PERFORMANCE EVALUATION OF J2EE & .NET WEB SERVICES INTERACTING WITH A .NET CLIENT

by

RAJA SANJEEV KUMAR NAKKA

B.Tech., Acharya Nagarjuna University, 2005

A REPORT

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computing & Information Sciences
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2007

Approved by:

Major Professor
Dr. Daniel Andresen

Abstract

The objective of this project is to build an e-commerce website using ASP.NET as well as J2EE technologies. Visual Studio 2005 provides a great IDE to build seamless front end layer for the websites with minimum effort. Hence, it is used to build the presentation layer. The business logic layer is developed in ASP.NET 2.0 and J2EE, exposing the functions as web services. Two similar clients have been developed in ASP.NET. These two clients invoke the .NET and J2EE web services.

The two similar websites are subjected to testing for correctness and performance. Stress and load testing is performed thoroughly on the website. A detailed analysis of the results is done using the response times and throughputs from various tests.

This report proves that we can build a website using a presentation tier developed in ASP.NET consuming any of the two technologies' (ASP.NET & J2EE) web services. This proves to be very useful in scenarios where we need the best overall user experience regardless of the technology, to replace or extend an existing business tier by using a different technology.

Table of Contents

List of Figures

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.

CHAPTER 1 - Introduction

In this project, two similar e-commerce websites are built. One website is a homogenous website i.e. .NET client and .NET web services are combined to form a working website. The other one is heterogeneous; the same .NET client invokes the web services written in J2EE. This project aims at proving the interoperability of a .NET client, accessing business logic from either of the two technologies (ASP.NET & J2EE). Interoperability is connecting and building applications that work with each other to such an extent that the presentation to the user is seamless. Interoperability is considered to be very useful as it facilitates the programmers to utilize the best features of various technologies in building an application.

The present e-commerce website is developed using an e-commerce template which is available from MSDN. The master page, themes and style sheets of this template gives the user a very rich user interface.

1.1 Salient Features of the website:

The website is provided with all the basic features required in an online store.

- Users can register themselves and can login to their account.
- Users can browse products of various categories. Reviews and ratings from other buyers will help the buyer a lot in making the purchase decision.
- The online store is provided with a flying shopping basket. When a user adds any product to the cart, the product “flies” to the cart location and the cart gets updated
- The website has Ajax-enabled dynamic enlargement of the picture of the product. Initially, all the products will be of thumbnail size to save the screen space. By placing the cursor on the image, it enlarges to a much larger size.

- Users can register themselves and if they forgot their password or userid, their new password or userid will be sent to their email
- The website has Google maps capability. The pickup stores for a product can be located on the Google maps. Users can look at price of the product at that particular location.
- The website is provided with the search results from the Google. Users can learn about the product more due to the high Google search relevancy.

1.2 Objective:

The objective of the report is to prove that an ASP.NET website's business logic can be rewritten as .NET web services as well as J2EE web services. The two different web applications work functionally the same as that of the previous website. So, transferring the business logic from .NET web services to J2EE web services or vice versa is indeed feasible.

Rigorous performance testing is performed on both the websites: ASP.NET client invoking ASP.NET web services and ASP.NET client invoking the J2EE web services. The performance results are analyzed resulting in the evaluation of the .NET and J2EE web services for various factors.

1.3 Document Overview:

The first part of the report gives the objective of the report and the features of the website. Chapter 2 provides with the necessary background knowledge about the technologies and tools used in this project. In the third chapter, design and implementation details will be explained with various diagrams. Screen shots of significant web pages are provided. Then the testing analysis and results will be shown, followed by conclusions derived from them.

CHAPTER 2 - Related Work

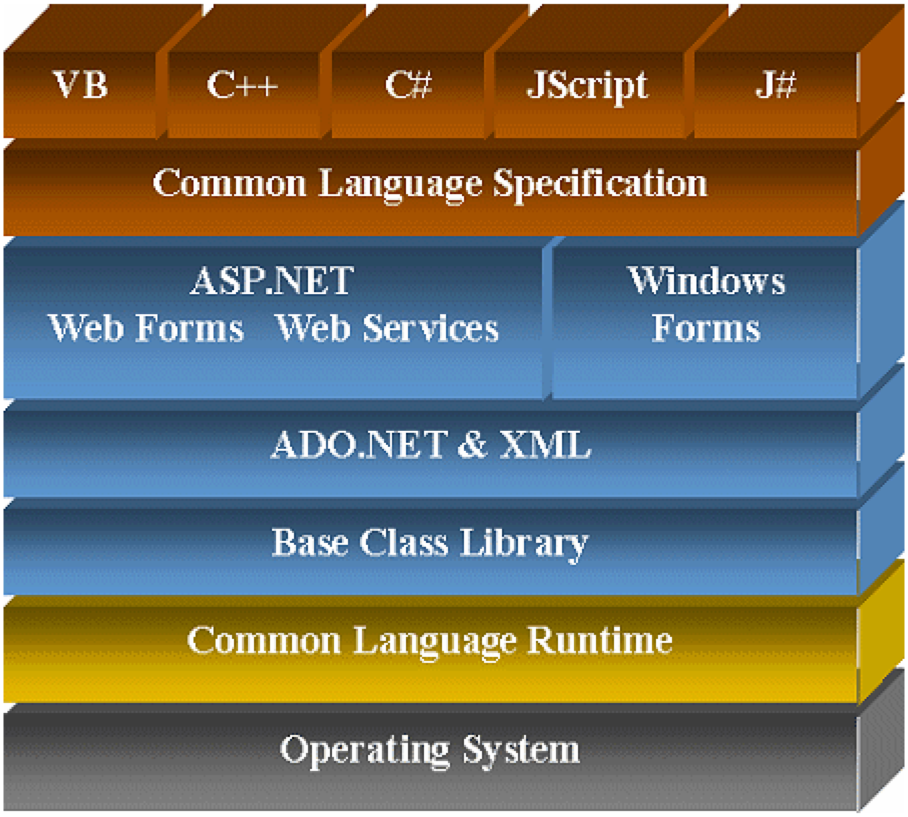
2.1 Tools and Technologies

2.1.1 ASP.NET:

The .NET framework is the infrastructure for the new Microsoft .NET platform. It is a common environment for building, deploying, and running web services and web applications. ASP.NET and Web Services are the main components of what is called the .NET framework. One of the most significant features of the .NET framework is it enables code written in multiple languages to work together seamlessly. One of the significant improvements in ASP.NET is the way the code is handled at run time. Instead of interpreting the page source each time a client requests a page; ASP.NET seamlessly compiles the page to Microsoft intermediate language (MSIL) code the first time the page is requested. Once the page is compiled in MSIL, the just-in-time (JIT) compiler converts the MSIL to native code.

Layered on top of the system services is the CLR which loads and runs code written in any language which targets the runtime. The .NET Framework classes provide classes that can be called from any .NET enabled language. On the top of the .NET Framework class library is ADO.NET and XML data. ADO.NET is a set of classes that provide data access support for the .NET Framework based on ADO but to work with XML and work in a disconnected environment. On top of ADO.NET and XML lies specific support for two different types of applications. One is the traditional client application that uses window Forms and the other is ASP.NET which offers Web Forms and XML Web Services. On top of these is the common language specification which ensures that each language has a common set of features.

Figure 2.1 ASP.NET Framework Architecture[12]:



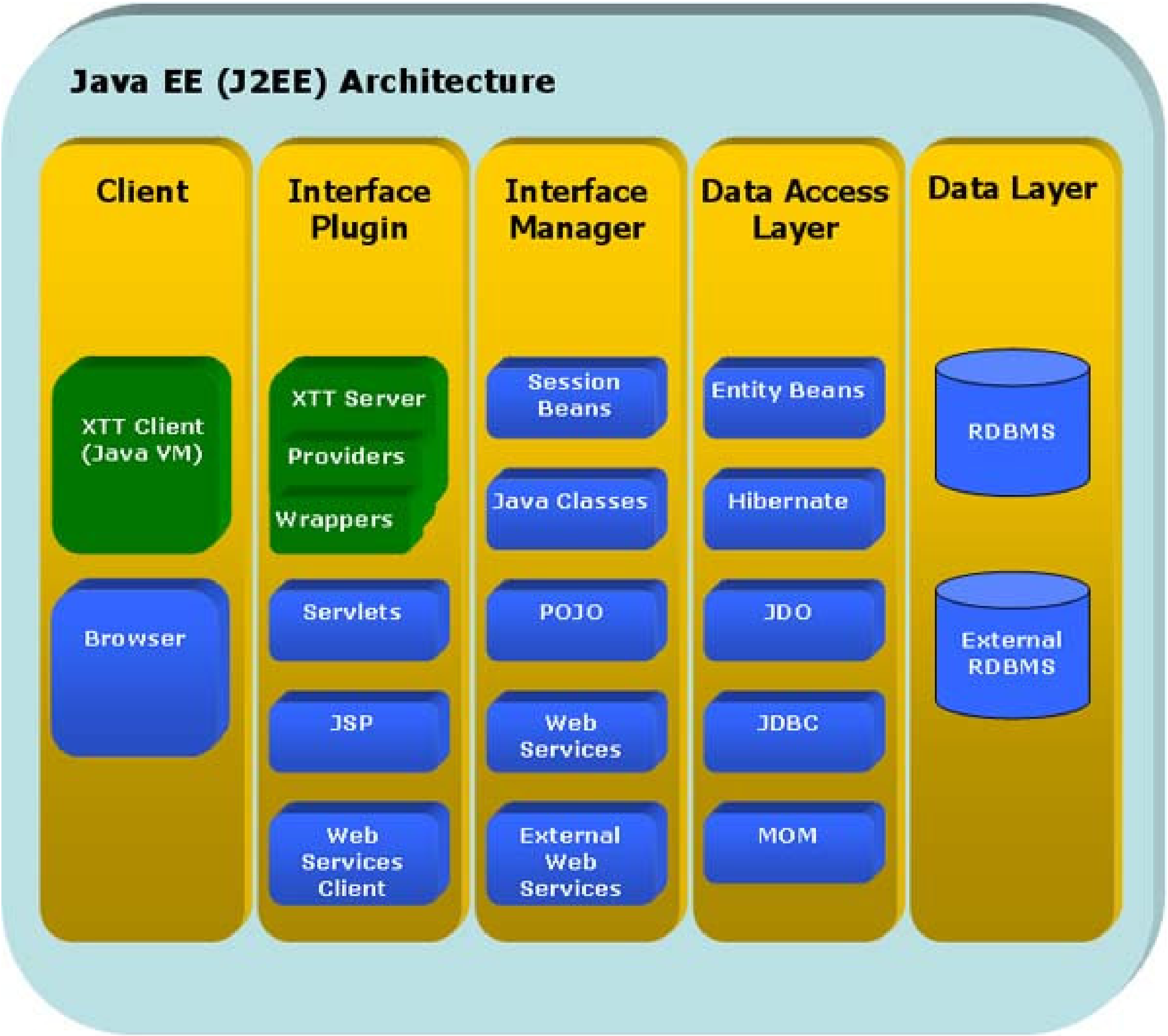
2.1.2 Visual Studio 2005:

Visual Studio.NET is the only advanced IDE that can be used as a rapid application development tool for ASP.NET. It helps the developers very much offering the differentiation of code and presentation elements in a web page. Visual Studio 2005 also includes a local web server, separate from IIS that can be used to host ASP.NET applications during development and testing. It can be used to create robust applications using the Microsoft .NET Framework 2.0, the .NET Compact Framework 2.0, and native code—all supported by Microsoft Visual Studio 2005[9].

2.2.1 J2EE:

The J2EE platform is a collection of related technology specifications that describe required APIs and policies. Java 2 platform Enterprise Edition is used to develop portable, scalable, secure server side Java applications. It provides a very robust framework to develop distributed multi-tiered applications. It can be used to build traditional web sites, software components, or packaged applications. J2EE has been extended to include support for building XML-based web services as well. These web services can interoperate with other web services that may or may not have been written to the J2EE standard. It was basically developed to introduce the concept of “write once, run anywhere” concept. The most important feature of J2EE is its substantial support for web services that enable web service endpoints based on servlets and Enterprise JavaBeans. J2EE has several API’s JDBC, RMI, XML, etc. and also includes Enterprise JavaBeans (EJB), servlets, JSP which allows the creation of enterprise applications [11]. The primary objective of the Java EE 5 platform is to provide developers a powerful set of APIs while reducing development time, reducing application complexity, and improving application performance.

Figure 2.2 J2EE Architecture [10]:



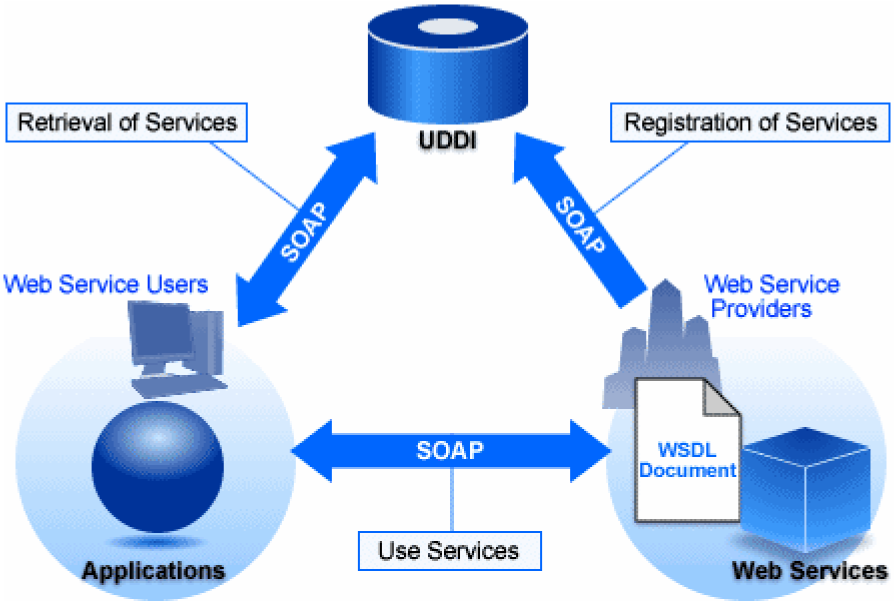
NetBeans 5.5:

The NetBeans IDE is a free, open-source Integrated Development Environment for software developers. The IDE runs on many platforms including Windows, Linux, Solaris, and the MacOS. It is a fast and feature rich tool for developing Java software. It is standards-compliant and runs on any operating system where a Java Virtual Machine is available. NetBeans 5.5 Beta supports the Java Enterprise Edition 5 platform, and most notably the Java Persistence, EJB 3 and JAX-WS 2.0 specifications. NetBeans IDE 5.5 adds support for Java EE 5 and Sun Java System Application Server PE 9. NetBeans 5.5 generates JAX-WS 2.0 artifacts for Java SE 6 and Java EE 5 projects. Configuration is done through binding's customization (like asynchronous calls, package info). NetBeans has simplified and updated template wizards. NetBeans5.5. has support for Enterprise Java Beans 3 session beans and facilitates for simplified development of session beans and message driven beans. It has enhanced Java source editor with code completion, error checking, and hints specifically targeting EJB 3. Over all, I find it as a very good and easy tool to develop web services.

2.2 Web Services

Web services are self-contained, self-describing, modular applications that can be published, located, and invoked across the Web. Web services perform functions, which can be anything from simple requests to complicated business processes...Once a Web service is deployed, other applications can discover and invoke the deployed service. The basic web services platform is XML plus HTTP. HTTP is a ubiquitous protocol, running practically everywhere on the Internet. The full-function web services platform can be thought of as XML plus HTTP plus SOAP plus WSDL plus UDDI. At higher levels, one might also add technologies such as XAML, XLANG, XKMS, and XFS -- services that are not universally accepted as mandatory. Web services provide the language and platform independence by separating the specification from implementation and provide the loosely coupled integration between applications by message-based, synchronous or asynchronous communication. Protocols do not make any assumption on the underlying implementation. The promise of web services is to enable a distributed environment in which any number of applications, or application components, can interoperate seamlessly among and between organizations in a platform-neutral, language-neutral fashion.

Figure 2.3 Web Services[13]:



Web services are composed of mainly four components, Extended Markup Language (XML), Simple Object Access Protocol (SOAP), Web Service Description Language (WSDL) and Universal Description, Discovery and Integration (UDDI).

2.2.1 SOAP:

SOAP is a protocol specification that defines a uniform way of passing XML-encoded data. In also defines a way to perform remote procedure calls (RPCs) using HTTP as the underlying communication protocol. Simple Object Access Protocol, a lightweight XML-based messaging protocol used to encode the information in Web service request and response messages before sending them over a network. SOAP is a platform-independent protocol and can be used to send messages of various internet protocols.

2.2.2 WSDL:

WSDL provides a way for service providers to describe the basic format of web service requests over different protocols or encodings. WSDL is used to describe *what* a web service can do, *where* it resides, and *how* to invoke it. WSDL is often used in combination with SOAP and XML Schema to provide web services over the internet. A client program connecting to a web service can read the WSDL to determine what functions are available on the server. Any special data types used are embedded in the WSDL file in the form of XML Schema.

2.2.3 UDDI

UDDI provides a mechanism for clients to dynamically find other web services. Using a UDDI interface, businesses can dynamically connect to services provided by external business partners. UDDI is a platform-independent, XML based registry for businesses worldwide to list themselves on the internet.

2.2.4 XML

XML is a simple and flexible format designed to meet the challenges of large scale electronic publishing. Like HTML, XML is a markup language and has its roots in SGML; thus, it is a specification for “tagging” documents in a meaningful way. Unlike HTML, which provides means for visualizing data, XML allows data to be self describing and structured and so is meant primarily for the interchange of the data, not its visualization. This helps in interchanging the data over different platforms.

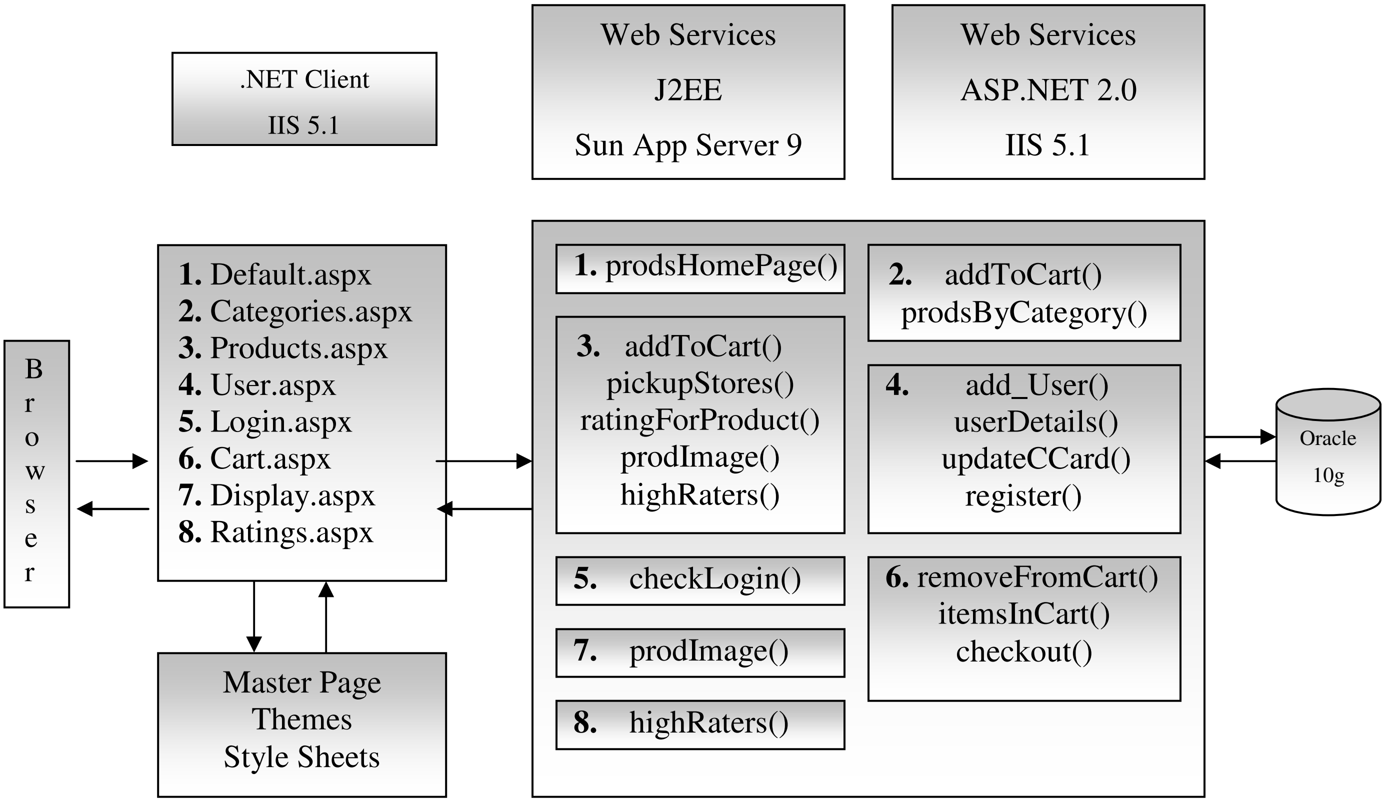
2.3 AJAX

AJAX stands for Asynchronous JavaScript and XML. Ajax enables the web applications to work behind the scenes, getting data as they need it, and displaying that data as we want. This is done by exchanging small amounts of data with the server, so that the entire web page does not have to be reloaded each time the user makes a change. This is meant to increase the web page's interactivity, speed, and usability. In this project, AJAX is used to verify the login information, to get the image for dynamic enlargement of the product’s image, to get the search results from google servers, to get the cart information, etc.

CHAPTER 3 - Implementation

3.1 System Architecture:

Figure 3.1 Website Architecture Diagram:



3.1.1 Client: The ASP.NET client is interactive and rich looking; themes and styles are used from the online e-commerce template available from MSDN. Master pages allow the structure of all web pages to be consistent. This website's master page consists of the menu, popular products, title images and logo. The client mainly consists of eight web pages. Client will just display the data retrieved by invoking the web services.

3.1.2 Browser: The browser makes the normal http request most of the time. It also sends out AJAX request where ever it is needed. The website is tested for Internet Explorer and Mozilla browser compatibility. It works fine and looks similar in the two browsers.

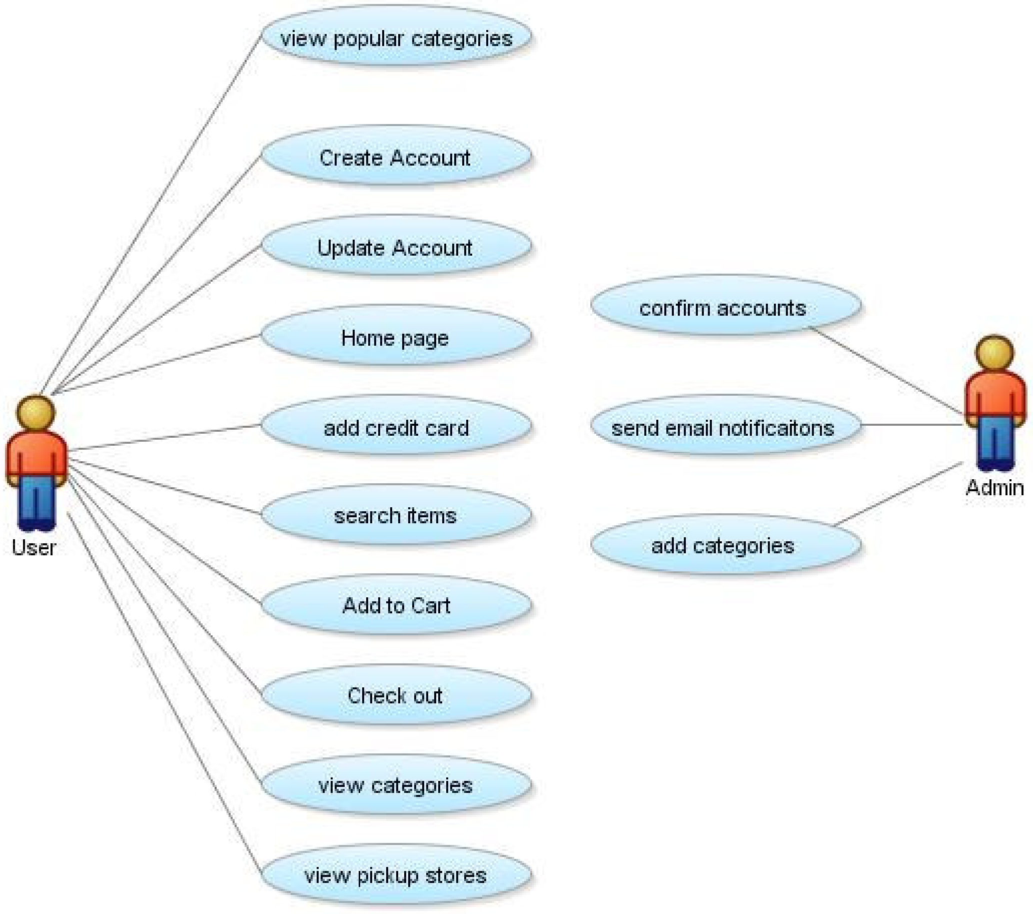
3.1.3 Business Logic (Web Services): Business logic is written as web services. For ASP.NET, Visual Studio 2005 is used for development and deployed over IIS 5.1 server. For J2EE, NetBeans is the IDE used to develop the web services and then they are deployed over the Sun Application Server 9. Website also used various other helpful APIs and client side scripts. Google Maps API is used to display the online store locations and prices at those locations. Flying basket shopping cart script is used for effect of product flying into the cart when it is added. Google search web service is used to display search results which describes more about the products. Images are sent from the web servers as byte arrays.

3.1.4 Database: The database used in the application is Oracle 10g. The database schema will be explained in the later section.

3.2 Use Case Diagram:

Figure 3.2 Use Case Diagram:

Use case diagrams describe what a system does from the standpoint of an external observer. The use case diagram above figure depicts the relationships among the actors and use cases where actors represent the external entities of the system and use cases represent the functional parts of the system.

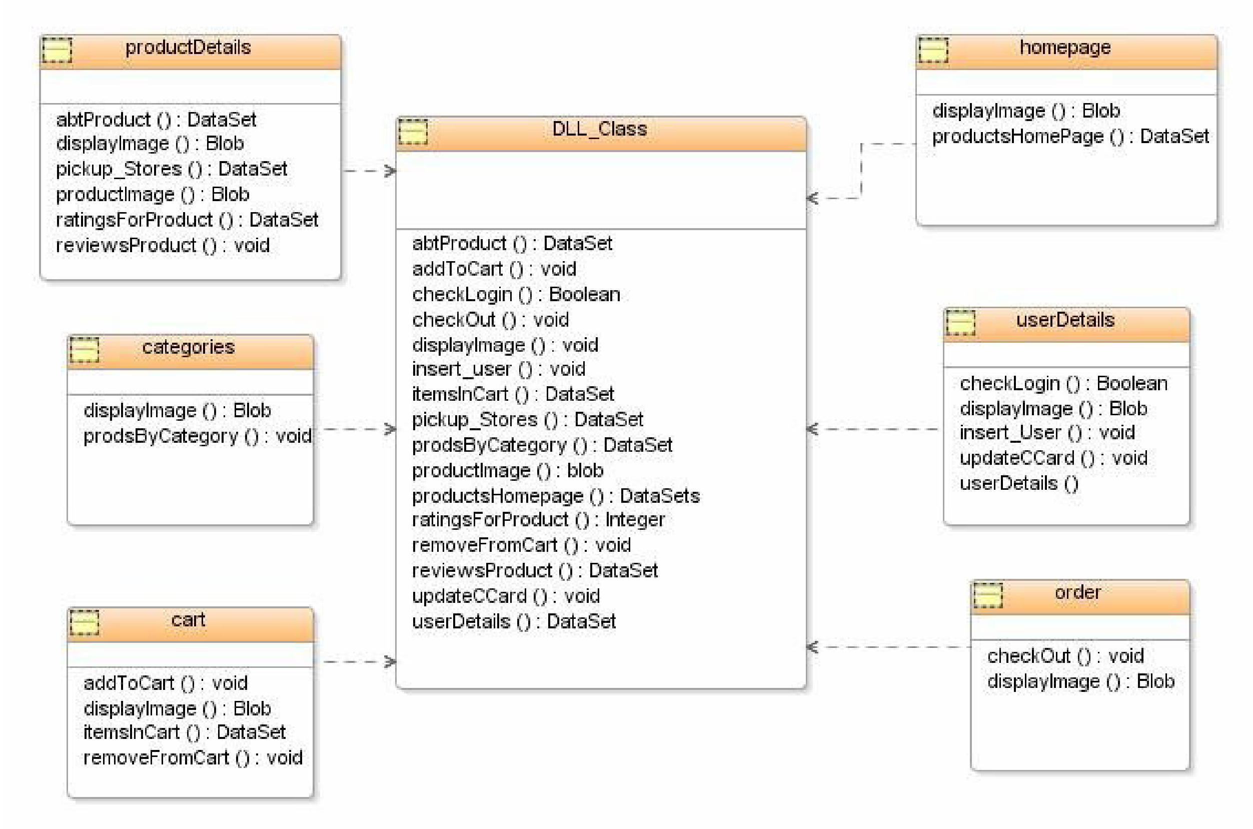


3.3 Class Diagram:

DLL_Class is the main class which has all the functions which have been developed separately. The other six classes are the classes used in the online store website. These classes extend the DLL_Class and utilize all the functions of the DLL_Class.

‘userDetails’ class uses checkLogin(), insert_user(), updateCCard() and userDetails() for verifying the Login information, adding a new user, changing the old credit card with

Figure 3.3 Class Diagram



new credit card and displaying the user's details respectively.

'Categories' class has one important function which displays the products belonging to a given category.

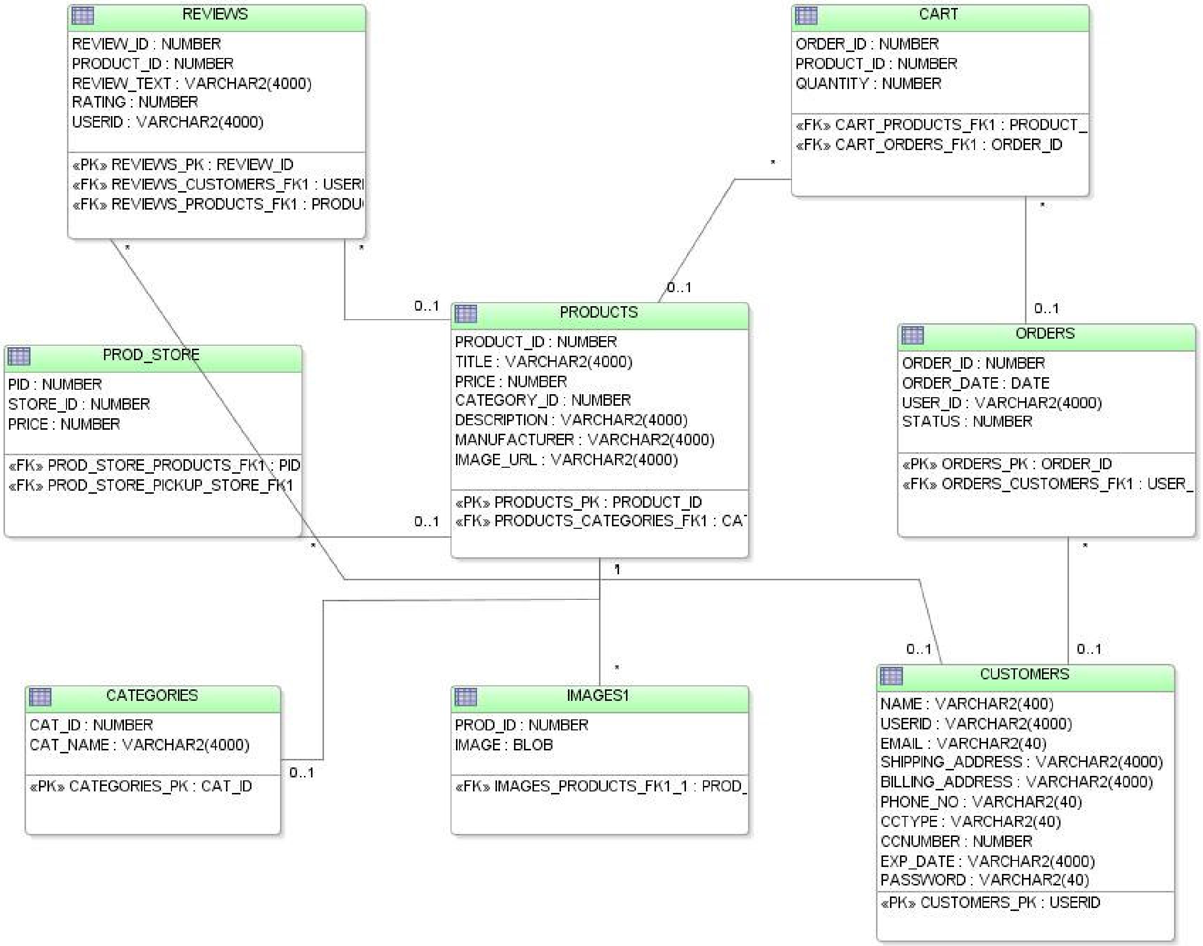
'productDetails' class has 6 functions. 'abtProduct' function gives all the details about the product. productImage() retrieves the image from the database where as displayImage() displays this image. RatingsForProduct() and reviewsProduct() gives the rating out of 5 and review texts for the product respectively.

'Order' class has two important functions. One is to find out the items in the cart and the other one is to check out the ordered items.

‘Cart’ class has four important functions. addToCart and removeFromCart functions are used to add an item and remove an item respectively. itemsInCart function is used to find out items in the Cart.

3.4 Database Diagram:

Figure 3.4 Database Schema:


REVIEWS
REVIEW_ID : NUMBER
PRODUCT_ID : NUMBER
REVIEW_TEXT : VARCHAR2(4000)
RATING : NUMBER
USERID : VARCHAR2(4000)
«PK» REVIEWS_PK : REVIEW_ID
«FK» REVIEWS_CUSTOMERS_FK1 : USERI
«FK» REVIEWS_PRODUCTS_FK1 : PRODU
CART
ORDER_ID : NUMBER
PRODUCT_ID : NUMBER
QUANTITY : NUMBER
«FK» CART_PRODUCTS_FK1 : PRODUCT_
«FK» CART_ORDERS_FK1 : ORDER_ID
PROD_STORE
PID : NUMBER
STORE_ID : NUMBER
PRICE : NUMBER
«FK» PROD_STORE_PRODUCTS_FK1 : PID
«FK» PROD_STORE_PICKUP_STORE_FK1
PRODUCTS
PRODUCT_ID : NUMBER
TITLE : VARCHAR2(4000)
PRICE : NUMBER
CATEGORY_ID : NUMBER
DESCRIPTION : VARCHAR2(4000)
MANUFACTURER : VARCHAR2(4000)
IMAGE_URL : VARCHAR2(4000)
«PK» PRODUCTS_PK : PRODUCT_ID
«FK» PRODUCTS_CATEGORIES_FK1 : CA
ORDERS
ORDER_ID : NUMBER
ORDER_DATE : DATE
USER_ID : VARCHAR2(4000)
STATUS : NUMBER
«PK» ORDERS_PK : ORDER_ID
«FK» ORDERS_CUSTOMERS_FK1 : USER_
CATEGORIES
CAT_ID : NUMBER
CAT_NAME : VARCHAR2(4000)
«PK» CATEGORIES_PK : CAT_ID
IMAGES1
PROD_ID : NUMBER
IMAGE : BLOB
«FK» IMAGES_PRODUCTS_FK1_1 : PROD_
CUSTOMERS
NAME : VARCHAR2(400)
USERID : VARCHAR2(4000)
EMAIL : VARCHAR2(40)
SHIPPING_ADDRESS : VARCHAR2(4000)
BILLING_ADDRESS : VARCHAR2(4000)
PHONE_NO : VARCHAR2(40)
CCTYPE : VARCHAR2(40)
CCNUMBER : NUMBER
EXP_DATE : VARCHAR2(4000)
PASSWORD : VARCHAR2(40)
«PK» CUSTOMERS_PK : USERID
*
*
*
*
*
*
*
0..1
0..1
0..1
0..1
0..1
0..1
0..1
1

I have used Oracle 10g as the database which ran on the local computer. The diagram below shows the database schema In addition to the tables, there are stored procedures, triggers and sequences. I have created sequences for all tables which require primary key. Triggers helped me to make multiple modifications based on one DML statement. Stored procedures increased the performance of the homepage, as multiple result sets can be retrieved from a single database call.

3.4.1 Stored Procedures:

Add_Image: This procedure is used to upload the image of the product into the database. The image is stored as binary large object (BLOB) in the database.

Default_Webpage: This procedure retrieves products from the three categories as multiple result sets. Cursors are used in this procedure to execute three different queries.

Add_Product: This procedure is used to add a product to the cart. If the product to be added is the first product in the cart, a new order should be created. Each order is given an unique order id. This order id is generated using sequence. The status of the order will be '1' i.e. user is in the process of buying other items. If the product to be added is not the first product, then status of the order by that user will be checked. If the status is '1', then the product is added with same order id. If the user is done with shopping and wants to pay the bill, then the user will change the status of the order to '2' i.e. checked out.

Insert_User: This stored procedure is used to add new user's information to the database. A trigger is used to check if a user already exists with the given userid. In this way, uniqueness of the user ids will be maintained.

Remove_Product: This stored procedure is used to remove a product from the shopping cart. If the product to be removed is the only product in the cart, then the order is deleted from the order table. If the cart contains more than 1 product, then the product is just removed from the cart table.

Update_User: This stored procedure is used to update the user's information. The user cannot change his user id during this process.

3.4.2 Triggers:

Before_insert_category: This trigger is used to insert the primary key value for category table. This is done for each addition of new category.

Before_insert_order: This trigger is used to create a new order for a user. A new order is created by inserting a new unique order id into the table. s

Before_insert_user: This trigger is used to check if a user exists already with the given user id.

Before_insert_review: This trigger is used to insert the primary key value for review table. This is done for each new review of a product.

Before_insert_product: This trigger is used to insert the primary key value for products table. This is done for each new product added to the table.

3.4.3 Sequences:

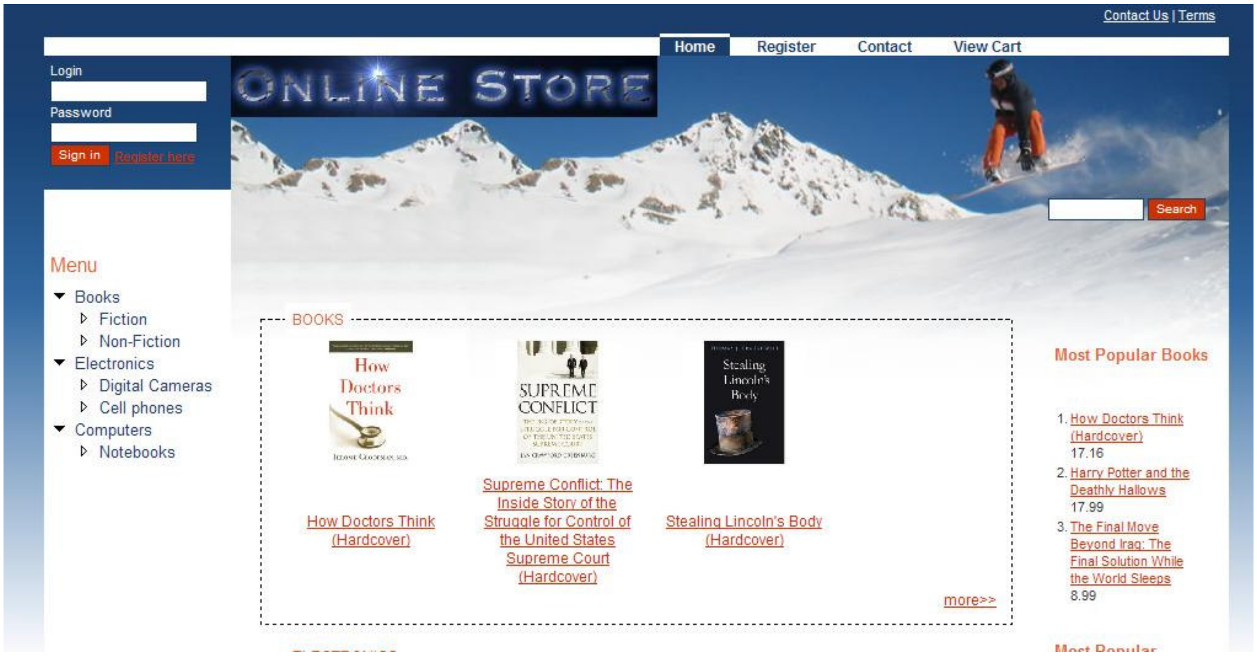
Category_seq, Order_seq, Product_seq and Review_seq are the sequences in the database. Sequences are used to generate primary keys for the tables in the database.

3.5 Website Screenshots:

3.5.1 Home Page:

This is the home page of my website. The home page displays the menu of categories. It displays the recently added three products of each category. It also contains popular products of each category. The "popularity" of the products is evaluated from the reviews given by the customers.

Figure 3.5 Home page Screenshot

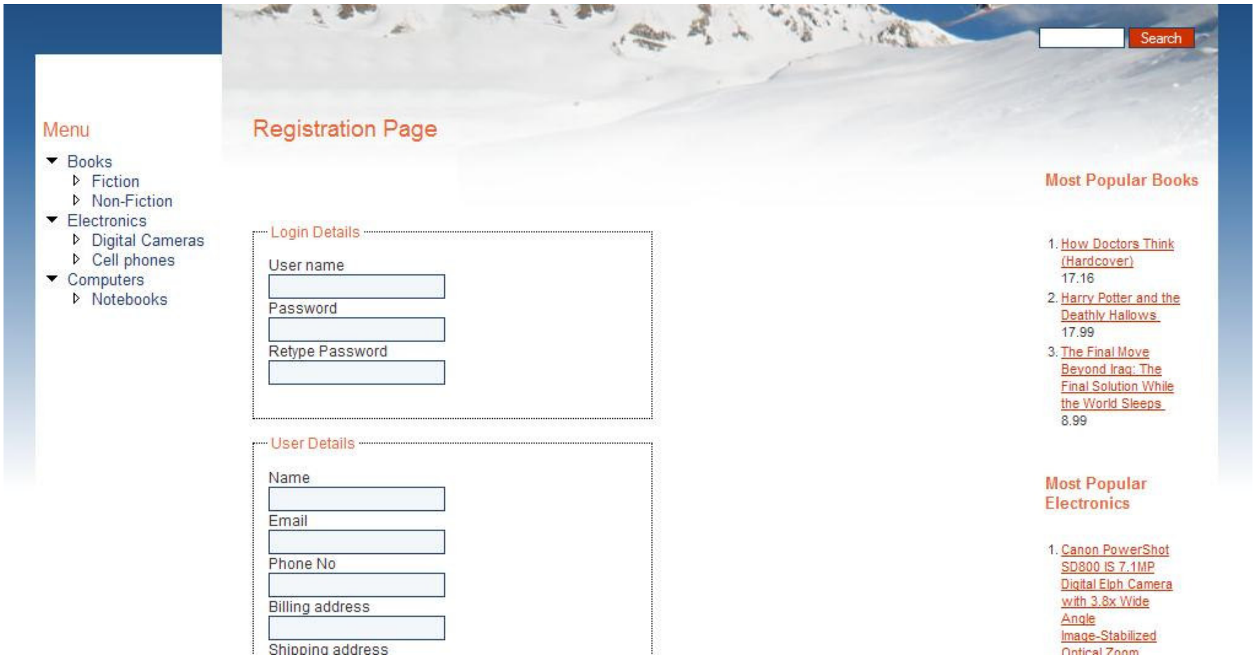

Contact Us | Terms
Home
Register
Contact
View Cart
Login
Password
Sign in
Register here
ONLINE STORE
Search
Menu
Books
Fiction
Non-Fiction
Electronics
Digital Cameras
Cell phones
Computers
Notebooks
BOOKS
How Doctors Think
Supreme Conflict
Stealing Lincoln's Body
How Doctors Think (Hardcover)
Supreme Conflict: The Inside Story of the Struggle for Control of the United States Supreme Court (Hardcover)
Stealing Lincoln's Body (Hardcover)
more>>
Most Popular Books
1. How Doctors Think (Hardcover)
17.16
2. Harry Potter and the Deathly Hallows
17.99
3. The Final Move Beyond Iraq: The Final Solution While the World Sleeps
8.99
Most Popular

3.5.2 Registration:

This is the registration page of the website. Users can register themselves for an account here. Users will enter the credit card information as part of the registration process. Users can also change their credit card during the check out.

After registering to the website customers can login by providing their email and password and start shopping after a successful authentication.

Figure 3.6 Register page Screenshot



Most Popular Books

1. How Doctors Think (Hardcover)
17.16
2. Harry Potter and the Deathly Hallows
17.99
3. The Final Move Beyond Iraq: The Final Solution While the World Sleeps
8.99

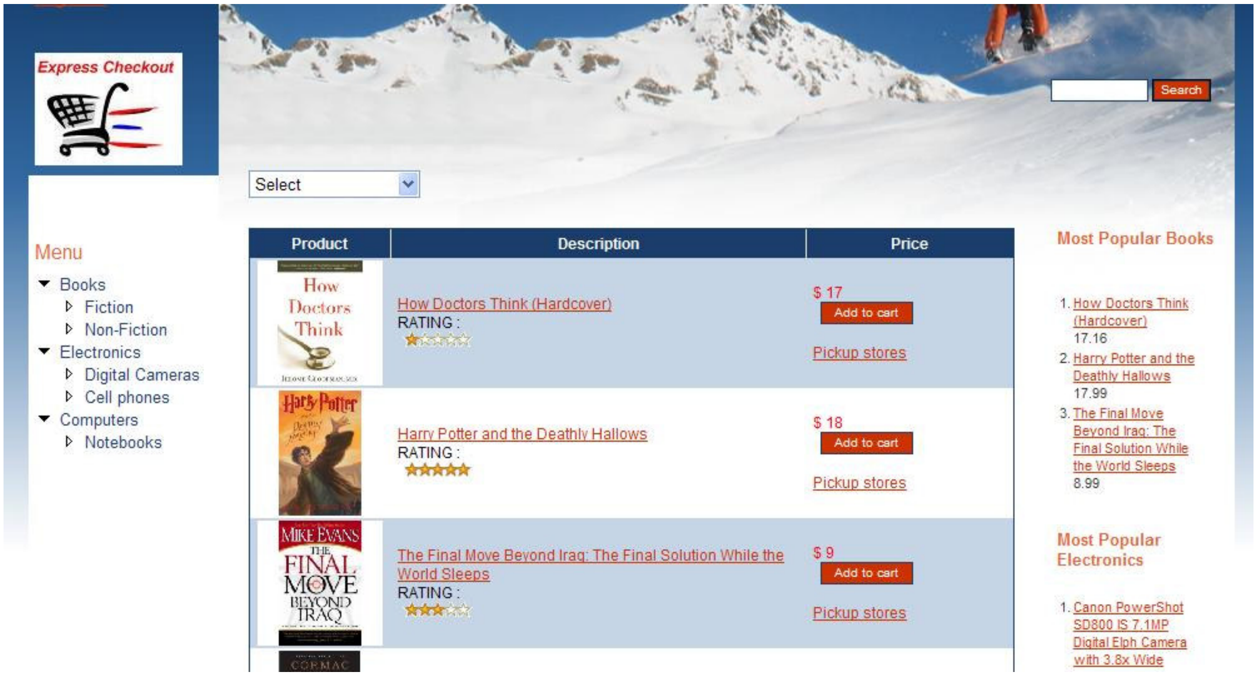
Most Popular Electronics

1. Canon PowerShot SD800 IS 7.1MP Digital Elph Camera with 3.8x Wide Angle Image-Stabilized Optical Zoom

3.5.3 Categories:

This web page displays products belonging to a particular category. There are three main categories. They are books, electronics and computers. There are sub categories for these categories. Books have fiction and non-fiction, electronics consists of cameras and cell phones, computers consists of notebooks.

Figure 3.7 Categories Screenshot



Product	Description	Price
How Doctors Think	How Doctors Think (Hardcover) RATING :	$ 17 Add to cart Pickup stores
Harry Potter	Harry Potter and the Deathly Hallows RATING :	$ 18 Add to cart Pickup stores
Mike Evans The Final Move Beyond Iraq	The Final Move Beyond Iraq: The Final Solution While the World Sleeps RATING :	$ 9 Add to cart Pickup stores

Most Popular Books

1. How Doctors Think (Hardcover) 17.16
2. Harry Potter and the Deathly Hallows 17.99
3. The Final Move Beyond Iraq: The Final Solution While the World Sleeps 8.99

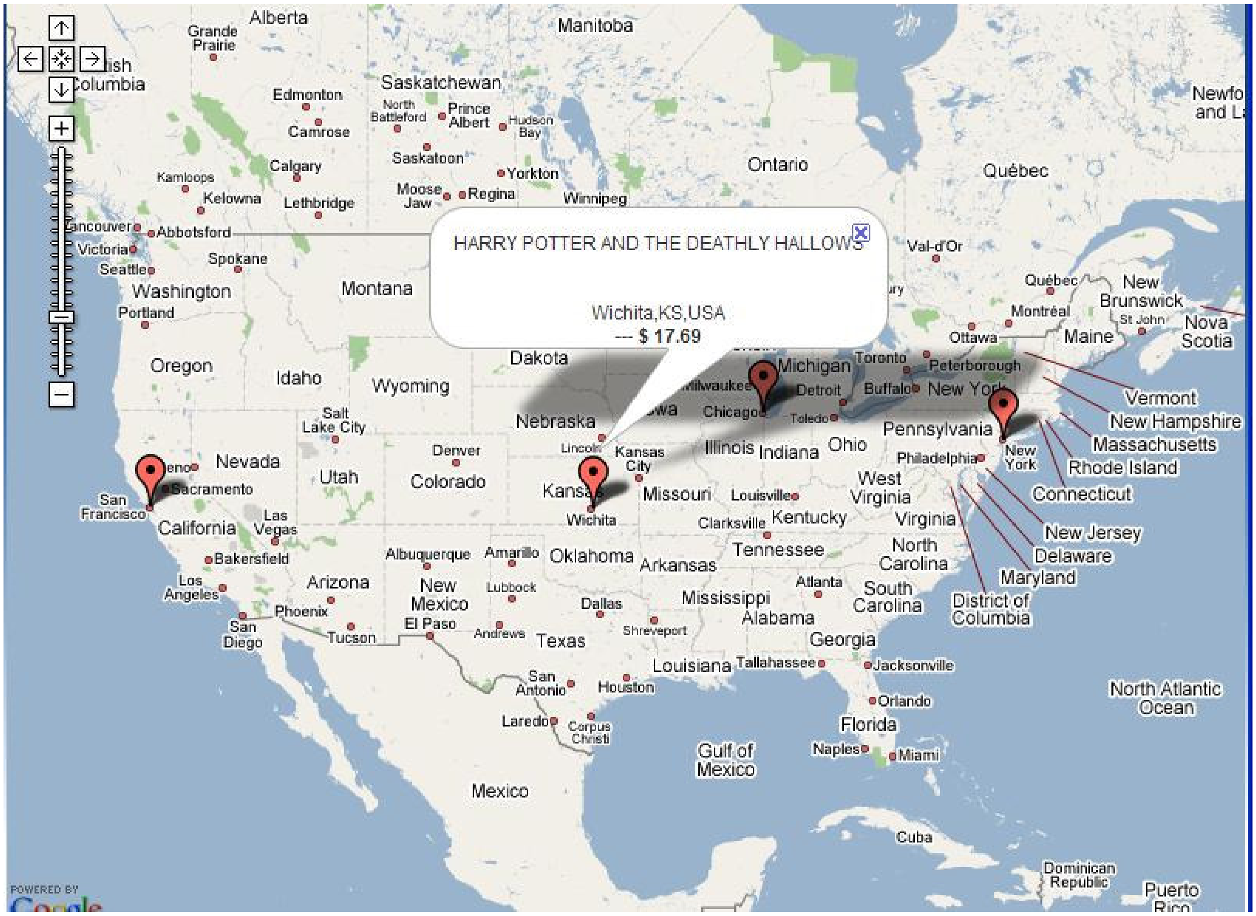
Most Popular Electronics

1. Canon PowerShot SD800 IS 7.1MP Digital Elph Camera with 3.8x Wide

3.5.4 Pickup stores:

This web page displays pick up locations for a particular product. Pick up stores are displayed as markers on the google maps. Users can check the price of the product by clicking on the marker.

Figure 3.8 Pickup stores Screenshot

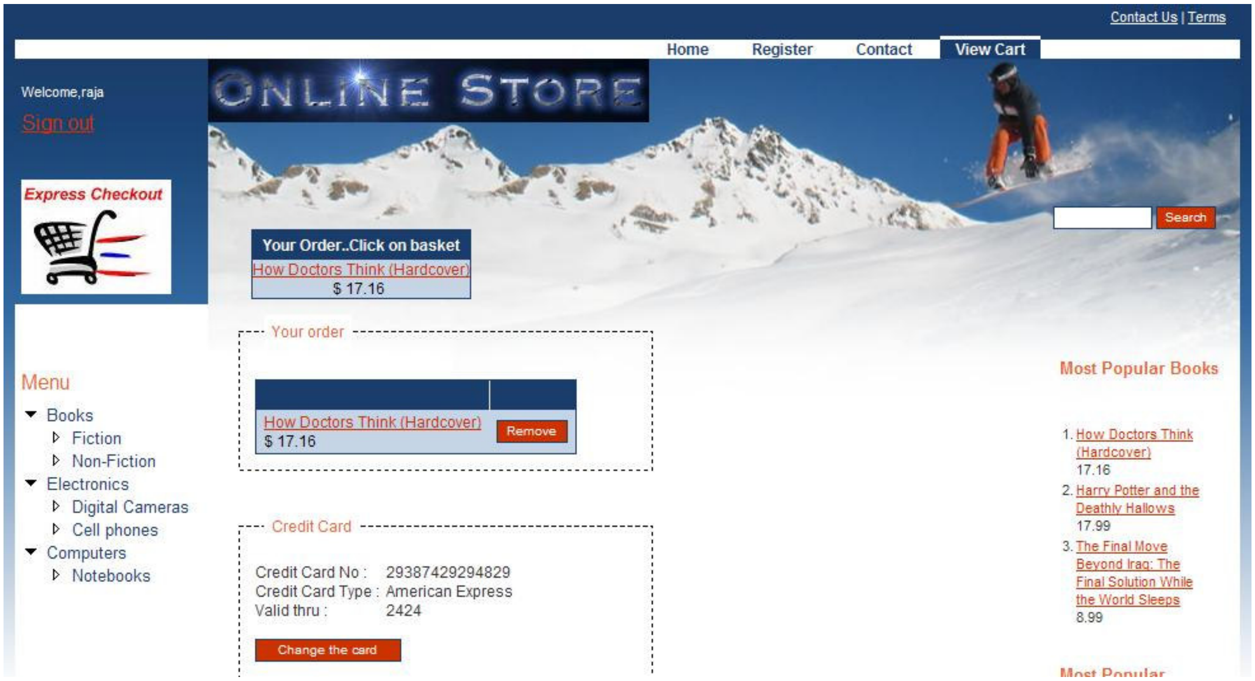

HARRY POTTER AND THE DEATHLY HALLOWS

Wichita,KS,USA
--- $ 17.69

3.5.5 Cart:

This web page shows the dynamic cart as well as the total order in a web page. At any time during shopping, Users can check the cart by just placing the mouse on the cart. By clicking on the cart, it takes the user to the cart page. Here, user can place the order for products in the card. User can also change the credit card information here.

Figure 3.9 Cart web page Screen shot



Welcome,raja

Your Order..Click on basket

How Doctors Think (Hardcover)
$ 17.16

Your order

How Doctors Think (Hardcover)
$ 17.16

Credit Card

Credit Card No : 29387429294829
Credit Card Type : American Express
Valid thru : 2424

Most Popular Books

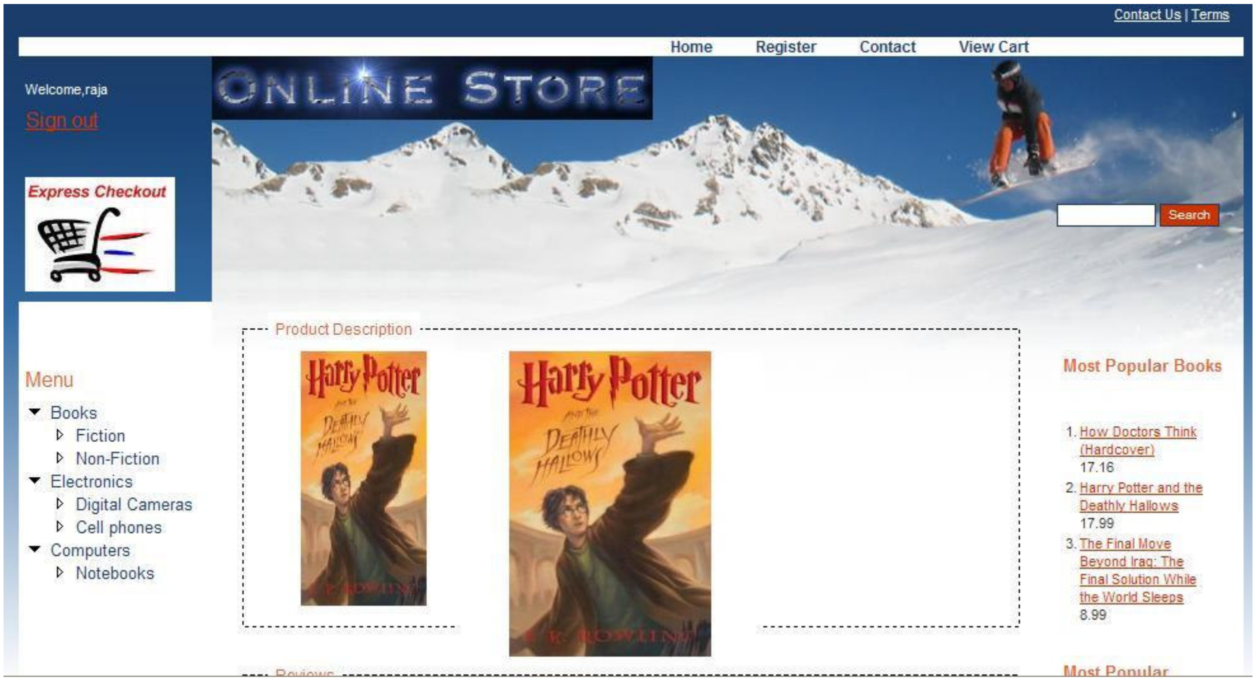
1. How Doctors Think (Hardcover)
   17.16
2. Harry Potter and the Deathly Hallows
   17.99
3. The Final Move Beyond Iraq: The Final Solution While the World Sleeps
   8.99

Most Popular

3.5.6 Image enlargement dynamically:

This web page gives the details and description of a product. The image of the product will be dynamically enlarged by moving the mouse cursor over it.

Figure 3.10 Product Image Screenshot



Most Popular Books

1. How Doctors Think (Hardcover)
   17.16
2. Harry Potter and the Deathly Hallows
   17.99
3. The Final Move Beyond Iraq: The Final Solution While the World Sleeps
   8.99

CHAPTER 4 - Evaluation

4.1 Performance Testing:

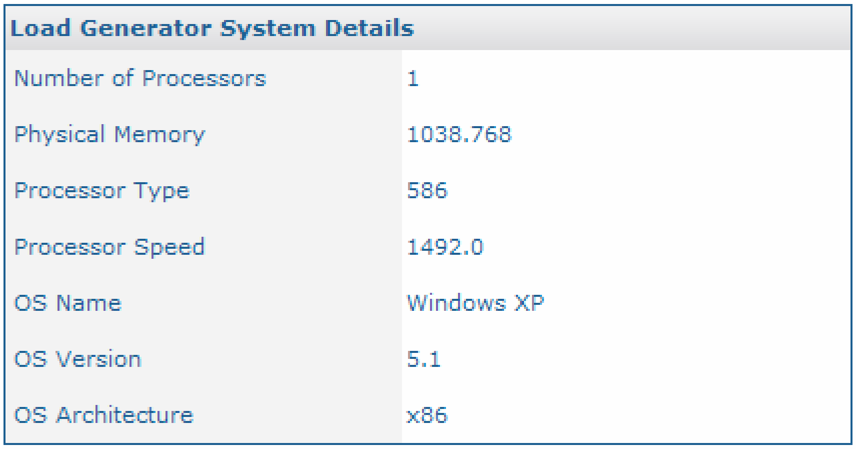
Today most websites and web applications function properly as long as a single user is accessing them. If several thousand users are going to access the website per day, then web performance testing ensures that all the users get appropriate results in an acceptable time. Web Performance testing helps you to gather the performance and stability information of your website by realistically simulating load in your web site.

4.1.1 Tool:

I have used Apache JMeter to perform load testing for most of the test suites. Apache JMeter is a 100% pure Java desktop application designed to load test functional behavior and measure performance [5]. It may 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. JMeter does not execute the Javascript found in HTML pages. Nor does it render the HTML pages as a browser does (it's possible to view the response as HTML etc, but the timings are not included in any samples, and only one sample in one thread is ever viewed at a time). JMeter can be used as a unit test for JDBC database connections, FTP, LDAP, Web services, JMS, and generic TCP connections. JMeter can also be configured as a monitor, although this is typically considered an ad-hoc solution in lieu of advanced monitoring solutions. I have done remote testing using two computers with one as a server and the other computer has JMeter client.

4.1.2 System Configuration:

Figure 4.1 System Configuration



Load Generator System Details	
Number of Processors	1
Physical Memory	1038.768
Processor Type	586
Processor Speed	1492.0
OS Name	Windows XP
OS Version	5.1
OS Architecture	x86

4.1.3 Test Plan:

All the tests were conducted on the machine having the configuration above. While testing J2EE web services, both IIS and Sun App Server were running. This might have put some extra effort on the CPU while calculating the results for J2EE web services results.

The test plan is as follows:

Every test run is done using 5 users sending out 200 requests. Evaluation of results is done based on two factors. They are throughput and response time.

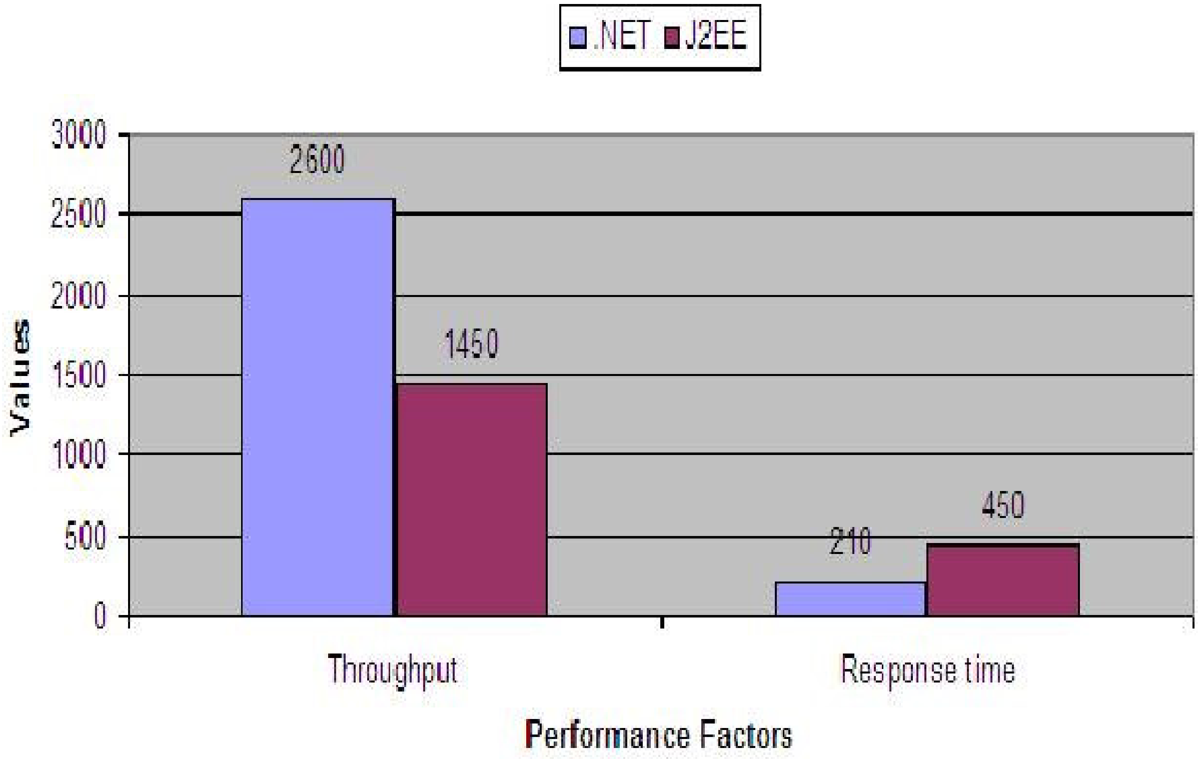
First, Home page is tested and its behavior will be observed to get a general idea about the performance of the application.

In the next section, we will see how this home page reacts in three network conditions: server and browser running on the same machine, server and browser separated by 100Mb Ethernet and finally, server and browser running on our K-State wireless network.

4.1.4 Test Results & Analysis:

4.1.4.1 Home page:

Figure 4.2 Home page performance



From the results above, we can see that ASP.NET web service responds faster than an EJB web service. The performance of a .NET – J2EE architecture is lower than the .NET - .NET architecture due to following bottlenecks.

In J2EE - .NET architecture, there is an additional overhead of xml wrapping and unwrapping. Data is retrieved as a result set from the database and wrapped up in xml format and passed to the client. The client then, has to parse the xml string and convert it into .NET's Data

Set. The low performance of the .NET – J2EE page can be attributed to this overhead of xml wrapping/unwrapping.

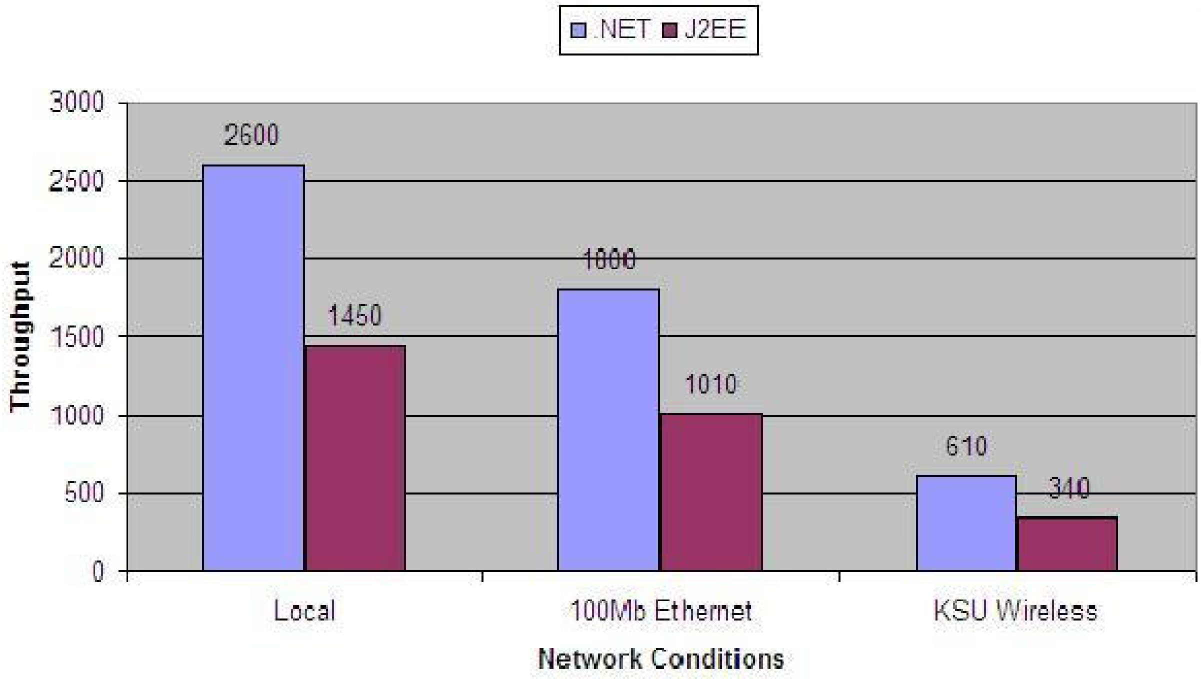
For J2EE - .NET architecture, the decrease in the performance can be attributed to the EJB container.

4.1.4.2 Performance under different network conditions:

The following diagrams represent the effect of network on the throughput and response time of the IIS web server and sun application server. Server is tested under three conditions: Server and browser running on the same machine, server and client separated by 100Mb Ethernet card and finally, server and client are in the K-State Wireless Network.

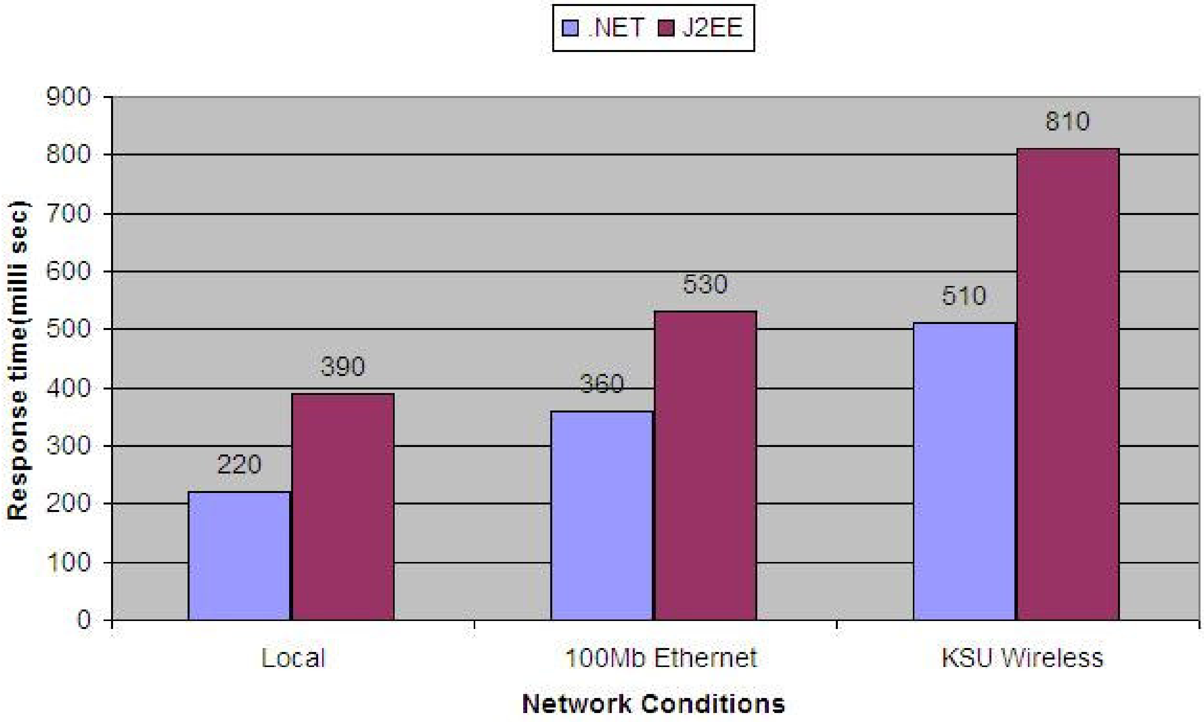
For J2EE web services, an important point to note is that both the web servers reside on the same machine. So for the cross platform testing there is no real network bottleneck.

Figure 4.3 Effect of Network conditions on throughput



From the response times' graph below, we can see that clients which access .Net web services download faster than clients which access J2EE web service.

Figure 4.4 Effect of Network conditions on response time



From observing the performance results, the .NET – J2EE application is no less efficient than the .NET-.NET application in terms of functionality. But performance is a key issue for the J2EE - .NET application considering the fact that it occupies most of the resources available for processing. The main overhead in the .NET - J2EE application is that the data needs to be wrapped in XML tags before sending it to the .NET presentation tier and then unwrap it from the XML tags at the client side and present to the user.

.

4.2 Unit Testing:

The functionality of the website depends upon the correctness of the web services in responding to the requests with a suitable and legitimate response. So in order to check the functionality of the website black box testing of the web services is needed.

4.2.1 Tool:

NUnit 2.2 is an excellent unit-testing framework for all .NET languages. It is written in C# and has been completely redesigned to take advantage of many .NET language features, for example custom attributes and other reflection related capabilities [6]. For NUnit testing, we have to create test cases for the functions to test in our web-application. We have to compile our project once we are finished writing test cases and load the project DLL file into NUnit GUI.

4.2.2 Test Results:

I have tested some of my basic functions. Following are the functions I have tested using NUnit:

i. Check if the user login (Username and password) match correctly.

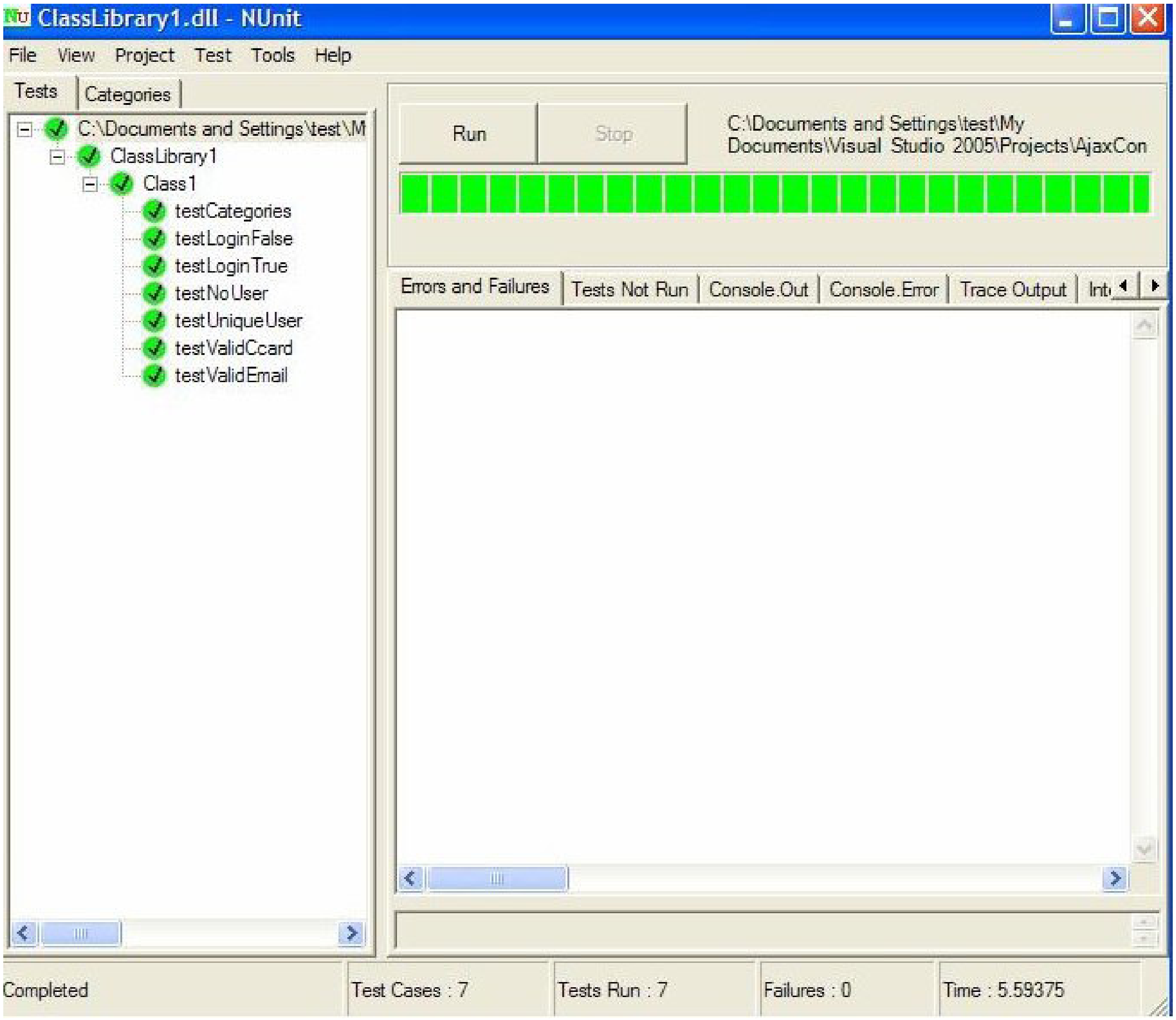
ii. Check if all the categories in the database are properly retrieved.

iii. Check if no user exists with the given userid.

iv. Check if given text is a valid email address or not.

v. Check if given credit card number is valid or not.

Figure 4.5 NUnit screen shot



C:\Documents and Settings\test\M
ClassLibrary1
Class1
testCategories
testLoginFalse
testLoginTrue
testNoUser
testUniqueUser
testValidCcard
testValidEmail

C:\Documents and Settings\test\My Documents\Visual Studio 2005\Projects\AjaxCon

Completed | Test Cases : 7 | Tests Run : 7 | Failures : 0 | Time : 5.59375

CHAPTER 5 - Conclusions and Future Work

5.1 Conclusions:

By building this E-Commerce website I had gained a thorough knowledge about both .NET and J2EE technologies. I have got hands-on experience on various tools and IDEs.

The purpose of the website is to provide the user with similar overall experience regardless of the technology. Interoperability of platforms is becoming more and more important to many organizations these days as many standards have been established for exchanging data between different platforms. The objective of the project is to determine the feasibility in accessing the business logic in J2EE from an ASP.NET platform. The project proves that instead of taking a technology-aligned development for applications, it would prove to be more productive and rewarding to consider the interoperability of different platforms.

5.2 Problems faced during the project:

Web services are platform independent and they can only return basic data types like Integer, String and Boolean etc. However, data types like result set (J2EE) and data set (ASP.NET) are frequently used while retrieving data from the database. A direct typecasting between the two is not possible as web services only support limited basic data types. For converting result set to data set, the resultset is converted into xml string and output as string. This string is retrieved and parsed to retrieve the database results.

Similarly, web services cannot transfer images directly. They are converted into byte streams and then they are sent to the client.

For IIS server 5.1 running on Windows XP OS, there is a 10 connection limit at the OS level. So, server was dropping many of the browser's requests. Increasing the connection limit did the trick.

As there are multiple servers and IDEs running during most of the development time, it occasionally got my system slowed down.

5.3 Future Work:

- Encrypting the user's credit card information.
- Add flash animations to enrich the user experience.
- To have recommendation list for the customers, based on their previous selling habits.
- To modify the search capability so that users will get more relevant search results.
- Customers can have the option of choosing multiple credit cards from a single user account.
- For this project, both the web servers run on the same machine. Running these web servers on two different machines and then evaluating performance might give more interesting results.

5.4 References :

[1] ASP.NET Design Starter Kit Templates. *MSDN*. At **http://msdn2.microsoft.com/en-us/asp.net/aa336613.aspx** Accessed on 10th July 2007

[2] Scott Mitchell. *Searching Google Using the Google Web Service.* At **http://aspnet.4guysfromrolla.com/articles/030503-1.aspx** Accessed on 13th July 2007

[3] Rob Howard. *10 Tips for Writing High-Performance Web Applications.* At **http://msdn.microsoft.com/msdnmag/issues/05/01/ASPNETPerformance/** *Accessed on 17th July 2007*

[4] Wikipedia. *ASP.NET.* At **http://en.wikipedia.org/wiki/ASP.NET** Accessed on 30th July2007.

[5] Wikipedia. *JMeter*. At **http://en.wikipedia.org/wiki/Jmeter** Accessed on 30th July 2007.

[6] Wikipedia. *NUnit.* At **http://en.wikipedia.org/wiki/Nunit** Accessed on 30th July 2007.

[7] Oracle Technology Network. *Database Development with JDeveloper.* At **http://www.oracle.com/technology/obe/obe1013jdev/dbdevelopment/obe_%20dbasedevmt.htm** *Accessed on 10th July 2007.*

[8] NetBeans IDE 4.1 Quick Start Guide for Web Services. *http://www.netbeans.org/kb/41/tutorial-webservice.html Accessed on 25th July 2007*

[9] Microsoft Visual Studio Professional 2005: Product Description http://www.amazon.com/Microsoft-Visual-Studio-Professional-2005/dp/B000BTA4LU *Accessed on 27th August 2007*

[10] Java EE Architecture http://www.insitechinc.com/what-is/WhatIsXTT_technologyPerspective.html *Accessed on 27th August 2007*

[11] Wikipedia. *J2EE definition and Architecture.* http://en.wikipedia.org/wiki/Java_EE *Accessed on 27th August 2007*

[12] The .NET Framework architecture http://msdn2.microsoft.com/en-us/library/ms973842.aspx *Accessed on 27th August 2007*

[13] Technologies - Core XML Specifications http://www.fujitsu.com/global/services/solutions/xml/tech/XML_wsdl.html