ONLINE BOOKS RESELLING WEBSITE PROJECT

Department: Computer Science and Engineering

Research paper on Online Books Reselling Website by

Kulla Vivek
Computer Science and Engineering
Lovely Professional University
Phagwara, India.

Venkata Mahesh Mopidevi
Computer Science and Engineering
Lovely Professional University
Phagwara, India.

Dintakurthi Kalki Vankata Subba Rao
Computer Science and Engineering
Lovely Professional University
Phagwara, India.

Anna Jaya Krishna
Computer Science and Engineering
Lovely Professional University
Phagwara, India.

Abstract

Nowadays everything is digitalized and everyone is using a smart phone, everything a person wants is in their hands because of the Online Marketing that is E-Commerce Websites which are offering a wide range of options or choices for the buyer to buy from them. This paper is about Online Books Reselling Website. This E-commerce site allows its users to sell their books o the other users and also to buy books which are for sale from the other users on the website. This is a very efficient way of buying a book which we need for a affordable rate and from a trusted seller or user, we can also sell the books which we don't need without

any hesitation of searching for person who likes to buy our book. This Site helps us to do everything from the tips of our fingers.

Keywords: E-Commerce site, Online Marketing, Digitalized

Paper Type: Research Work.

Introduction

In the world of software development there lots of improvement in the area of Architectural design and principles. The philosophies and implementation details are changing as the people guiding the development of the application. In this fantastic and yet sometimes complex world of software development there are some tried and true architecture patterns and software development guidelines employed by most architects. Also your design must have an ability to turn towards innovation instead of lending itself to common practices.

Web services are one such area where architects must lean on their creative side and hope that their solutions are still successful. In this report, we will explain an exciting voyage down the road of Web services application. From requirements to use cases, to database design, to component frameworks, to user interfaces, we will cover each and every aspect of system design required to build an application with collaborative Web services. The reason why we selected online Bookstore web service is everybody walking down the street has some idea about bookstores.

The objective of this project is to develop an e- bookstore where books can be bought from the comfort of home through the Internet. An online bookstore is a virtual store on the Internet where customers can browse the catalog and select books of interest. The selected books may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number.

Methodology

The system comprises of 2 major modules with their sub-modules as follows:

1. **User:**
   a. **Registration:** User need to register first with their basic registration details and nee to create a valid login id and password.
   b. **Login:** Using valid login credentials, user need to login into the system in order to access the system.

c. **View Books:** Once user is logged into the, he/she may view all the added books with their details.

d. **Buy Books:** User can surf various book of their choice and buy them.

e. **Sell Books:** System allows user to sell their books online buy added the book name and its details.

f. **Upload Books:** User need to upload a book if he/she wants to sell a book.

g. **My Orders:** All the purchase history of user will be displayed with details.

h. **View Buyer:** If any buyer is interested in buying a book or anyone buys a book from the user then buyer's details will be displayed.

2. **Admin:**

a. **Login:** Using valid login credentials, admin need to login into the system in order to access the system.

b. **View Books:** Admin can view all the added books online with their details.

c. **View Transaction:** System allows admin to view all the transaction details of buying and selling a book.

View User: All the registered user details will be displayed to the admin.

Proposed System

- Considering the anomalies in the existing system computerization of the whole activity is being suggested after initial analysis.

- The web application is developed using Asp.net with C# and SQL Server.

- Proposed system is accessed by two entities namely, Admin and Employee.

- Admin need to login with their valid login credentials first in order to access the web application.

- After successful login, admin can access all the modules and perform/manage each task accurately.

- User need to register with their basic registration details along with their valid login id and password details.

- Once a user is registered, he/she need to login using valid credentials and access the system.

- User can perform various tasks such as buying or a selling a book.

- In order to sell a book, users need to upload a book with its details such a name, cost, author and description.

- User can view the buyers who have purchased a book from the respective user.

- System allows user to buy a book using this online web portal.

- User can surf various books and view their details and also can buy a book if user is interested to buy.

- All the purchase history will be displayed to the user with details.
- System allows admin to view all the transaction history with details as well as can view all the registered user details

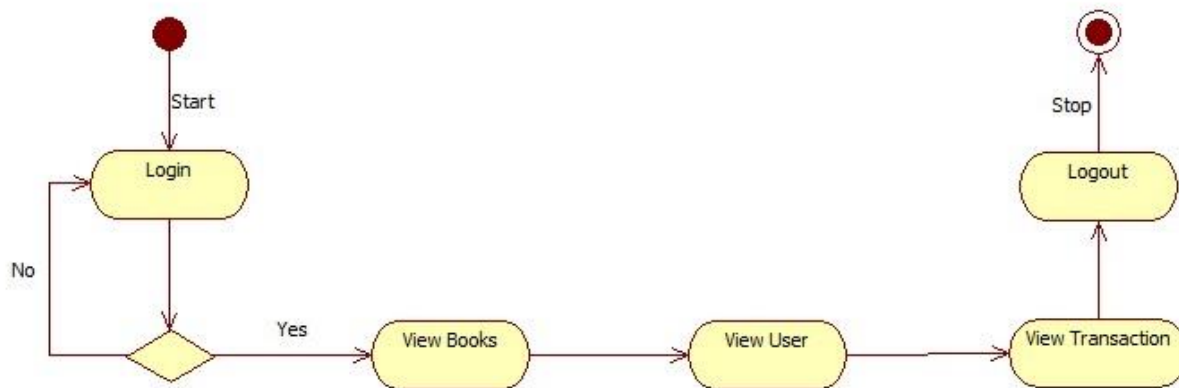**Project Design**

- **Activity Design**
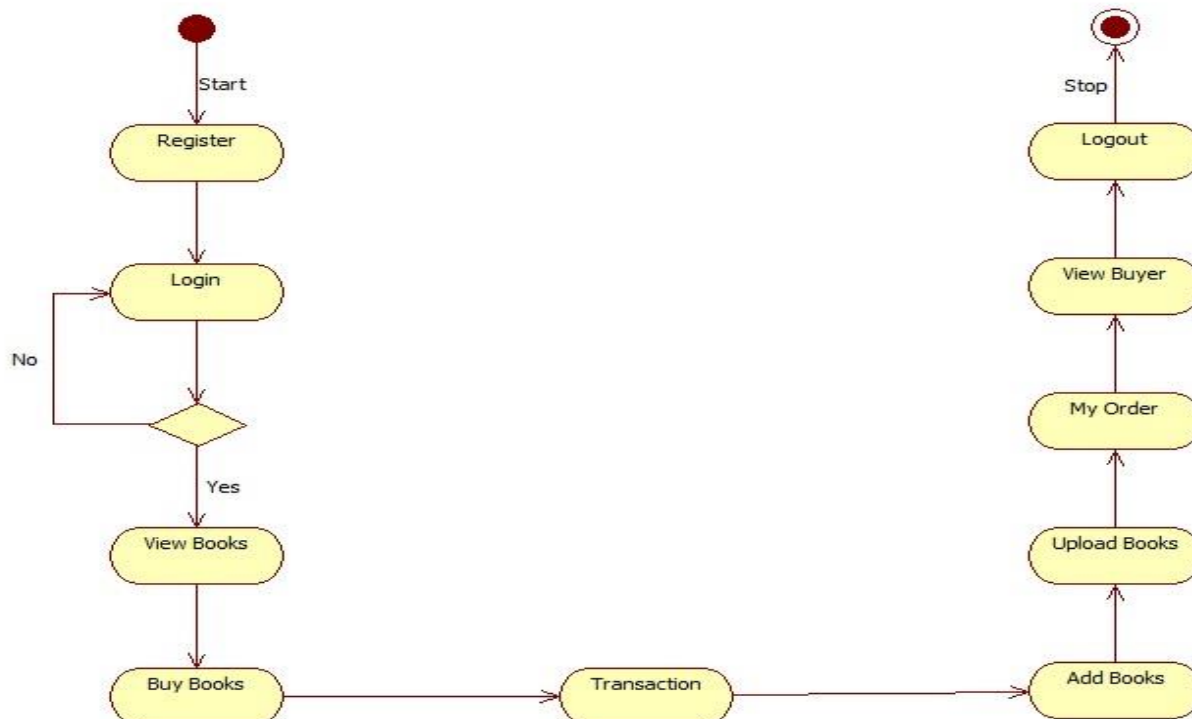

Fig.1.Activity Design of Admin
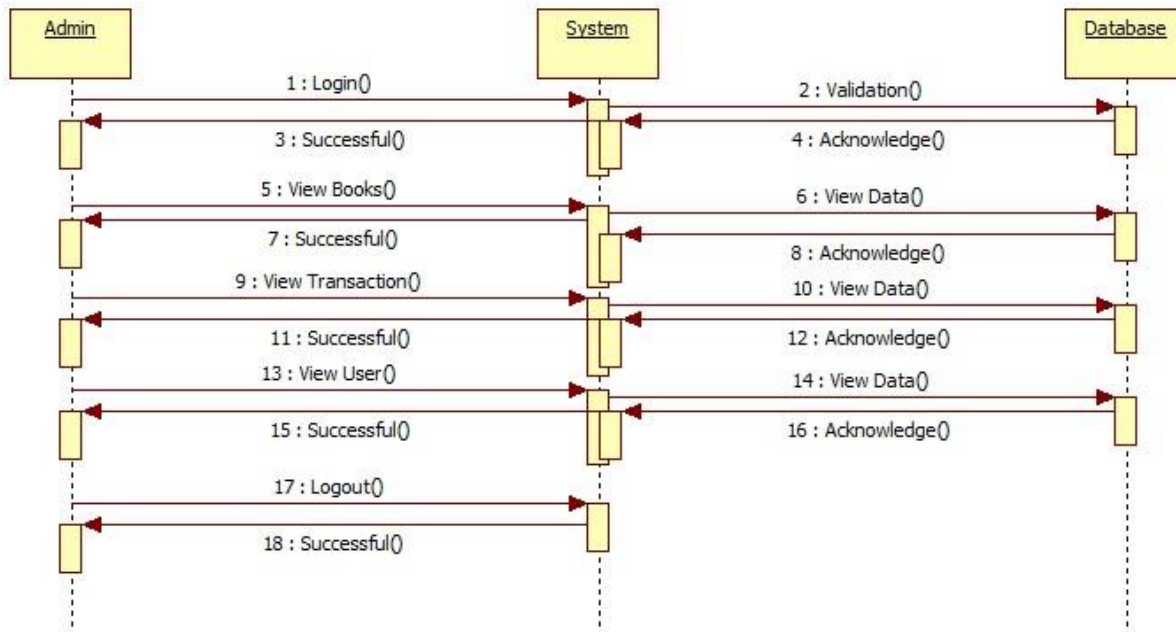

Fig.2.Activity Design of User

- **Sequence Design**

Fig.3.Sequence Design of Admin
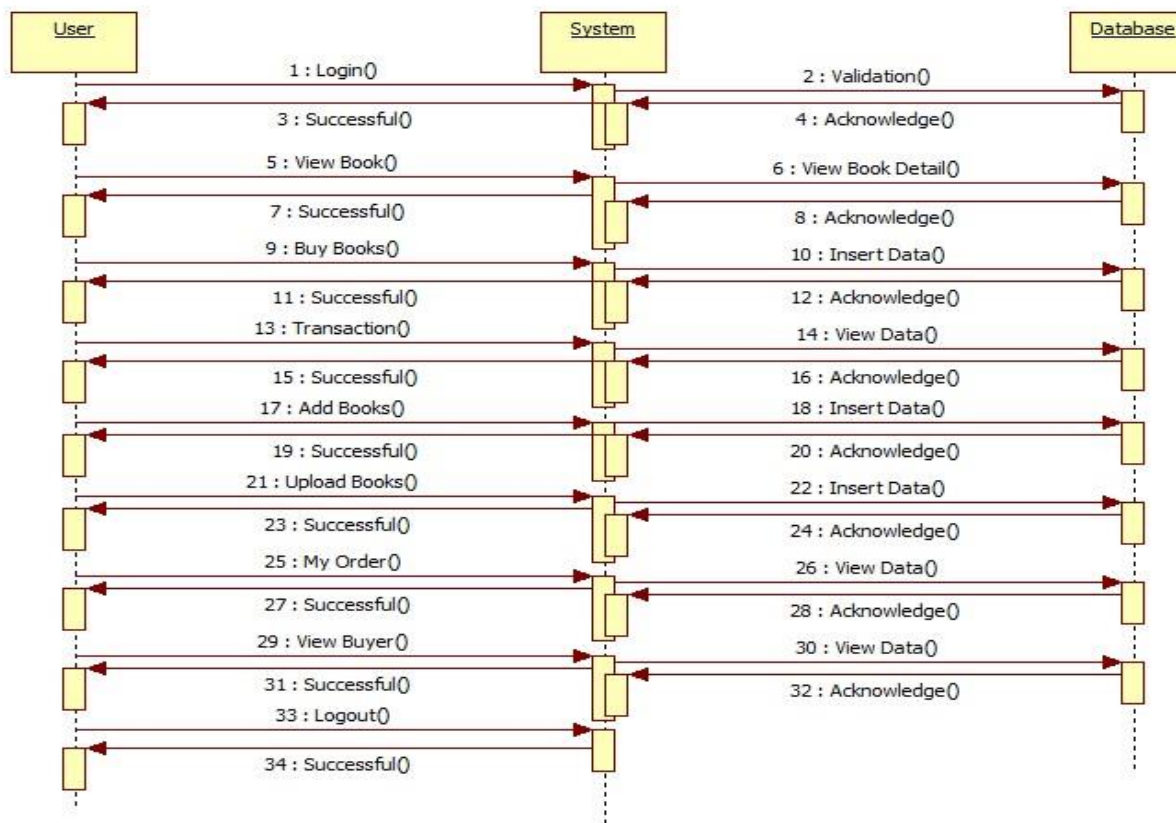
Fig.4.Sequence Design of User

Class Diagram

USER
- User_id   :   String
- Password   :   String
+ Login()
+ btn_Click ()
+ Logout ()

New Registration
- Name: String
- DOB   : Int
- Gender   : String
- Address   : String
- Mobile No. : Int
- Email id   : String
- User id   : String
- Password   : String
+ Submit ()
+ btn_Click ()

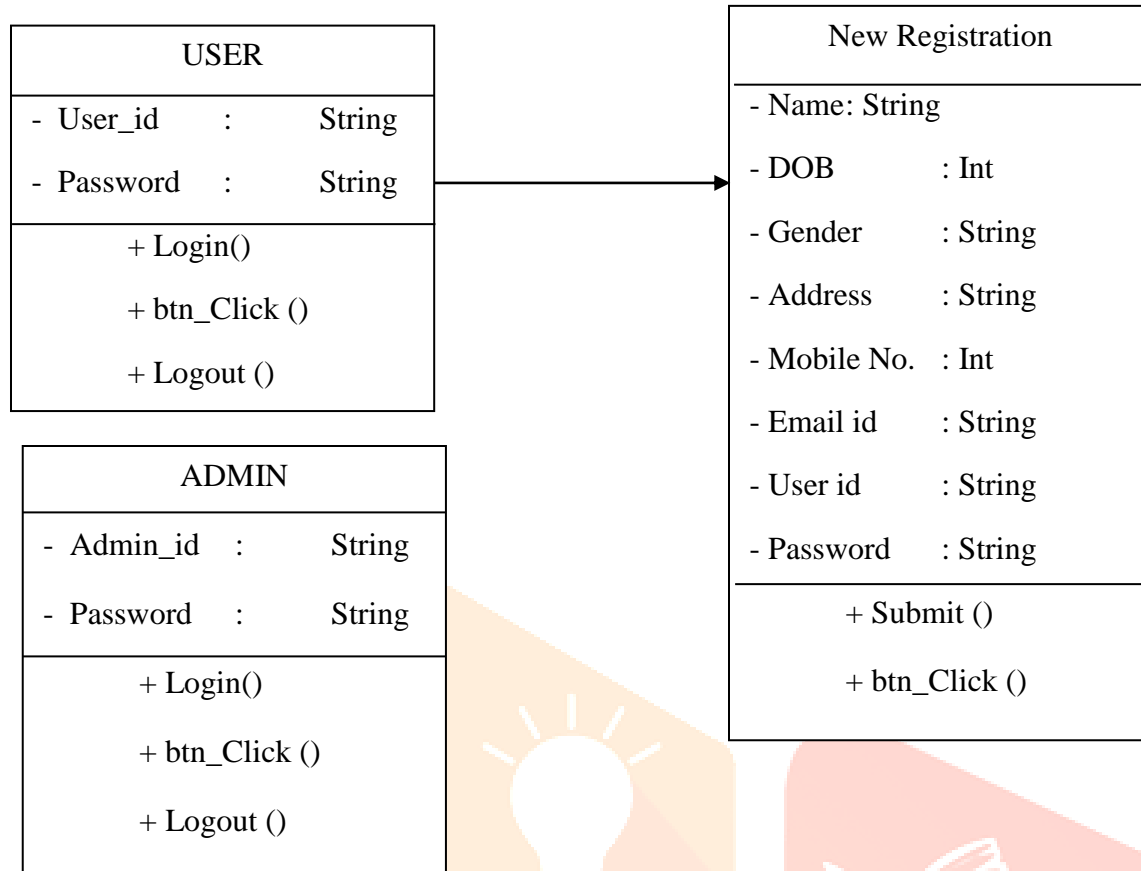| ADMIN |
| --- |
| - Admin_id   :   String |
| - Password   :   String |
| + Login() |
| + btn_Click () |
| + Logout () |
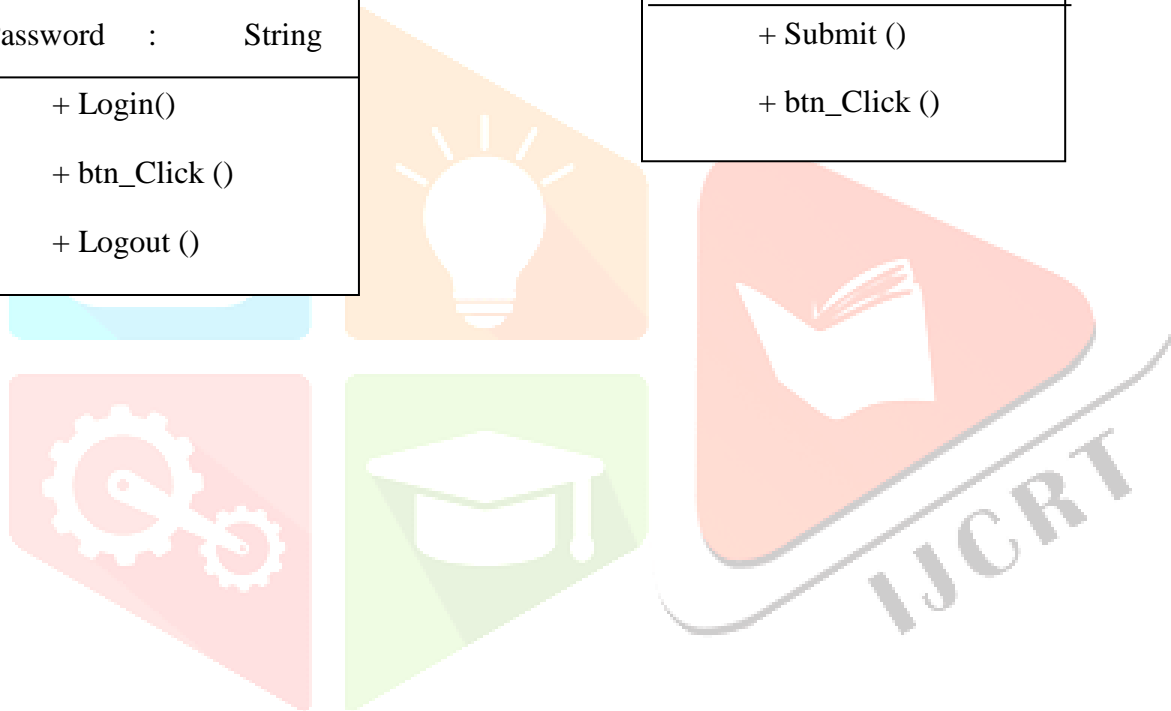
Implementation

- **Technology Used**

The Project is loaded in Visual Studio 2010. We used Visual Studio for Design and coding of project. Created and maintained all databases into SQL Server 2008, in that we create tables, write query for store data or record of project.

- **Hardware Requirement:**
    - i3 Processor Based Computer or higher
    - Memory: 1 GB RAM
    - Hard Drive: 50 GB
    - Monitor
    - Internet Connection

- **Software Requirement:**
    - Windows 7 or higher
    - Visual studio 2010.
    - SQL Server 2008.

Front End Technology

Microsoft .NET Framework

- The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet. The .NET Framework is designed to fulfill the following objectives:

- To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.

- To provide a code-execution environment that minimizes software deployment and versioning conflicts.

- To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party.

- To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.

- To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.

- To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code.

The.NET Framework consists of two key components: the common language runtime and the.NET Framework class library. The.NET Framework is built on top of the common language runtime. The runtime can be thought of as an agent that manages code at runtime, offering core services including memory management, thread management, and remote access, as well as enforcing strict type safety and other types of code accuracy that ensure security and robustness. The philosophy of code management is, in reality, a basic runtime theory. Managed code is code that targets the runtime, while unmanaged code is code that does not target the runtime. The class library, the.NET Framework's other key feature, is a large, object-oriented set of reusable types that you can use to create applications ranging from web to desktop applications, traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services.

Unmanaged components can host the.NET Framework by loading the common language runtime into their processes and initiating the execution of managed code, resulting in a software environment that can take advantage of both managed and unmanaged functionality. Not only does the.NET Framework provide many runtime hosts, but it also encourages the development of third-party runtime hosts.

For example, ASP.NET hosts the runtime to provide managed code with a scalable, server-side environment. Web Forms applications and XML Web services, both of which are addressed later in this topic, are enabled by ASP.NET working directly with the runtime.

An example of an unmanaged application that hosts the runtime is Internet Explorer (in the form of a MIME type extension). You can embed controlled components or Windows Forms controls in HTML documents when the runtime is hosted by Internet Explorer. Controlled mobile code (similar to Microsoft® ActiveX® controls) is possible with this hosting method, but with substantial benefits that only managed code can provide, such as semi-trusted execution and safe isolated file storage.

Back End Technology:

About Microsoft SQL Server

Microsoft SQL Server is a Structured Query Language (SQL) based, client/server relational database. Each of these terms describes a fundamental part of the architecture of SQL Server.

Database

A database is similar to a data file in that it serves as a repository for information. A database, like a data file, does not explicitly present information to a user; however, the user runs an application that accesses data from the database and presents it to the user in a readable format.

The files that contain the physical database and the database management system (DBMS) software that applications use to access data are the two main components of a database. The database management system (DBMS) is in charge of implementing the database structure, which includes:

- Maintaining the relationships between data in the database.

- Ensuring that data is stored correctly and that the rules defining data relationships are not violated.

- Recovering all data to a point of known consistency in case of system failures.

Relational Database

There are many methods for organising data in a database, but relational databases are one of the most effective. The application of mathematical set theory to the problem of effectively organising data is known as relational database structures. Data is organised into tables in a relational database (called relations in relational theory).

When it comes to arranging data into tables, there are several different ways to describe tables. Normalization is a process defined by relational database theory that ensures that the collection of tables you define will effectively organise your data.

Client/Server:

The server in a client/server system is a relatively large device in a central location that manages a resource that is used by a large number of people. Individuals that require the resource link to the server through the network from their machines, or clients.

Servers have the following: The database files and database management applications reside on a computer in a client/server database architecture. Applications can run on different clients and communicate with the database server over a network thanks to a communications component. The SQL Server communication aspect also facilitates communication between a server-based application and SQL Server.

Middleware Technology

Active Data Objects.Net Overview

ADO.NET is an extension of the ADO data access model that explicitly addresses user needs for scalable application growth. It was created with scalability, statelessness, and XML in mind specifically for the web.

ADO.NET incorporates existing ADO artefacts, such as the Link and Command objects, as well as creating new ones. The Dataset, Data Reader, and Data Adapter are three of the most important new ADO.NET properties.

The main difference between this stage of ADO.NET and previous data architectures is that there is a different entity called a Dataset that is independent of any data stores. As a result, the Dataset can be used as a stand-alone object. You can think of the Dataset as an often disconnected record set that has no idea where the data it contains came from or where it is going. Inside a Dataset, much like in a database, there are tables, columns, relationships, constraints, views, and so forth.
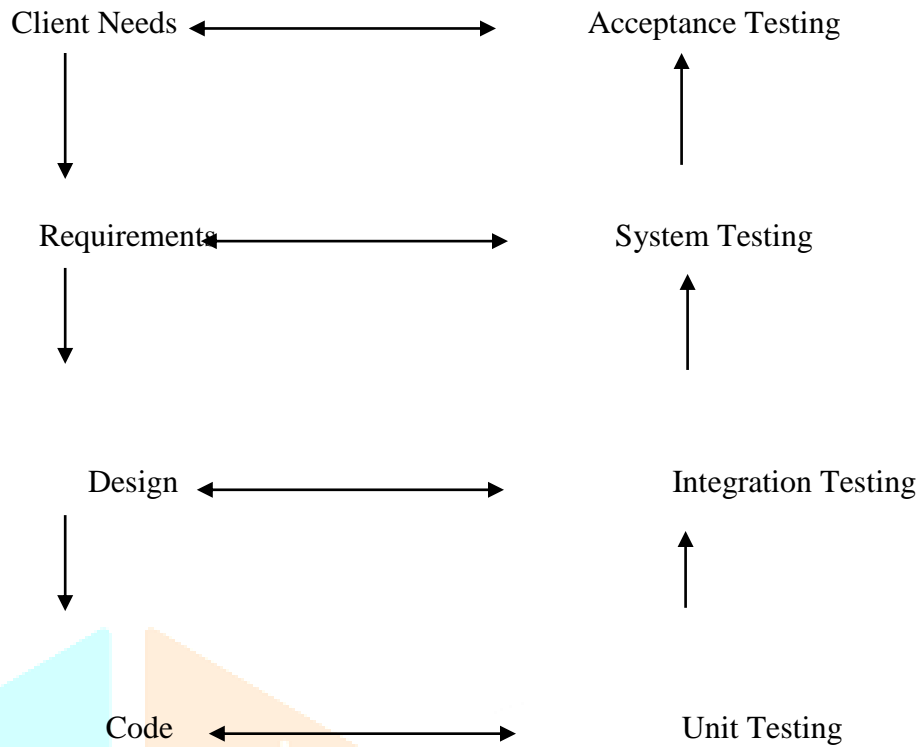
Testing

As the project is on bit large scale, we always need testing to make it successful. If each components work properly in all respect and gives desired output for all kind of inputs then project is said to be successful. So the conclusion is-to make the project successful, it needs to be tested.

The testing done here was System Testing checking whether the user requirements were satisfied. The code for the new system has been written completely using ASP .NET with C# as the coding language, C# as the interface for front-end designing. The new system has been tested well with the help of the users and all the applications have been verified from every nook and corner of the user.

Although some applications were found to be erroneous these applications have been corrected before being implemented. The flow of the forms has been found to be very much in accordance with the actual flow of data.

Levels of Testing

In order to uncover the errors present in different phases we have the concept of levels of testing. The basic levels of testing are:

Client Needs ⟷ Acceptance Testing

Requirements ⟷ System Testing

Design ⟷ Integration Testing

Code ⟷ Unit Testing

A series of testing is done for the proposed system before the system is ready for the user acceptance testing.

The steps involved in Testing are:

o **Unit Testing**

Unit testing focuses verification efforts on the smallest unit of the software design, the module. This is also known as "Module Testing". The modules are tested separately. This testing carried out during programming stage itself. In this testing each module is found to be working satisfactorily as regards to the expected output from the module.

o **Integration Testing**

Data can be grossed across an interface; one module can have adverse efforts on another. Integration testing is systematic testing for construction the program structure while at the same time conducting tests to uncover errors associated with in the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here correction is difficult because the isolation of cause is complicate by the vast expense of the entire program. Thus in the integration testing stop, all the errors uncovered are corrected for the text testing steps.

o **System testing**

System testing is the stage of implementation that is aimed at ensuring that the system works accurately and efficiently for live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, then goal will be successfully achieved.

o **Validation Testing**

At the conclusion of integration testing software is completely assembled as a package, interfacing errors have been uncovered and corrected and a final series of software tests begins, validation test begins. Validation test can be defined in many ways. But the simple definition is that validation succeeds when the software function in a manner that can reasonably expected by the customer. After validation test has been conducted one of two possible conditions exists.

One is the function or performance characteristics confirm to specifications and are accepted and the other is deviation from specification is uncovered and a deficiency list is created. Proposed system under consideration has been tested by using validation testing and found to be working satisfactorily.

o **Output Testing**

After performing validation testing, the next step is output testing of the proposed system since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated by the system under consideration. Here the output format is considered in two ways, one is on the screen and other is the printed format. The output format on the screen is found to be correct as the format was designed in the system designed phase according to the user needs.

For the hard copy also the output comes as the specified requirements by the users. Hence output testing does not result any corrections in the system.

o **User Acceptance Testing**

User acceptance of a system is the key factor of the success of any system. The system under study is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required.

Test Cases

User Login/Registration: To begin with login, user need to register by filling up basic registration details. There are multiple fields in registration page and every field has to fill by user. User cannot use character in the login id field.

Admin Login: - Admin login id and password is kept compulsory fields, and if the admin id or password doesn't match then it will show an error message.

Following below is the interface of the user login, if he/she is a new user then he/she can create a new account by clicking on the link below mentioned under the login button which is indicated as click here.
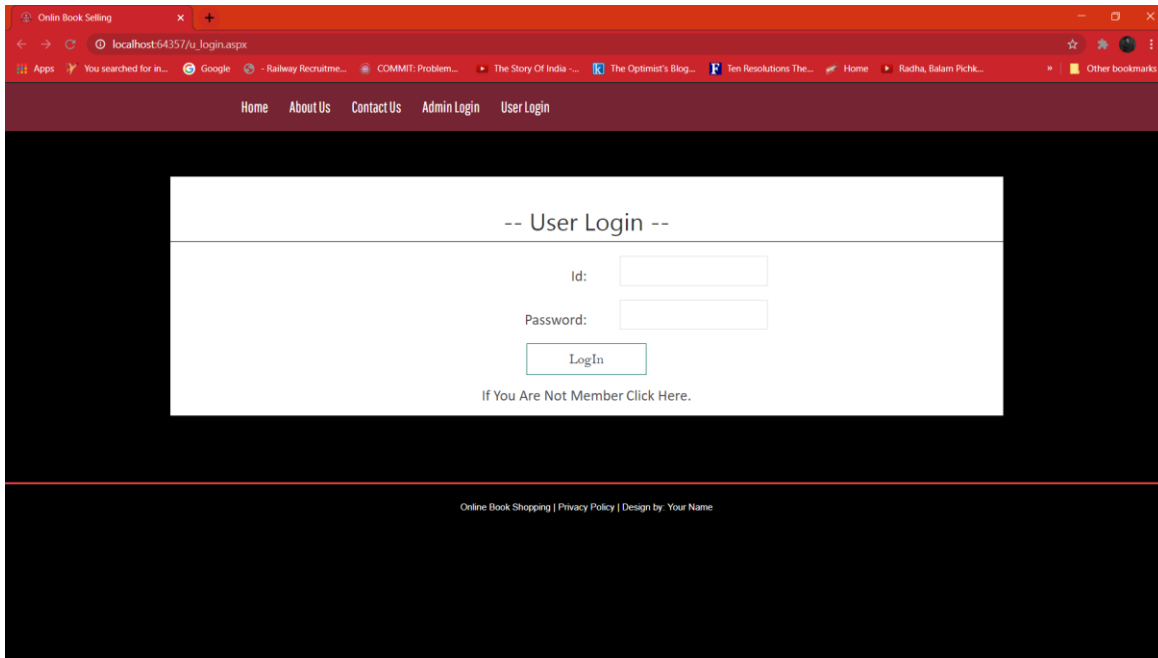
Fig.5.User Login Interface Snapshot

This the registration interface for the user here the user ID and the password are created automatically and after that the user is required to fill the given details and then create the account.
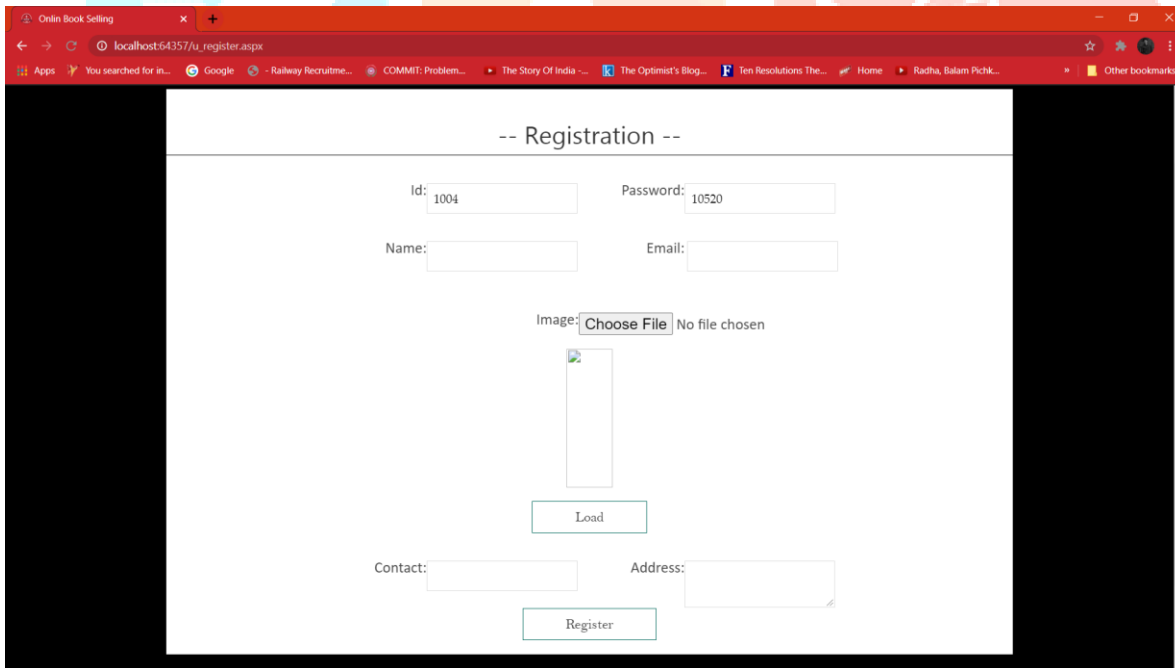

Fig.6. User Registration Interface Snapshot

This is the interface of the Adminstrator Login and below given images are the powers given to the admin i.e. he can control and overlook howmany users are there what are the books uploaded by each user which user bought which book the transactions and all.
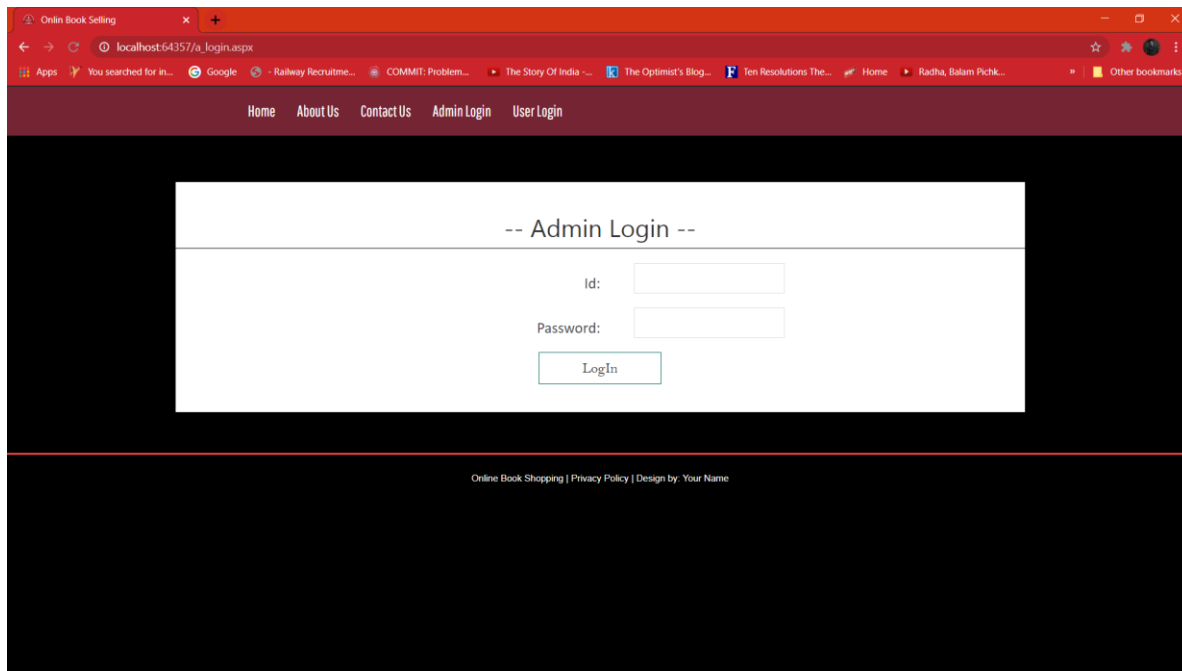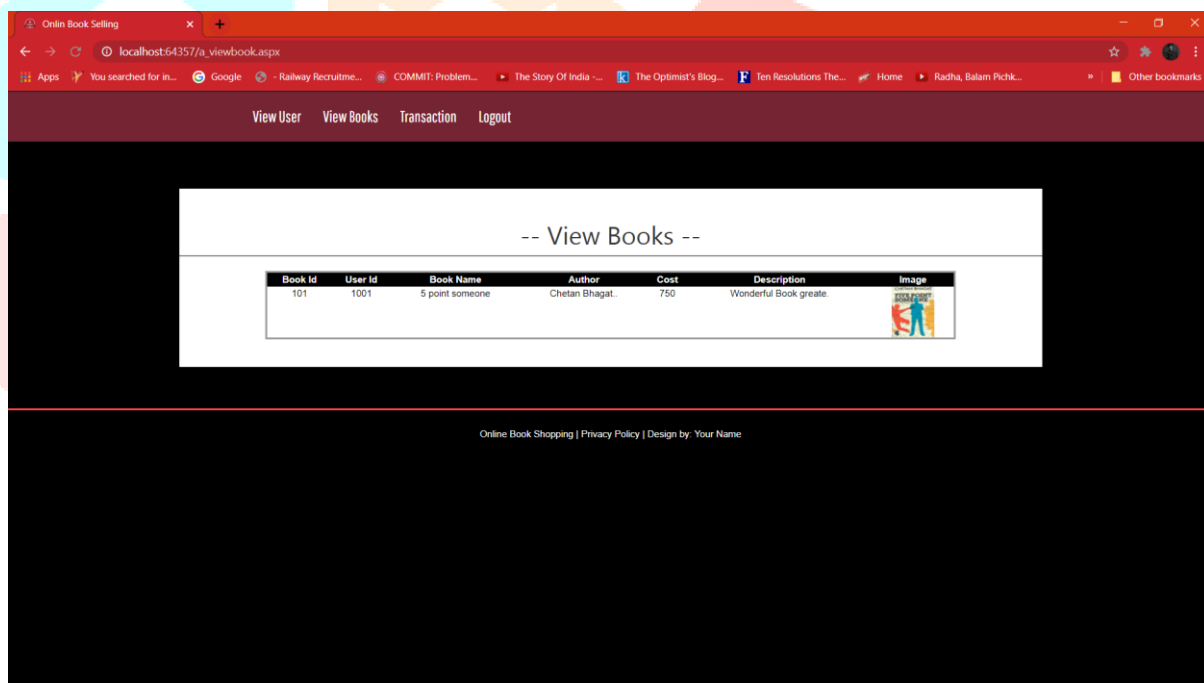
Fig.7.Admin Login Interface Snapshot

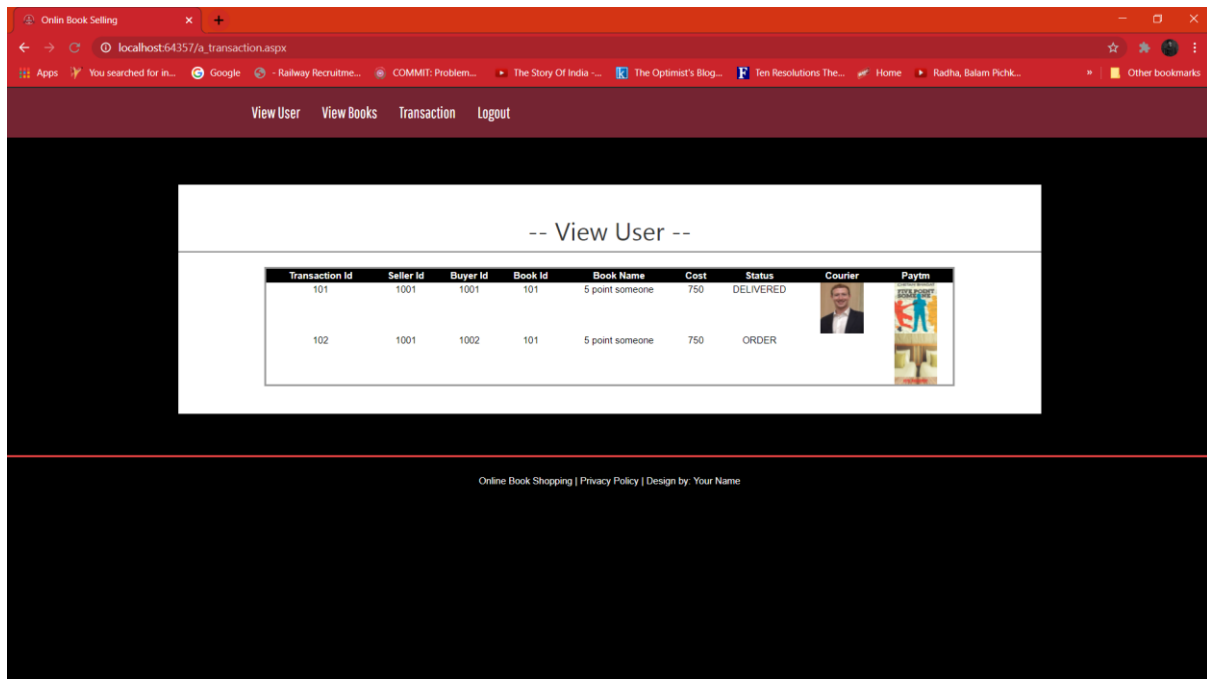*Fig.8. View Books Interface from Admin ID Snapshot*

Fig.9. View Users Interface from Admin ID Snapshot

Validation Criteria

1. In each form, no field which is not null able should be left blank.

2. All numeric fields should be checked for non-numeric values. Similarly, text fields like names should not contain any numeric characters.

3. All primary keys should be automatically generated to prevent the user from entering any existing key.

4. Use of error handling for each Save, Edit, delete and other important operations.

5. Whenever the user Tabs out or Enter from a text box, the data should be validated and if it is invalid, focus should again be sent to the text box with proper message.

Advantages of Project

- Customers can get their book delivered instead of actually going and buying the book.

- Feasibility of making payment online itself.

- Managing of inventory of the books for user becomes easier as customers are not visiting and ordering it online.

- This system saves both time and travelling cost of customers.

- User can get to know different kinds of books that they were unaware of by just searching in the system using keywords.

Conclusions

This was our project of System Design about "**Online Book Reselling Website**" is web application based on Asp .Net language. The Development of this system takes a lot of efforts from us. We think this system gave a lot of satisfaction to all of us. Though every task is never said to be perfect in this development field even more improvement may be possible in this application. We learned so many things and gained a lot of knowledge about development field. We hope this will prove fruitful to us.

References

- **Websites**
  - en.wikipedia.org
  - Microsoft Developer Network (MSDN): http://msdn2.microsoft.com/en-us/default.aspx: This is a valuable online resource, and is a must for any developer using Microsoft tools.
  - http://www.asp.net/: This is the official Microsoft ASP.NET web site. It has a lot of: tutorials, training videos, and sample projects.
  - https://www.ecwid.com/sell-books-online
  - https://www.researchgate.net/profile/Sanjay_Agal/publication/259478827_Online_Book_Store/links/0046352c11063a8ca8000000/Online-Book-Store.pdf
  - https://www.sites.google.com/site/bcafinalyearproject/project-report/online-book-store-project-report