diagram

3.3 Use Case Diagram

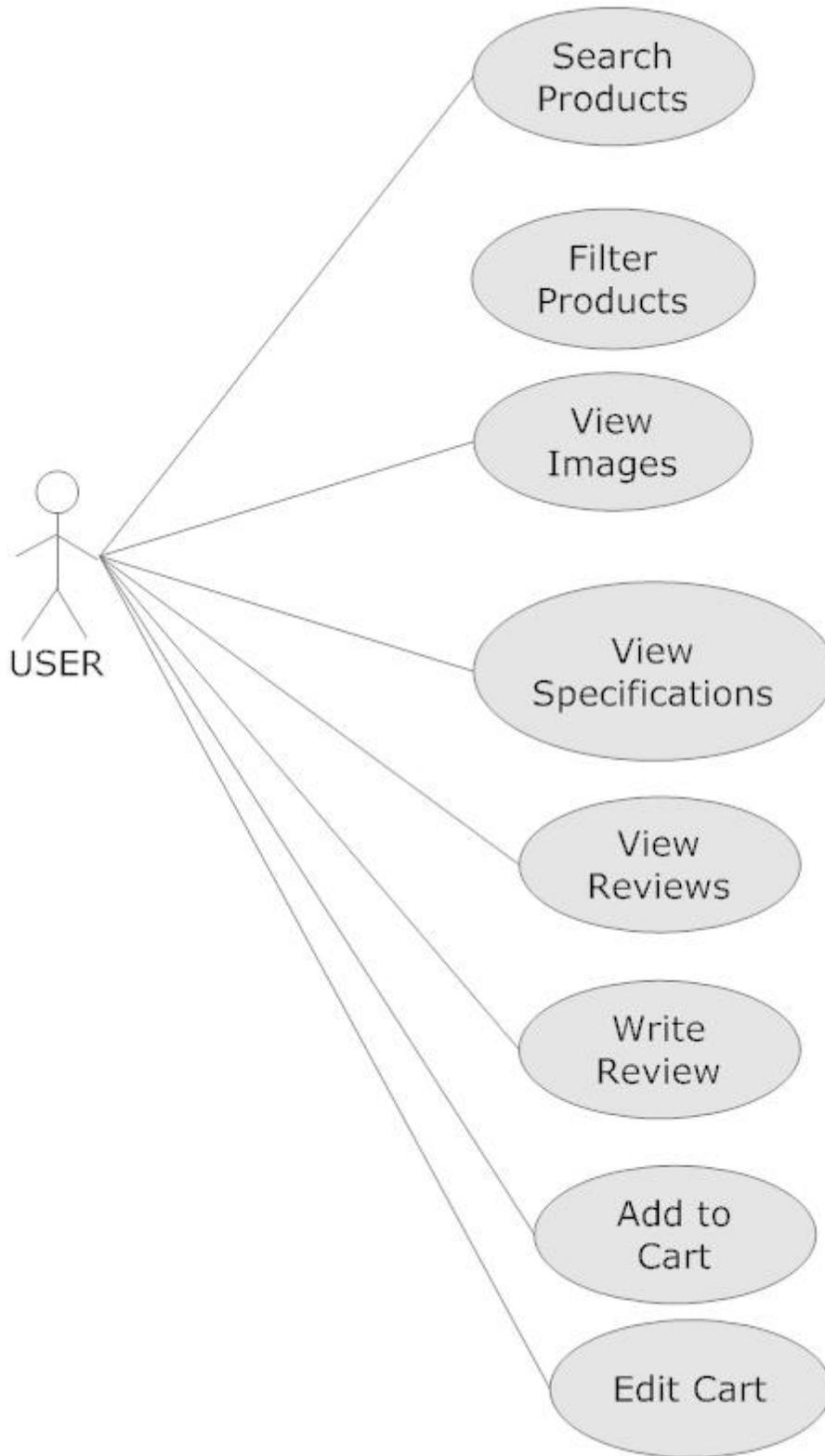
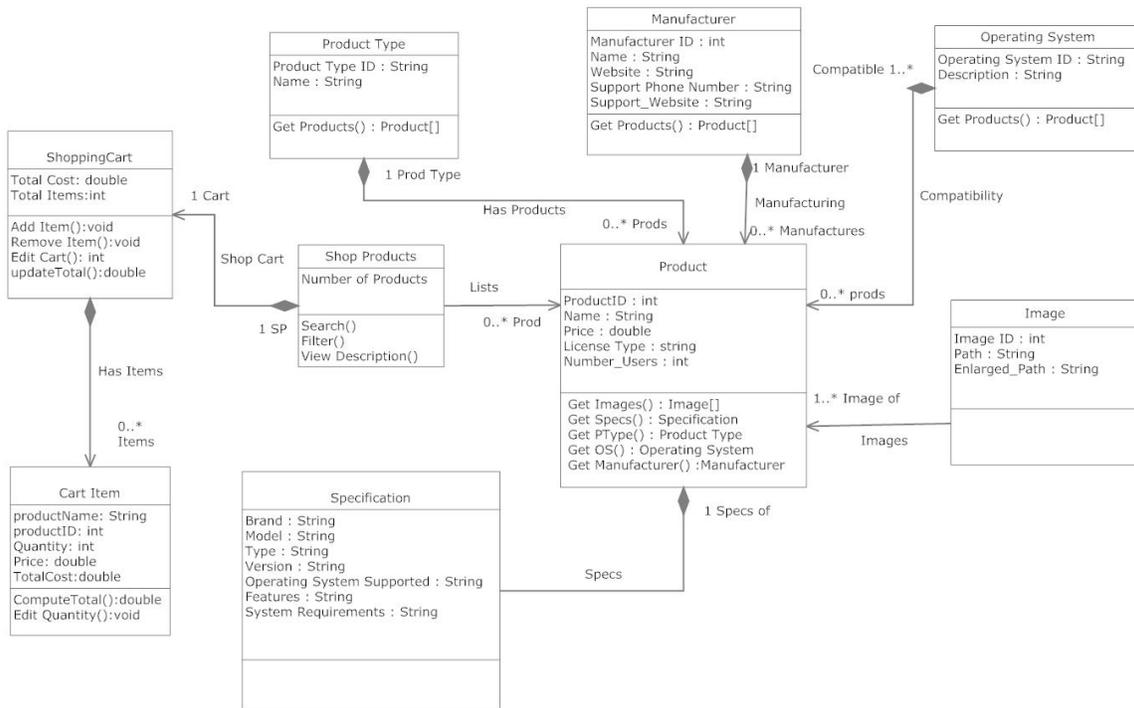


Figure 3.6: Use Case Diagram

3.4 Class Diagram



CLASS DIAGRAM

Figure 3.7: Class Diagram

4. Design

4.1 Design Goals

- The design of the web application involves the design of the forms for listing the products, search for products, display the complete specification for the product, and design a shopping cart that is easy to use.
- Design of an interactive application that enables the user to filter the products based on different parameters.
- Design of an application that has features like drag and drop etc.
- Design of application that decreases data transfers between the client and the server.

4.2 Architectural Design

4.2.1 Architectural Context Diagram

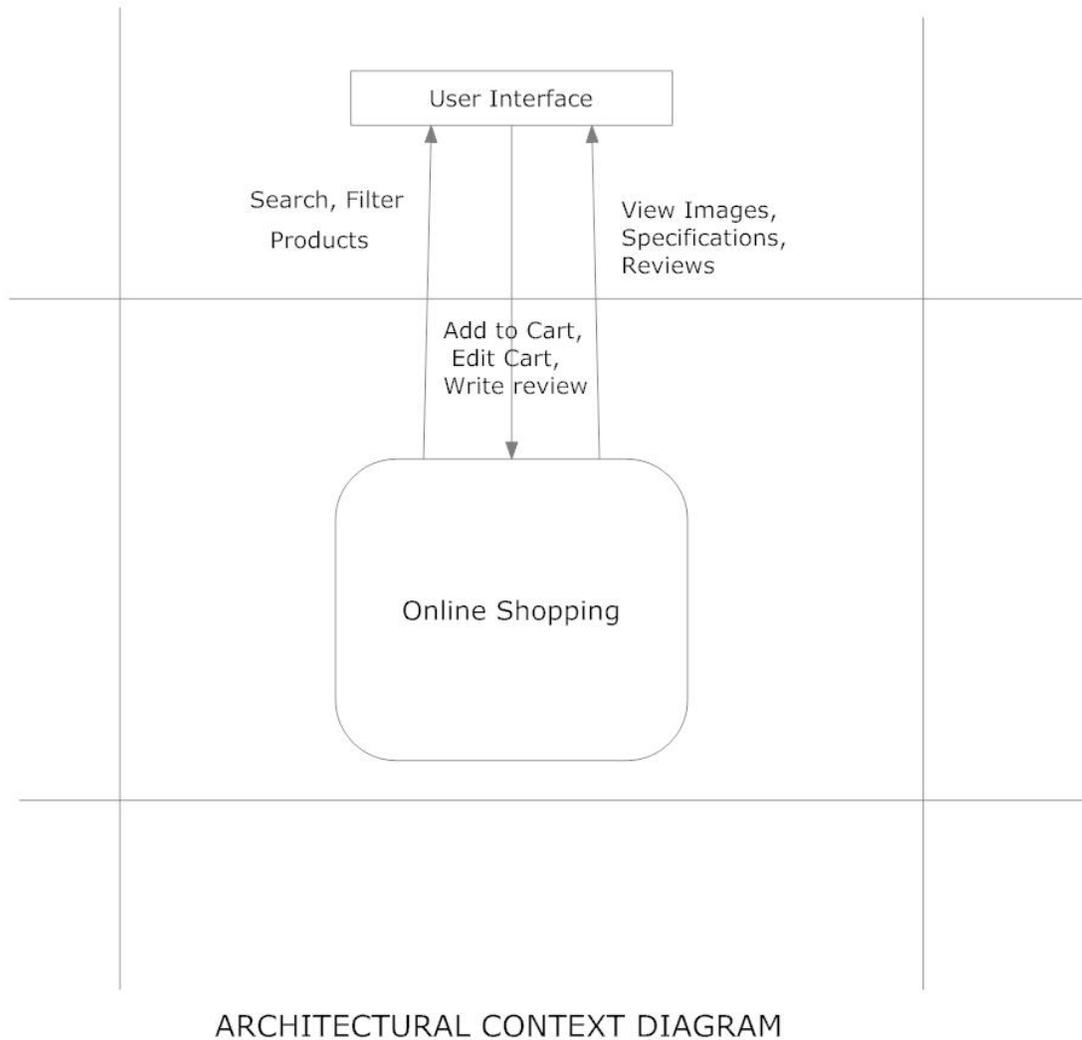


Figure 4.1 Architectural Context Diagram

4.2.2 Description of Architectural Design

In this context diagram, the information provided to and received from the 'Online Shopping' is identified. The arrows represent the information received or generated by the application. The closed boxes represent the set of sources and sinks of information.

In the system, we can observe that the user interacts with the application through a graphical user interface. The inputs to the system are the Search and Filter criteria provided by the user and a new review written by the user. Also, the output is in the form of Repeater and grid views which present the users with list of Products available. The users can view complete specification, view Images and reviews by other users.

4.3 Procedural/Modular Approach

Following are all the modules designed for the Online Shopping System.

4.3.1 Shop Products Module

This module starts when the user visits the home page or when a user searches for a product by entering a search term. This part of the application includes displaying all the products that are available or the products that match the search term entered by the user. The user can then filter these products based on various parameters like manufacturer, product type, operating system supported or a price range. The user browse through the products and each product would be displayed with an image and its features like operating system supported, number of user licenses and if it is a full version or an upgrade version. A user can add a product to the cart either by dragging the product and dropping it in the cart or by clicking a button. The user would be able to see the shopping cart summary.

4.3.2 Product Description Module

This module starts when a user visits the product description page. A user can view various images of the product of different sizes. The use can see an enlarged image in a popup window. The user can view the complete specification of the product like its features, operating system supported, system requirements etc. A user can also view the manufacturer information and also information about rebates, exchange policies etc. A user can also view the reviews of the product. A user can also write a review for the product.

4.3.3 Shopping Cart Module

This module starts when the user views the shopping cart. All the products that have been added to the shopping cart by the user are listed along with their price and the quantity. The total price of all the products added to cart is displayed. A user can edit the quantity of each product or remove the product from the shopping cart. A user can remove the product from the cart by clicking a button or by dragging the product and dropping it outside the cart. The total price changes accordingly when a user edits the quantity of a product or when a product is removed from the cart.

5. Implementation

5.1 Database Design and Implementation

The design of the database was similar to the analysis phase. The database has been developed using SQL Server 2005.

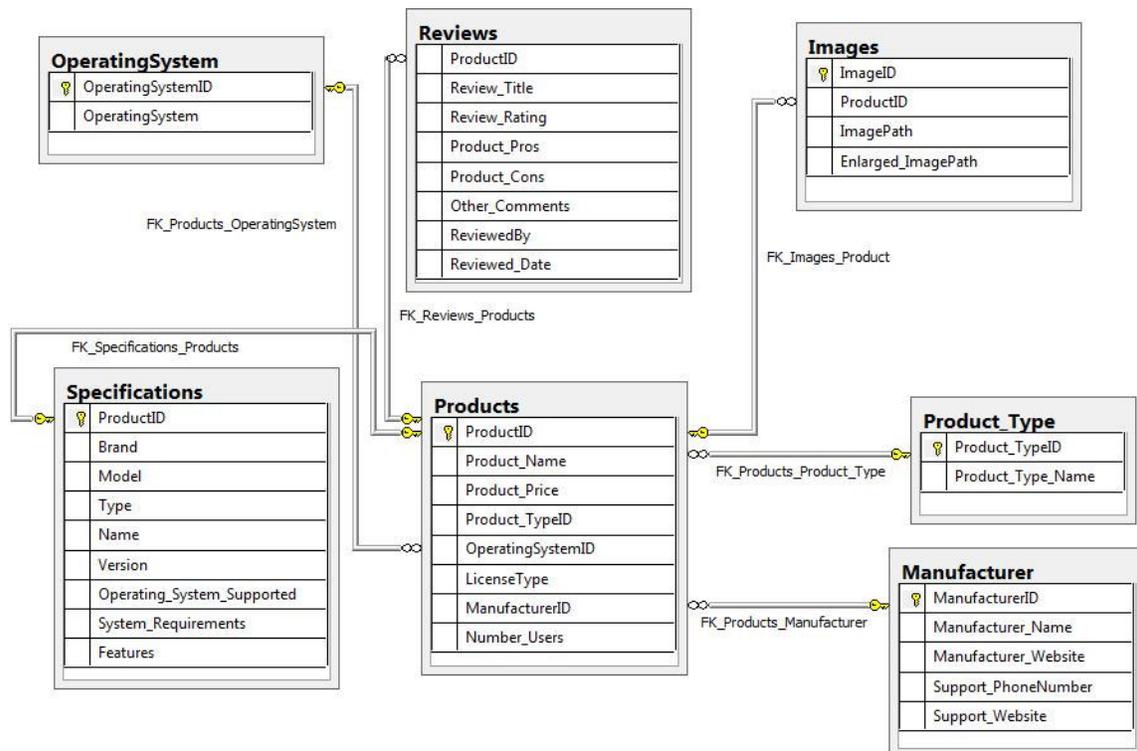


Figure 5.1 Database Implementation using SQL Server 2005

These are the main tables in the application and others are lookup and query tables. The tables were derived from the ER-Diagram.

5.2 User Interface Design and Implementation

The user interface of the application has been designed using Microsoft Visual Studio 2008. The main controls used in the design are Repeaters, Ajax Update panels, Ajax toolkit controls like modal popup, textbox watermark and Auto complete controls.

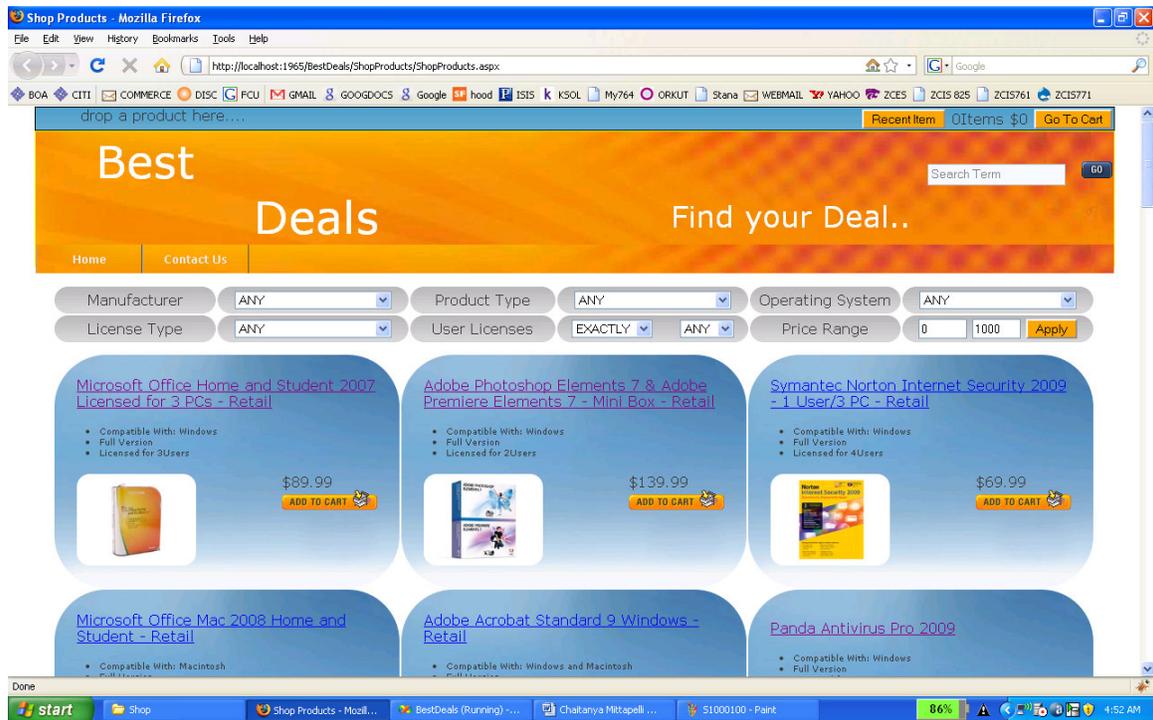


Figure 5.2 Shop Products Page

The user can see the list of products that are available. The user can search for products by entering the search term into the search textbox provided on the top. This text box is watermarked with the words “Search Terms” to let the user know that this is the place to enter the search terms. The user can filter the products by using the dropdown lists.

A user can drag and drop a product in the shopping cart. This is shown in the figure 5.3. The cart is updated accordingly as shown in figure 5.4

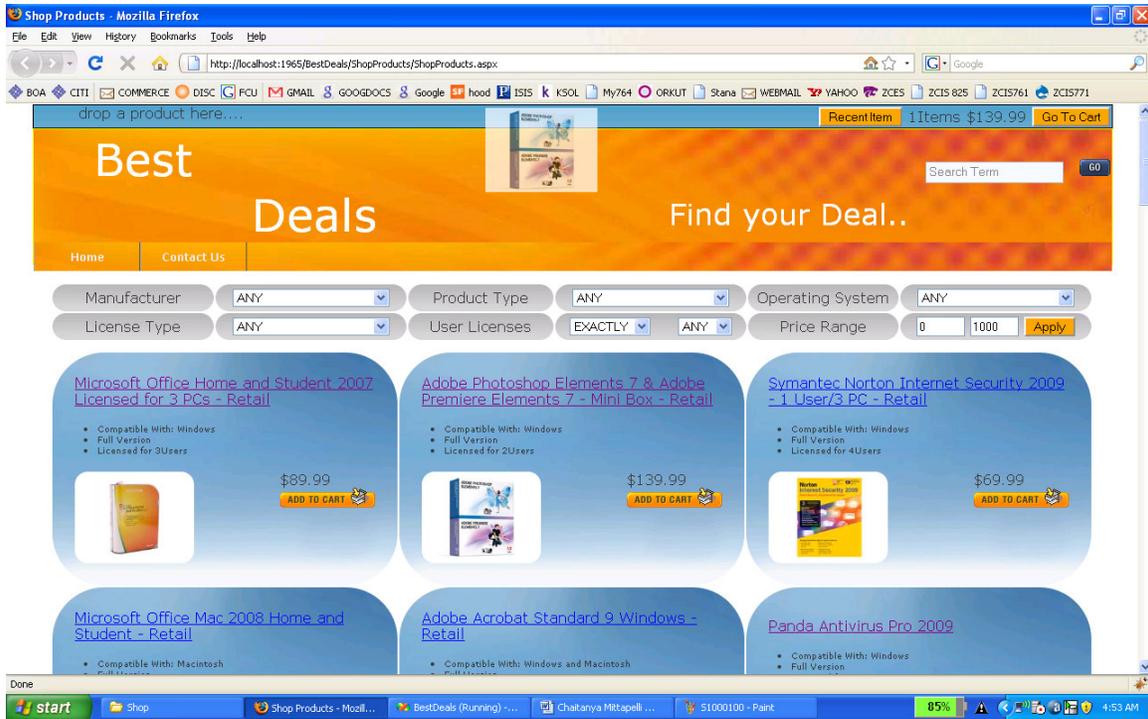


Figure 5.3 Figure showing Drag and Drop a product in the cart

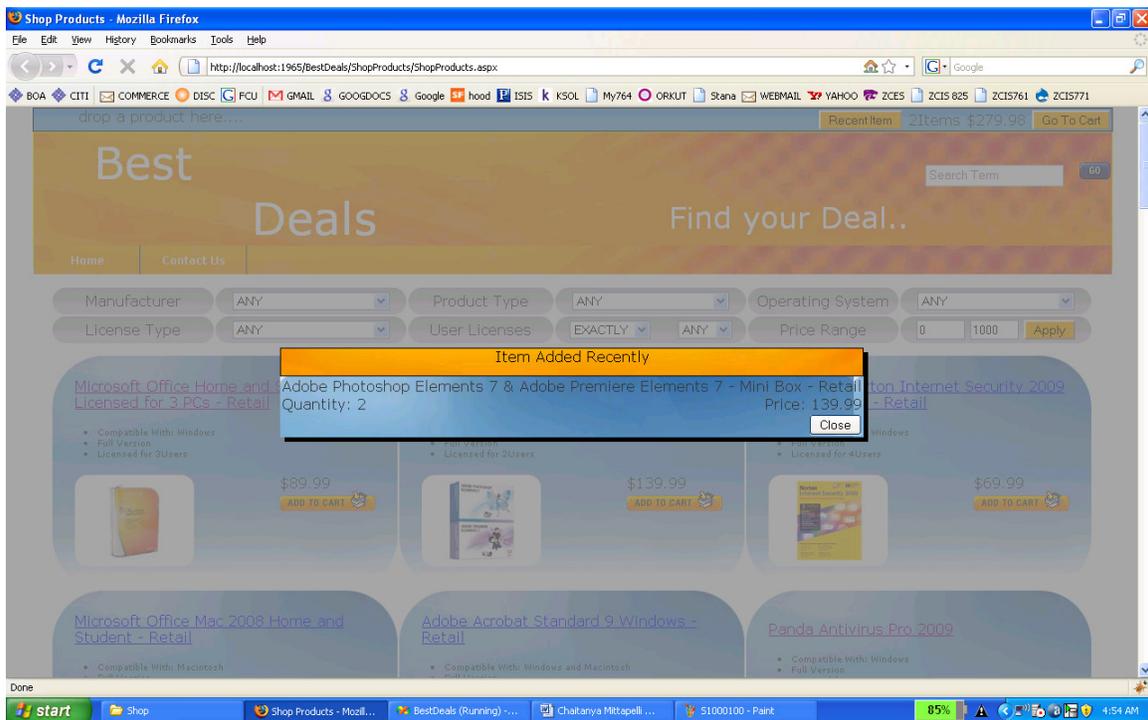


Figure 5.4 figure showing that Cart has been Updated

A user can view the complete description of the product by clicking on the product link. This is shown in figure 5.5. The user can move the cursor to the small images to view the same image in the enlarged position. The user can click on the enlarged picture to see a still bigger image in a pop up window as shown Figure 5.6

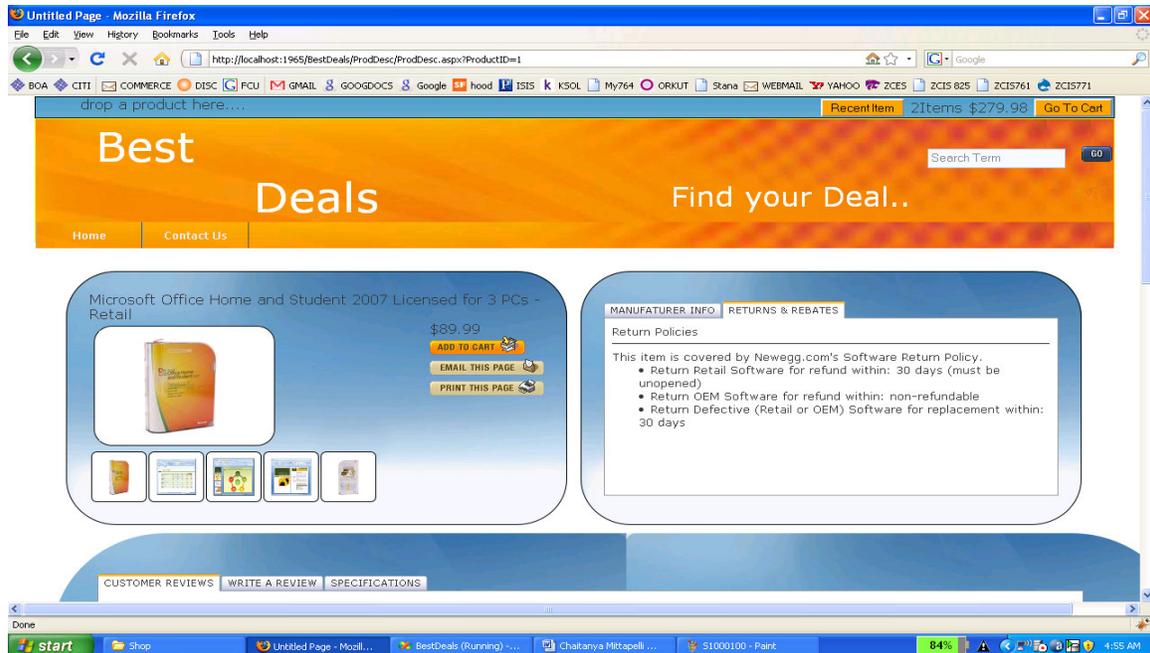


Figure 5.5 Product Description Page

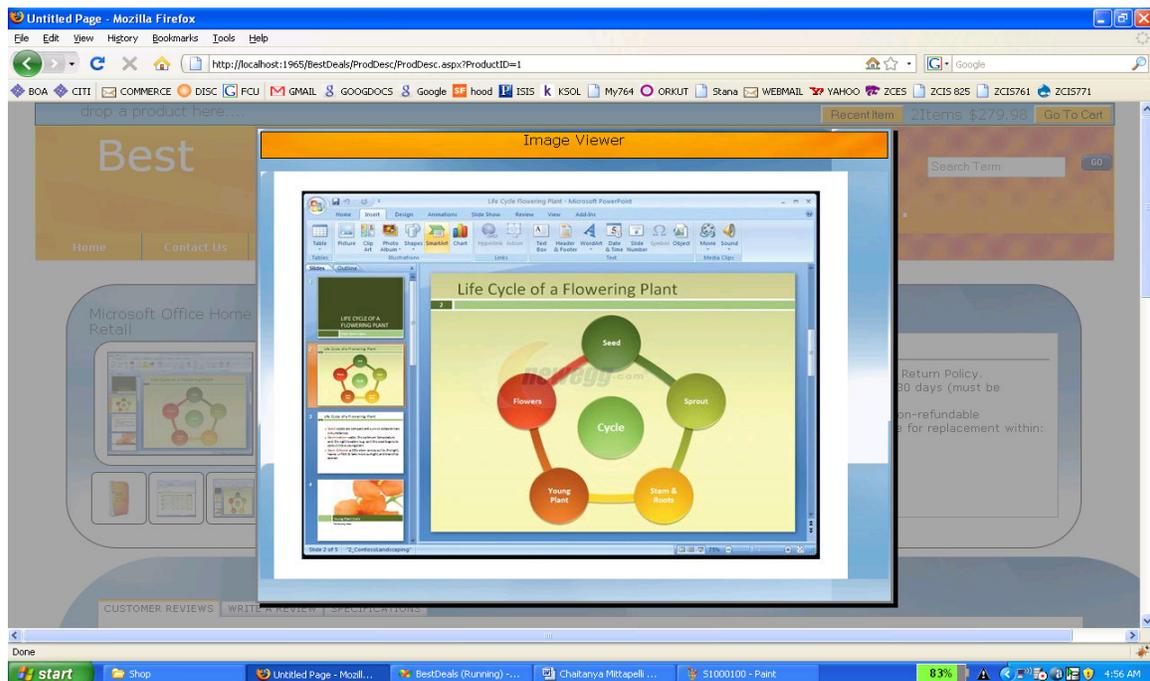


Figure 5.6 Pop up window showing an enlarged Image

A user can also write a review by clicking on the write review tab panel as shown in figure 5.7. Similarly a user can click on the tab panel customer reviews, specifications, manufacturer Info etc to see the respective information.

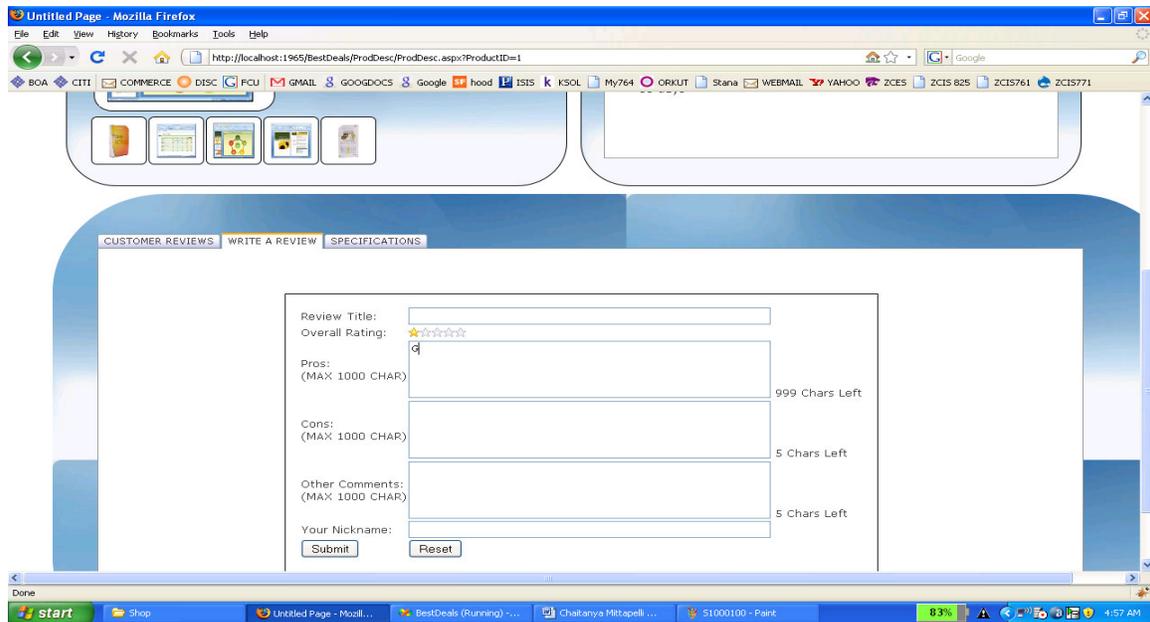


Figure 5.7 Write Review Panel

The user can click on the Go to cart button to see the Items in the cart as shown in figure 5.8. The Summary is at the bottom of the page.

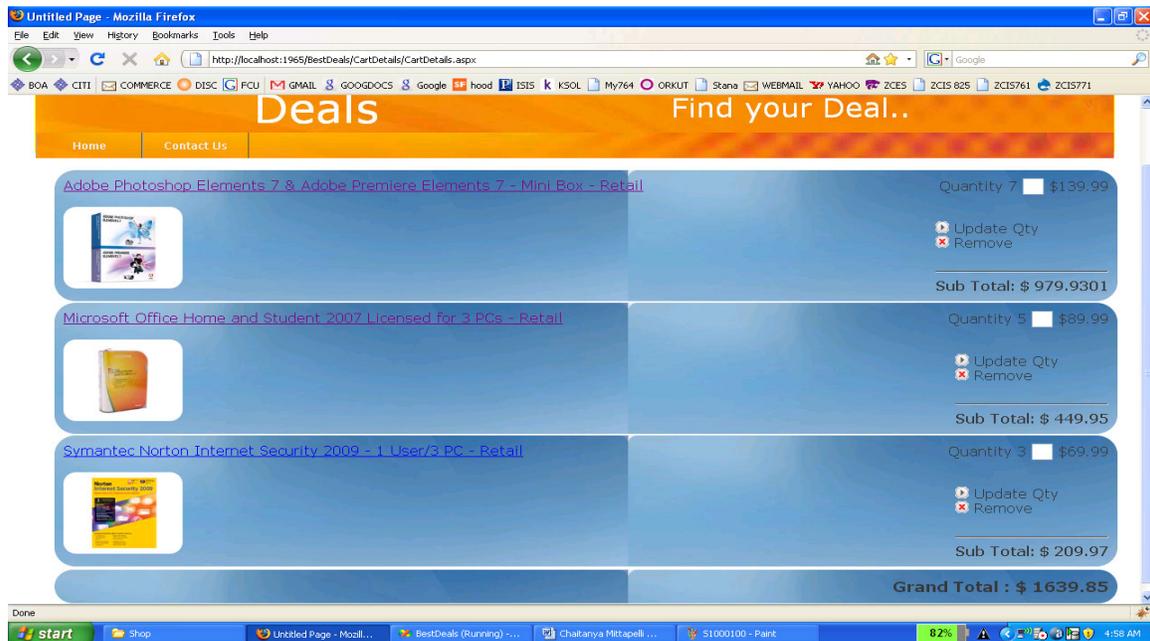


Figure 5.8 Cart Details Page

5.3 Technical Discussions

The products can be filtered based on various parameters like Manufacturer, Product Type, Operating System supported etc. Initially it was decided to have the various list items predefined. But with time new manufacturers and product types could be added. So the values for the list of manufacturers and product types are loaded dynamically by retrieving from the database. Also it was decided initially to have a drop down list for price range and the user could select a price range from the ranges available. But this would limit the user's ability to filter the products based on different price ranges. Instead providing two text fields so that the user can enter their price range would give them more flexibility.

A product could be added to a shopping cart by dragging it and dropping it in the cart area. Items in the cart could be removed by clicking a button. To maintain symmetry and ease of use products could be removed from the cart by dragging the product out of the cart.

A product can be added to the cart by dragging it and dropping it in the cart. Initially it was decided that when a product is dropped in the cart the cart summary label could be updated on the client side without any call to the server and later the session variables (Shopping cart) could be updated. This would result in loss of information when the user loses internet connection. So when a product is dropped in the cart area a web service is called and this service updates the session variables for the shopping cart and the cart summary is recalculated and sent back to the client. This would improve the reliability of the application.

6. Testing

Software testing is a process of running with intent of finding errors in software. Software testing assures the quality of software and represents final review of other phases of software like specification, design, code generation etc.

6.1 Unit Testing

Unit testing emphasizes the verification effort on the smallest unit of software design i.e.; a software component or module. Unit testing is a dynamic method for verification, where program is actually compiled and executed. Unit testing is performed in parallel with the coding phase. Unit testing tests units or modules not the whole software.

I have tested each view/module of the application individually. As the modules were built up testing was carried out simultaneously, tracking out each and every kind of input and checking the corresponding output until module is working correctly.

The functionality of the modules was also tested as separate units. Each of the three modules was tested as separate units. In each module all the functionalities were tested in isolation.

In the Shop Products Module when a product has been added to cart it has been made sure that if the item already exists in the shopping cart then the quantity is increased by one else a new item is created in the shopping cart. Also the state of the system after a product has been dragged in to the shopping cart is same as the state of the system if it was added by clicking the add to cart button. Also it has been ensured that all the images of the products displayed in the shop products page are drag gable and have the product property so that they can be dropped in the cart area.

In the Product Description Module it has been tested that all the images are displayed properly. Users can add review and the as soon as a user adds a review it is updated in the view customer review tab. It has been checked to see if the whole page refreshes or a partial page update happens when a user writes a review.

In the Cart Details it has been tested that when a user edits a quantity or removes a product from the cart, the total price is updated accordingly. It has been checked to see if the whole page refreshes or a partial page update happens when a user edits the cart.

Visual Studio 2008 has in built support for testing the application. The unit testing can be done using visual studio 2008 without the need of any external application. Various methods have been created for the purpose of unit testing. Test cases are automatically generated for these methods. The tests run under the ASP.NET context which means settings from Web.config file are automatically picked up once the test case starts running.

Methods were written to retrieve all the manufacturers from the database, strings that match a certain search term, products that match certain filter criteria, all images that belong to a particular product etc. Unit test cases were automatically generated for these methods and it can be seen in figure 6.1 that the tests have passed.

6.2 Integration Testing

In integration testing a system consisting of different modules is tested for problems arising from component interaction. Integration testing should be developed from the system specification. Firstly, a minimum configuration must be integrated and tested.

In my project I have done integration testing in a bottom up fashion i.e. in this project I have started construction and testing with atomic modules. After unit testing the modules are integrated one by one and then tested the system for problems arising from component interaction.

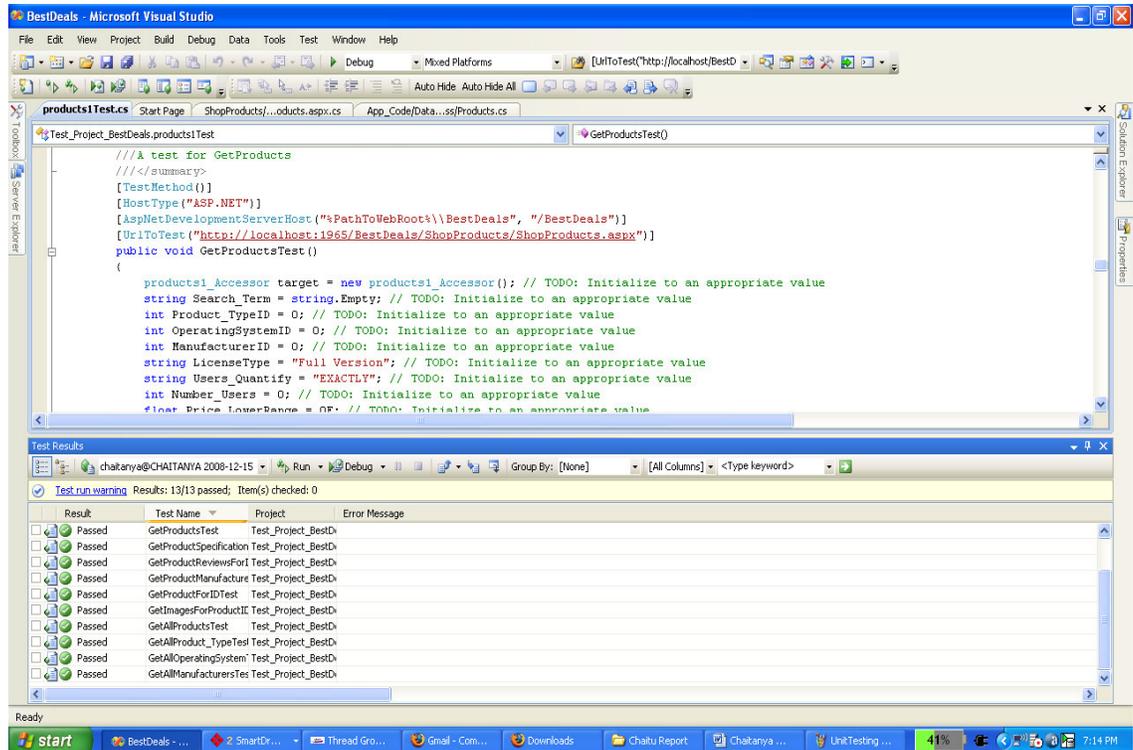


Figure 6.1 Unit Testing in Visual Studio 2008

6.3 Validation Testing

It provides final assurances that software meets all functional, behavioral & performance requirement. Black box testing techniques are used.

There are three main components

- Validation test criteria (no. in place of no. & char in place of char)
- Configuration review (to ensure the completeness of s/w configuration.)
- Alpha & Beta testing-Alpha testing is done at developer's site i.e. at home & Beta testing once it is deployed. Since I have not deployed my application, I could not do the Beta testing.

Test Cases- I have used a number of test cases for testing the product. There were different cases for which different inputs were used to check whether desired output is produced or not.

1. Addition of a new product to the cart should create a new row in the shopping cart.
2. Addition of an existing product to the cart has to update the quantity of the product.

3. Any changes to items in the cart have to update the summary correctly.
4. Because same page is inserting data into more than one table in the database atomicity of the transaction is tested.
5. The state of the system after a product has been dragged in to the cart should be same as the state of the system if the same product is added to the cart by clicking a button.

6.4 White Box Testing

In white box testing knowing the internal working of the product, tests can be conducted to ensure that internal operations are performed according to specification and all internal components have been adequately exercised. In white box testing logical path through the software are tested by providing test cases that exercise specific sets of conditions and loops.

Using white-box testing software developer can derive test case that

- Guarantee that all independent paths within a module have been exercised at least once.
- Exercise all logical decisions on their true and false side.
- Exercise all loops at their boundaries and within their operational bound.
- Exercise internal data structure to ensure their validity.

At every stage of project development I have tested the logics of the program by supplying the invalid inputs and generating the respective error messages. All the loops and conditional statements are tested to the boundary conditions and validated properly.

6.5 Performance Testing

Jakarta JMeter, a tool for testing applications was used to simulate the virtual users (clients) and test the performance of the system. It can be used to test performance both on static and dynamic resources (files, Servlets, Perl scripts, Java Objects, Data Bases and Queries, FTP Servers and more). It can be used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types. It can be used to make a graphical analysis of performance and test the server/script/object behavior under heavy concurrent load.

I have done performance testing to achieve an estimate of the peak and sustained load the application. This has done with few pages like the Shop Products (extensive Database access, business logic Intensive and more Images) and the Cart Details (simple page). A few sample Screenshots of test results are shown below. The tests have been conducted by running the application (server) and JMeter on same machine. These test results do not include factors like network bandwidth etc as the server is running on the same machine along with JMeter.

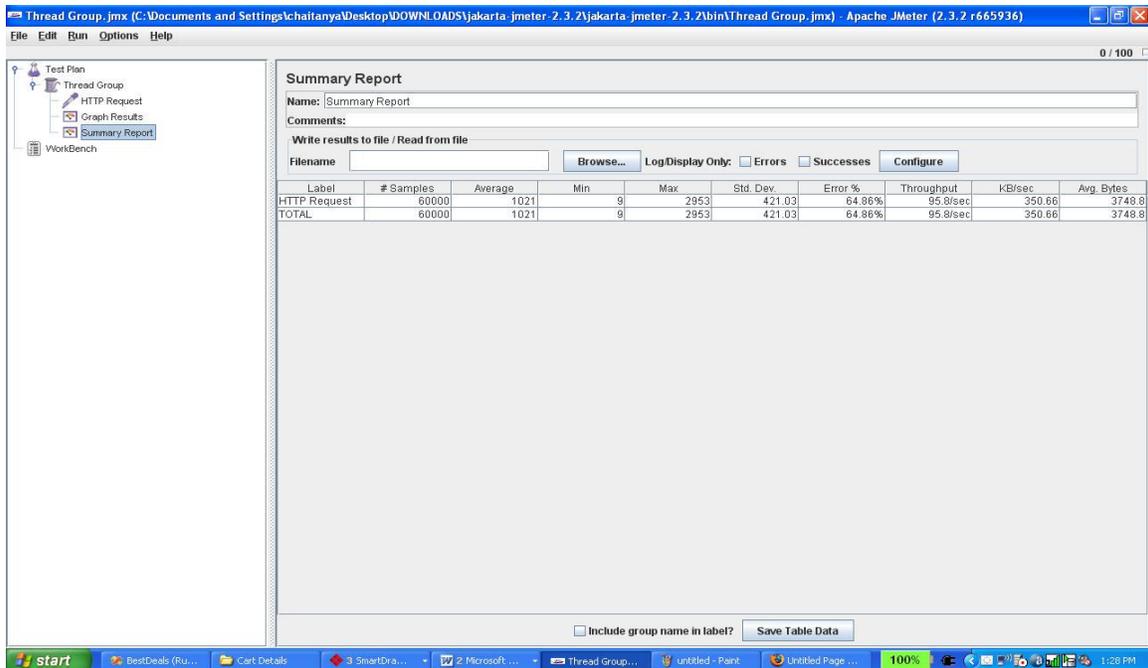


Figure 6.3 Summary Reports for 100 Users and 600 Loop Count for Cart Details Page

Comparison of Constant Users vs. Constant Loop-Count

In the following test, the number of users has been kept constant and the Loop Count has been increased.

Users	Loop Count	Ramp Up period(sec)	Cart Details Page Average Response(ms)
100	150	10	1060
100	300	10	1071
100	600	10	1021
100	1000	10	1033

The above data shows high response average which increases with the loop count.

Observations

Response Time increases rapidly with number of users but not very much when the users are kept constant and only loop-count is increased. This is because, if the number of users is kept constant and only the loop count is increased, the number of requests handled by the server per second remains constant for every execution of the loop count and for every increase in the loop count. Hence the response time will not increase drastically in this case. Whereas, if the users are increased and loop count is kept constant, the requests handled by the server per second increases with increasing users and hence the longer response time.

Comparison of Response Times of the 2 WebPages

Local Testing:

Users	Loop Count	Ramp Up period(sec)	Cart Details Page(ms)	Shop Products page(ms)
100	150	10	1060	28177
500	150	10	8075	105388
1000	150	10	11993	135273

Observations:

Response Time of a complex webpage with database and business logic functions is far more than a simple webpage.

Factors affecting Response Time:

- Limited System Hardware Resources (CPU, RAM, Disks) and Configuration
- JMeter Tests and Application running on the same machine.

Remote Testing:

Users	Loop Count	Ramp Up period(sec)	Cart Details Page(ms)	Shop Products page(ms)
100	150	10	792	8312
500	150	10	6392	99069
1000	150	10	20457	227056

Observations:

Response Time of a complex webpage with database and business logic functions is far more than a simple webpage. The Response times of remote testing are better than those of local testing when the number of users is comparatively lesser.

Factors affecting Response Time:

- Better Hardware Resources (CPU, RAM, Disks) and Configuration for the Application as it was hosted on a web server.
- JMeter had better access to hardware resources as the application is not on the same machine.

Using this above tabular data, I can say that the system is adequate to handle the normal load and the users won't lose their focus.

7. Results & Challenges

The application can be used for any Ecommerce application. It is easy to use, since it uses the GUI provided in the user dialog. User friendly screens are provided. The application is easy to use and interactive making online shopping a recreational activity for users. It has been thoroughly tested and implemented.

7.1 Challenges

- Compatibility with browsers like Mozilla Firefox, Internet explorer etc
- Using a layered approach in developing the application which would make the application maintainable.
- Learning new technologies like using JavaScript for drag and drop behavior and Ajax toolkit controls with little guidance.

The overall idea of doing this project is to get a real time experience. Learn new technologies.

8. Conclusions

The 'Online Shopping' is designed to provide a web based application that would make searching, viewing and selection of a product easier. The search engine provides an easy and convenient way to search for products where a user can Search for a product interactively and the search engine would refine the products available based on the user's input. The user can then view the complete specification of each product. They can also view the product reviews and also write their own reviews. Use of Ajax components would make the application interactive and prevents annoying post backs. Its drag and drop feature would make it easy to use.

8.1 Limitations

This application does not have a built in check out process. An external checkout package has to be integrated in to this application. Also users cannot save the shopping carts so that they can access later i.e. they cannot create wish lists which they can access later. This application does not have features by which user can set price ranges for products and receive alerts once the price reaches the particular range.

8.2 Scope for Future Work

The following things can be done in future.

- The current system can be extended to allow the users to create accounts and save products in to wish list.
- The users could subscribe for price alerts which would enable them to receive messages when price for products fall below a particular level.
- The current system is confined only to the shopping cart process. It can be extended to have an easy to use check out process.
- Users can have multiple shipping and billing information saved. During checkout they can use the drag and drop feature to select shipping and billing information.

9. References

- All about Microsoft controls in C#
<http://www.msdn.microsoft.com/>
- Wikipedia for various diagrams & testing methods
<http://www.wikipedia.org/>
- Cool text for Images and Buttons
<http://cooltext.com/>
- K-State Research Exchange for samples in report writing
<http://krex.k-state.edu/dspace/handle/2097/959>
- Smart Draw for drawing all the Diagrams used in this report.
<http://www.smartdraw.com/>
- Sample Ecommerce Application
<http://www.NewEgg.com>
- Ajax Toolkit controls
<http://asp.net/ajax>