Table of Contents

ABSTRACT

Today advancement happening in a world very fast and people using latest technologies to make their lives easy and flexible. By this point of view, we have introduced project that is the "**Notes - Marketplace**"**.**

The basic idea is to implement a **Notes - Marketplace** like platform through which different students of various professions from around the Country from different universities and colleges can come together, collaborate with one another and help each other. Students will be able to acquire free notes from other students, as well as can buy and or sell notes and old textbooks, and handwritten notes for any specific category (or profession).

LIST OF FIGURES

CHAPTER-1 INTRODUCTION

1.1 Project Definition

- The basic idea is to implement a **Notes - Marketplace** like platform through which different students of various professions from around the Country from different universities and colleges can come together, collaborate with one another and help each other. Students will be able to acquire free notes from other students, as well as can buy and or sell notes and old textbooks, and handwritten notes for any specific category (or profession).

1.2 Scope of the Work

Our **Web Application** can be used for various purpose like with the user of this web application one university's student can buy another university's students note and they can also buy there notes to another university's students. University had can also can sell their notes to the students of their university.

1.3 Technology Used

The following web-based Software is used to develop the proposed system.

Component	Reason
Development Tool **Visual Studio 2019**	• Powerful debugging tool. • It has a quick navigation, adaptive matching of commands, simultaneous editing .
Database **SQL Server 2017**	• Compatible with PHP. .
Programming Language **ASP.Net MVC 5.**	• New and Involving Language. • Compatible with Programmer.
Rich Media Photoshop	• Best tool for available for graphics.
Other **CSS**	• Best tool for Design

Table 1.1 Software Component

> ## .Net Technologies

- .NET technology helps in creating applications . The versatility of the technology has been the foremost factor that allows its widespread use. The .NET software is from Microsoft. Microsoft launched this technology stating it as an advanced new platform built on highly developed protocols with tools that help in faster and better communication and computing. software development company using this technology to make information and data available on any device at any time and something that can be accessed at any time. This technology is used to develop both desktop applications and web-based solutions.

- There are four main principles of .NET from the perspective of the user:
1. It erases the boundaries between applications and the Internet. Instead of interacting with an application or a single Web site, .NET will connect the user to an array of computers and services that will exchange and combine objects and data.

2. Software will be rented as a hosted service over the Internet instead of purchased on a store shelf. Essentially, the Internet will be housing all your applications and data.

3. Users will have access to their information on the Internet from any device, anytime, anywhere.

4. There will be new ways to interact with application data, such as speech and handwriting recognition.

- This is how Microsoft describes it: ".NET is the Microsoft Web services strategy to connect information, people, systems, and devices through software. Integrated across the Microsoft platform, .NET technology provides the ability to quickly build, deploy, manage, and use connected, security-enhanced solutions with Web services. .NET-connected solutions enable businesses to integrate their systems more rapidly and in a more agile manner and help them realize the promise of information anytime, anywhere, on any device."

- Microsoft views this technology as revolutionary, enabling Internet users to do things that were never before possible, such as integrate fax, email and phone services, centralize data storage and synchronize all of a users computing devices to be automatically updated.

- Microsoft .NET consists of four major components:
- 1,Common Language Specification (CLS)
- 2,Framework Class Library (FCL)
- 3,Common Language Runtime (CLR)
- 4,.NET Tools

➤ **SQL Server**

- Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet). Microsoft markets at least a dozen different editions of Microsoft SQL Server, aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users.

➤ **CSS & HTML - As designing tool:**

- Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation (that is, the look and formatting) of a document written in a markup language. Its most common application is to style web pages written in HTML and XHTML, but the language can be applied to any kind of XML document.

- CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the colors, fonts, and layout. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content. CSS can also allow the same markup page to be presented in different styles for different rendering methods,

➤ **HTML**

- HTML(Hyper Text Mark-up Language) is markup language for designing a web page. It provides a means to describe the structure of text-based information in a document by denoting certain text as links, headings, paragraphs, lists, etc. and to supplement that text with interactive forms, embedded images, and other objects. HTML is written in the form of tags that are surrounded by angle brackets.

➤ **JavaScript - As scripting language**

- JavaScript is a scripting language used to enable programmatic access to objects within other applications. It is primarily used in the form of client-side JavaScript for the development of dynamic websites.

- JavaScript was designed to add interactivity to HTML pages. We can say it is light weighted programming language. It is useful to react to event, to read ant write HTML elements, to validate input data

1.4 Hardware-Software Used

Application will be accessed through a Browser Interface. The interface would be viewed best using 1024 x 768 and 800 x 600 pixels resolution setting. The software would be fully compatible with Microsoft Internet Explorer for version 8 and above.

➢ **H/W requirement for Server**

- Processor Minimum: Pentium 4 CPU, 2.40 GHz
- Memory RAM: 6 GB
- Hard Disk: 250 GB

➢ **H/W requirement for Clients**

- Internet
- Browser interface supporting device

➢ **Software Requirements**

      o **FRONT - END**
       JavaScript, CSS, HTML

      o **BACK – END**
       SQL Server, ASP .NET MVC 5

      o **DESIGNING TOOL**
       Visual Studio

      o **DOCUMENTATION TOOL**
       Power Point
       Microsoft Word

      o **MICROSOFT OPERATING SYSTEM**
       Microsoft Windows

CHAPTER-2 SYSTEM ANALYSIS AND REQUIREMENT

2.1 System Analysis

2.1.1 Study of current System

As we know that now a day we are facing the problem to find the handwritten notes of the particular subject of the university. We are not able to find the some helpful guided book or any reference book of particular subject of our courses.

2.1.2 Problems and Weakness of Current System

- ✓ Time consuming...

- ✓ More expensive...

- ✓ Searching problem...

- ✓ Less accuracy...

- ✓ More stationary...

2.2 Requirements of System

Main Requirement Of new System is that, User should be able to search the particular Notes of any University and college he want he/she is able to download a particular book he/she went to buy on this portal.

2.2.1   User Requirements

User requirements include not many things, but the most important thing is user must be aware that system works properly with full availability, reliability, security and safety. The user responsibilities are as follow:

- Should know the data needed to address the problem.
- Should know how to use it
- Should adhere to guidelines and prescribed standards.

2.2.2   System Requirements

Health locater is mainly design to reduce the searching time of the particular medicine and to get the nearest place it is available at.
Actors: (User)
        Admin, Member, Super Admin
Functional Requirements are as follow.

2.3 Feasibility Study

A feasibility study is an evaluation of a proposal designed to determine the difficulty in carrying out a designated task. Generally, a feasibility study precedes technical development and project implementation. In other words, a feasibility study is an evaluation or analysis of the potential impact of a proposed project. As far as this project is concerned here the possibilities and the viability are concerning itself with the techniques, cost analysis, economic analysis, etc. used in these projects which are as follows:

Technical Feasibility: -

Technical feasibility is considered in terms of technical requirement and their availability in the market it determines whether the current level of technology supports the proposed system or not. The technical possibility of proposal system is as follow:

- At least one person is required which works as administrator.
- Manual describing Task file are available as soft copy.
- A miscellaneous tool like SQL Server Data Manger is also available. SQL Data Manager is available with Operating System.
- The unit does process the hardware as well as related Software for the project.
- The proposed system does not require much technical detail.
- The current manual working system is not so much sufficient.
- It just requires windows operating system.
- The organization has cared purchased all the enough devices for latest technology.

Economic Feasibility:

The Economic feasibility is considered in terms of money of or price value. The organization measures the cost effectiveness of the Project.

- Software purchase cost incurred (which is zero in our case).
- Data Gathering costs which includes information gathering costs.
- Man Power which includes developer and guides.
- The organizations are ready to invest in proposed system for latest technology and best result.

- As the personnel and the manager know the computer operating the unit need not have to appoint any computer operator.
- The unit has not to spend much amount for the computer hardware and software.

- The organization is in position and also incapable to pay amount for the system.
- As a result the project processes the economic feasibility.

Operational Feasibility: -

At the ultimate feasibility consideration of the proposed system will fulfill the department's requirements. Operational feasibility as under:

- The proposed system will fulfill the organization's requirements.
- The proposed system covers all the aspects of current manual system.
- The responsible user of the system that is managers, accountants and partners are honesty enthusiastic for the now system.
- The changes of the system being operational are quite strong.
- It will be easier and efficient to take various decisions and actions for the institute as it is based on the official Documents.
- People with a basic knowledge of computers would be able to use our system very effectively and easily, as the system would have and intuitive GUI. The officers have a working knowledge of computers so understanding the working of the system and using it would be easy from the decision maker's point of view.

Consequently, we can say this system is also processes the operational feasibility hence with study of the above three major area of feasibility. It is quite natural that the organizations are ready to go ahead and work with proposed system.

Schedule Feasibility:-
- It is the most important part of the feasibility study because it deals with time-limits of the system to be complete.
- The system schedule is first completing system design and after that all implementation will be performed. If system complete in 3 months so I can add more features.

2.4 Feature of System

- Health Locater has below features.

- ✓ Time saving...

- ✓ Less expensive...

- ✓ Powerful searching...

- ✓ More accuracy...

2.5 Main Module of System

Non-registered users (Anonymous):

Without doing any registrations, Different students of various professions from around the Country should able to search for any country, profession or course specific notes (whatever notes are published from the registered seller members over portal.)

There can be two type of notes User can download from the portal: Free & Paid. For Downloading notes user need to register themselves at portal.

We can also show to users about note ratings/reviews added by various downloaders, preview of their notes so that one can make the decision for download/purchase the notes.

Registered users (Members):

When user want to download/buy any specific notes or want to sell their notes – they need to register themselves at portal.

Registered users (Members) can raise a request to system administrator for publish their notebooks or handwritten notes over the portal so other members can download/buy. After downloading notes, register member can see their past downloaded notes and provide ratings or review feedback about notes. User also can mark any notes as inappropriate.

Registered users (Members) can see download/sell history.

Administration users (Administrators):

We will have the same portal for admin users to login.

- Admin can have access to all members registered and one's published notes over portal.

- Admin also can see which member has downloaded which note(s).

- Admin has the full control over approval process for notes is being published at portal.

- Admin can reject member request with appropriate remarks.

- At any moment admin can delete inappropriate reviews of an admin.

- Admin also should have ability for see Statistical data.

- At any moment admin can unpublished the notes which will lead to removal of notes to display from the non-registered or registered user portal.

2.6 System Requirement Study and Assumption

Requirement study

Interviewing Technique: The primary purpose of interviewing is to obtain both quantitative and qualitative data regarding user requirements, policies, procedures and practices. To have a successful interview a set of etiquettes must be followed.

Questionnaires: Detail questionnaires are useful to gather quantitative information such as scaling of projections, the methodology followed, the use of projections, types of projections and their details etc. They are no substitutes for interviews as questionnaires do not get qualitative information. They are useful for quantitative data. The shorter a questionnaire, the higher is the probability of getting replies back quickly.

Fact Finding: The Results of fact-finding are summarized as a report. The team agreed on following a standard format in which fact-finding results were summarized. The report contained the following data:

- Background Information
- Sources used in obtaining information
- Method used in gathering information
- Current procedures
- Currently available data

Assumptions:

In general it has been assumed that the user has complete knowledge of the system that means user is not a naïve user. Any data entered by him/her will be valid. To make the web software as user friendly as possible but at the same time keeping in minds user requirements.

- User has the knowledge of internet and basic knowledge of application`s working system

- User has the device which supports the web as well as android/IOS apps.

Software development consisted of a programmer writing code to solve a problem or automate a procedure. Nowadays, it is must to try to see any conventional system as replaceable by the new system to provide user friendly and time saving experience.

CHAPTER-3 PROJECT MANAGEMENT

3.1 Project Planning3.1.1 Project Development Approach

3.1.1 Project Development Approach

A software development model is a simplified description /abstract representation of a software process which is presented from a particular perspective. A development model for software engineering is chosen based on the nature of the project and application, the methods and tools to be used, and the controls and deliverables that are required.

All software development can be characterized as a problem solving loop in which four distinct stages are encountered:

- o Status quo
- o Problem definition
- o Technical development
- o Solution integration

*1) Status Quo*represents the current state of affairs.

2) Problem Definition identifies the specific problem to be solved.

*3) Technical Development*solves the problem through the application of some technology.

*4) Solution Integration*delivers the results (e.g. documents, programs, data, and new product) to those who requested the solution in the first place.

There are many software development models. Model followed in this project is Incremental.

Incremental model is an evolution of waterfall model. The product is designed, implemented, integrated and tested as a series of incremental builds. It is a popular model software evolution used many commercial software companies and system vendor. Incremental software development model may be applicable to projects where:

- Software Requirements are well defined, but realization may be delayed.
- The basic software functionality are required early

Analysis	Design	Code	Test

Analysis	Design	Code	Test

Analysis	Design	Code	Test

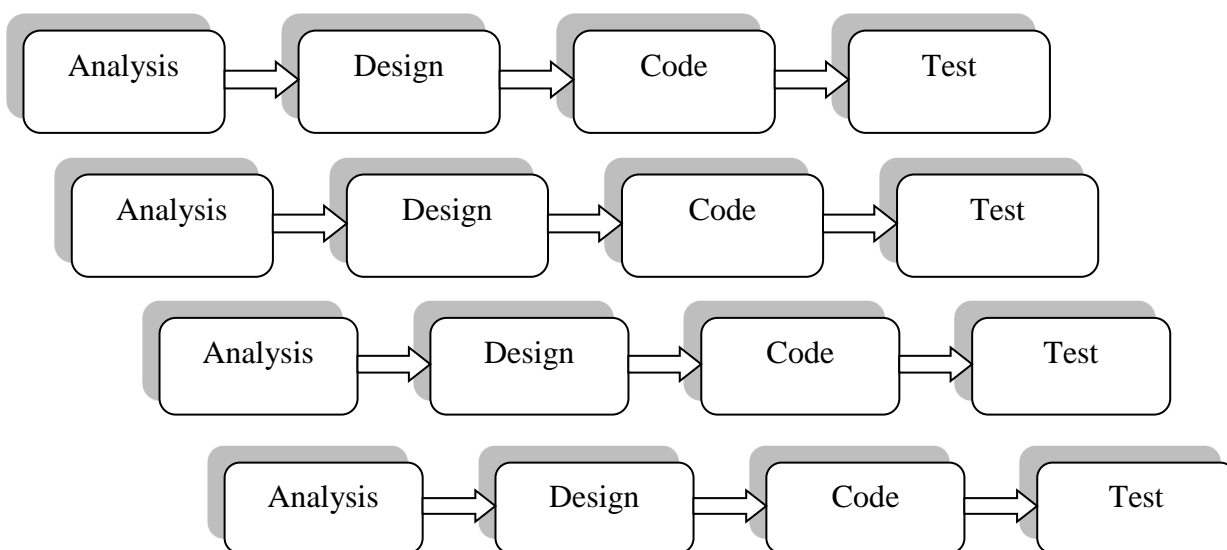| Analysis	Design	Code	Test

Figure: 3.1 Incremental Model

The **incremental build model** is a method of software development where the model is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This model combines the elements of the waterfall model with the iterative philosophy of prototyping.

The product is decomposed into a number of components, each of which are designed and built separately (termed as builds). Each component is delivered to the client when it is complete. This allows partial utilization of product and avoids a long development time. It also creates a large initial capital outlay with the subsequent long wait avoided. This model of development also helps ease the traumatic effect of introducing completely new system all at once.

The incremental model contain following phase:

1.   System Analysis:

It is first stage of process model. This provides primary analysis of the system

Requirement and data gathering. It's proces of gethring data to understand the neature of software.the enggniear must understanding the information domain for software as well as required function behaviear performance and interface.

2.   System Design:

As per analysis of system will be designed. It includes designing of template of website. Software design is multi step process that focus is on fourattribute of the programmer attribute, Data structure, Software arechistructure, Interface represention.Process design translate requirment into graphical represent four coding of software.

3.   System Coding:

Design of system is implemented in this phase. The system design translate into computerunderstable language into this step. This step is also knows asprograming step.

4.   System Testing:

After coding each module is tested using different methods. After the coding of the system all staement have been tested in this step. The goal of testing to un curve the requirment design and coding error.

5.   Maintenance:

After completing system administrator maintain it as per user requirement. Software will change or update after it is supplied to custmer at that time when error is detected for chang will occure are solved in this step.

3.1.2 Project Plan

Project Plan considered as below:

Development & Maintenance

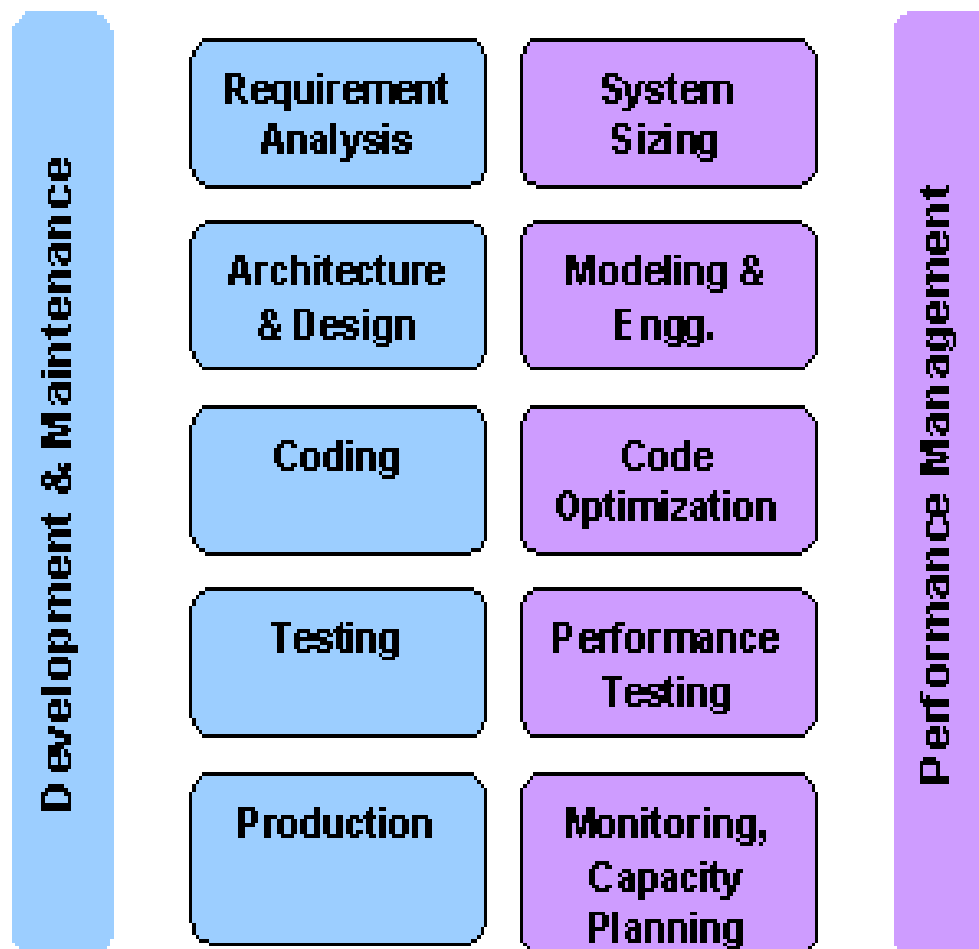|	
Requirement Analysis	System Sizing
Architecture & Design	Modeling & Engg.
Coding	Code Optimization
Testing	Performance Testing
Production	Monitoring, Capacity Planning

Performance Management

Figure: 3.2 Project plan

Once A Project Is Found To Be Feasible, Software Project Managers Undertake Project Planning. Project Planning Is Undertaken And Completed Even Before Any Development Activity Starts. Effective Management Of A Software Project Depends On Thoroughly Planning The Progress Of The Project.

The Objective Of Project Planning Is To Provide A Framework That Enables The Manager To Make Reasonable Estimates Of Resources, Cost And Schedule. These Estimates Are Made Within A Limited Time Frame At The Beginning Of The Software Project And Should Not Be Updated Regularly As The Project Progresses.

Project Planning Process Starts With An Assessment Of The Constraints Affecting The Project.

- ✓ Identify Objectives
- ✓ Determine Information Requirements
- ✓ Analyze System Needs
- ✓ Designing
- ✓ Documentation
- ✓ Development

3.2 Milestone and Deliverables

Milestones	Deliverables
Project Specification	Definition, Objectives, Goals and Scope etc related discussion.
Analysis of Project	Gatherings information, Requirements of user.
Design of Project	DFD, Database Design, Form design etc.
Coding	Decide coding strategies, standards etc.
Testing andDocumentation	Testing methods, Strategy, cases like required input and output.

Table: 3.1 Milestones and Deliverables

3.3 Roles and Responsibilities

Role	Responsibility
Project Manager	Defining scope
	Providing required resources
	Project planning, tracking and monitoring
	Analysis
	Co-ordination between project teams
Project Leader	Designing and Documentation
	Execution of project as per defined schedule
	Reporting to PM
	Testing
Team Member	Software development as per the design and documentation

Table:3.2 Roles and Responsibilities

3.4 Project Schedule

```
┌─────────────────┐     ┌─────────────────┐     ┌─────────────────┐     ┌─────────────────┐
│ Project Profile │ ──> │     Project     │ ──> │  Logical Design │ ──> │    Database     │
│                 │     │    Analysis     │     │                 │     │     Design      │
└─────────────────┘     └─────────────────┘     └─────────────────┘     └─────────────────┘
                                                                                 │
                                                                                 ▼
┌─────────────────┐     ┌─────────────────┐     ┌─────────────────┐     ┌─────────────────┐
│   Evaluation    │ <── │  Documentation  │ <── │     Testing     │ <── │      Code       │
│                 │     │                 │     │                 │     │  Implementatio  │
└─────────────────┘     └─────────────────┘     └─────────────────┘     └─────────────────┘
```
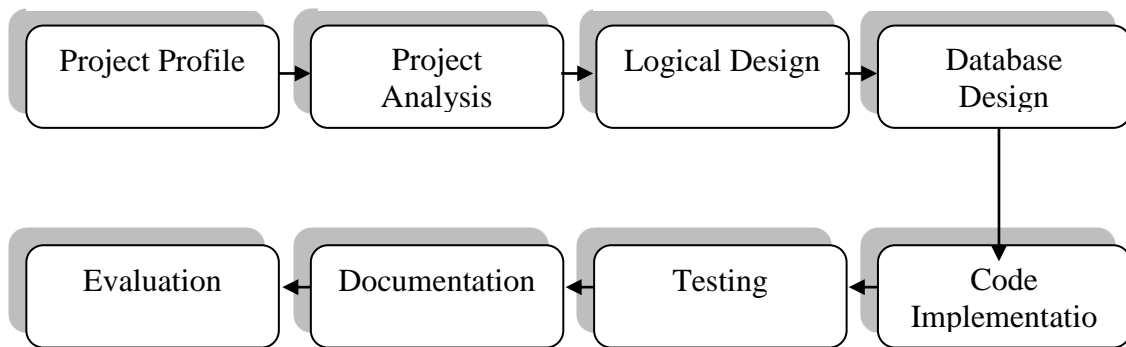
Figure: 3.3 Project Schedule

Month=>	Jul - Sep	Oct - Dec	Jan - March	April
Days=>	1....15.....30/31	1....15.....30/31	1....15....30/31	1.....15.......15
Phase 1				
Phase 2				
Phase 3				
Phase 4				
Phase 5				
Phase6				
Phase7				

🟥	**Comm. problem identification**		🟪	**Project start**
🟨	**Preliminary investigation**		🟫	**Submit IDP**
🟩	**System analysis**		⬛	**Completion of Project Analysis & design**
🟦	**System Design**		🟥	**Project Complete**
🟧	**Coding**		🟦	**Continuous project phase**
⬛	**Testing**			
🟦	**Implem & maintance**			
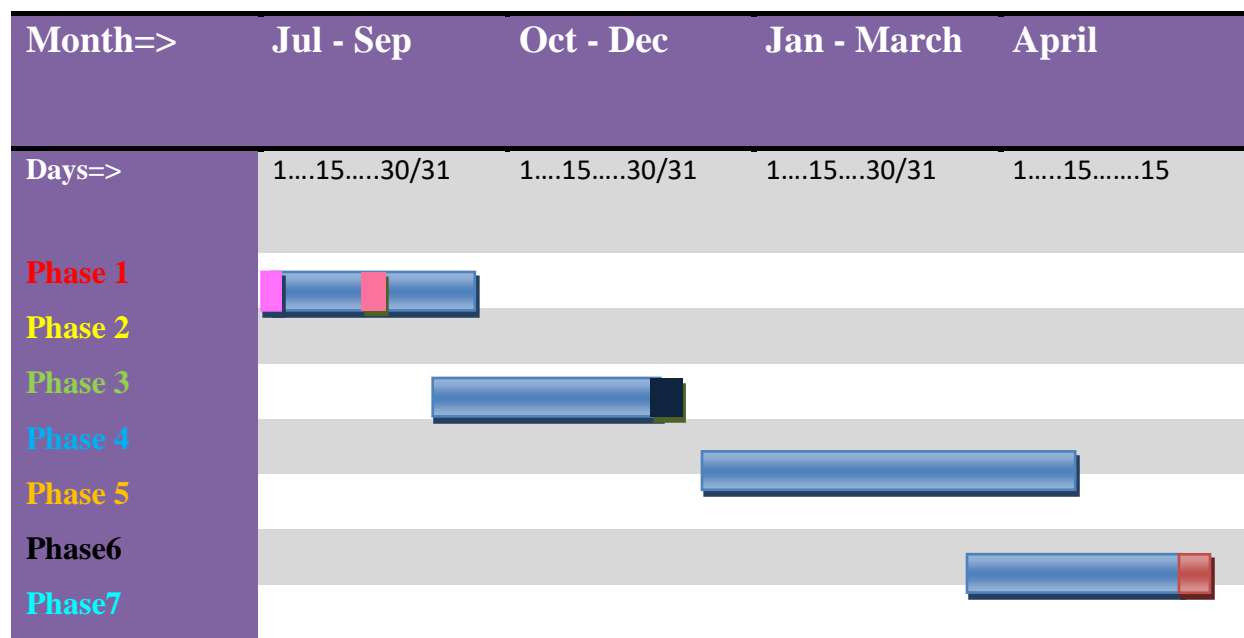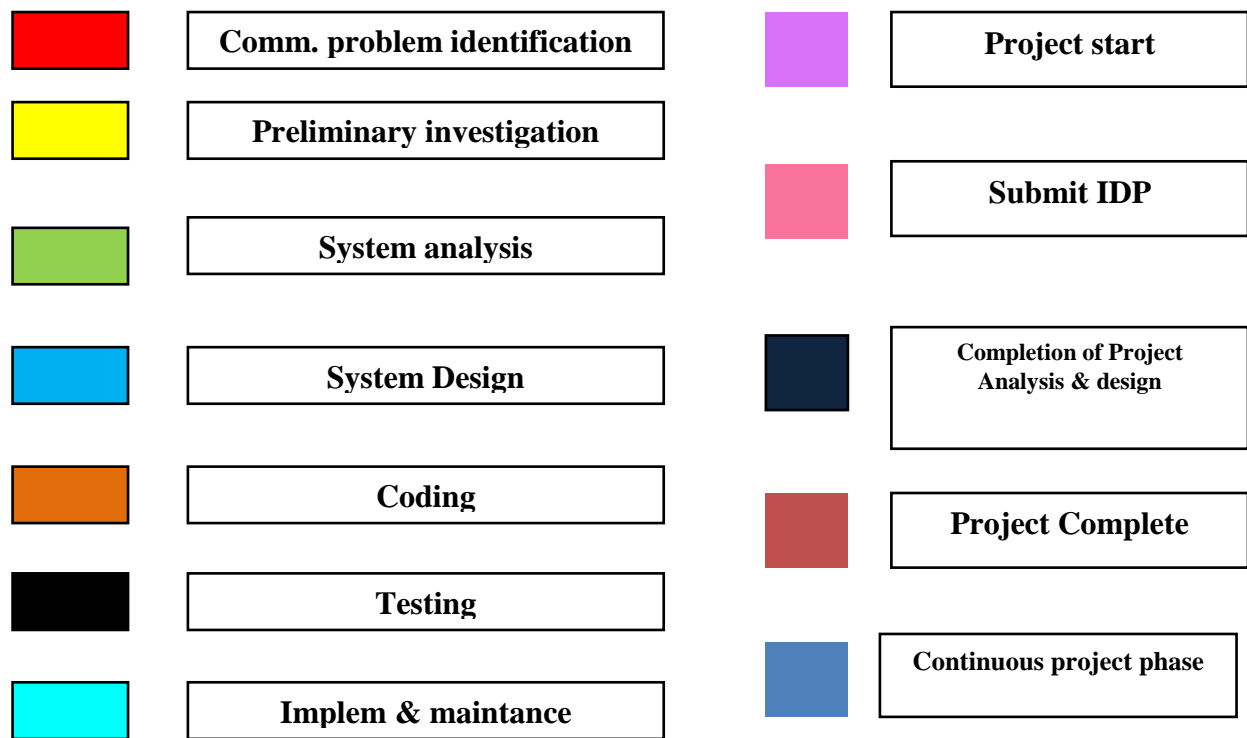
Figure: 3.4 Schedule Representations

The project schedule provides a road map for a software project manager. If it has been properly developed, the project schedule defines the task and milestones that must be tracked and controlled as the project proceeds. Tracking can be accomplished in a number of different ways.

- Conducting periodic project status meeting in which each team member reports progress and problems.

- Evaluating the results of all reviews conducted throughout the software engineering process.

- Determining whether formal project milestone have been accomplished by the schedule date.

- Comparing actual start date for each project task listed in the resource table.

- Meeting informally with practitioners to obtain their subjective assessment of progress to date and problems on the horizon.

3.5 Risk Management

3.5.1 Risk Identification:

During the analysis of our whole application, we discovered some of the risks that could affect this website while developing and also while using. These are those risks:

- Technological Risk
- Economical Risk
- Political Risk

3.5.2 Risk Analysis:

Here is the description of how the above mentioned risks were to affect this application:

- **Technological Risk:** First of all, this application requires web browser that supports JavaScript and flash. The most common problems are low processor configuration, low buffer memory, low speed etc…

- **Economical Risk:** This project is intranet application .there is no extra economic changes are there. This project is totally cheapest.

- **Political Risk:** There are many political risks such as wrong information about any place in city, images that are not permitted to show, also permit of all private organization which information includes in our intranet application.

During the risk analysis process, each identified risk is considered in turn and a judgment made about the probability and the seriousness of the risk. It relies on the judgment and experience of the project manager.

Risk	Probability	Effects
Financial Problem	Low	Catastrophic
S/W component which should be reused contain defects	Moderate	Serious
Changes to requirements which require major design	Moderate	Serious
Required training for staff is not available	Moderate	Serious
The rate of defect repair is underestimated	Moderate	Tolerable
The time required to develop the s/w is underestimated.	High	Serious

Table: 3.3 Risk& Probability

3.5.3 Risk Planning:

Here is how we deal with all the above said risks:

- **Technological Risk:**To avoid this risk, I planned that use JavaScript whenever it must be required and avoiding use of flash.

- **Economical Risk:**There is no need to solve economical risk. Because there is no problem about economical problem.

- **Political Risk:**Permitting of the organization to display or advertisement information is solves this problem.

The risk which might be uncounted after setting up the server is shown in the table below. All the applications have different internal and external risks. Internal risks basically comprise with hardware failure, server down, power interruption for which the solution is specified. External risks are associated with the application like virus, hacking and the corruption of files. The solution is mentioned in the table below, which is again not much difficult to handle if proper risk planning is done.

Sr. no	Risk	Type	Probability	Impact of Risk	Solution
1	H/W failure	Internal	Low	Serious	Back up
2	Server down	Internal	High	Serious	Mirror site
3	Power Interruption	Internal	Low	Low	UPS
4	Virus	External	High	Serious	Antivirus
5	Hacking	External	Low	Moderate	Firewall
6	File corruption	External	Moderate	Serious	Back up

Table: 3.4Risk & Solution

3.5.4 Risk Transfer

Means causing another party to accept the risk, typically by contract or by hedging. Insurance is one type of risk transfer that uses contracts. Other times it may involve contract language that transfers a risk to another party without the payment of an insurance premium. Liability among construction or other contractors is very often transferred this way. On the other hand, taking offsetting positions in derivatives is typically how firms use hedging to financially manage risk.

Some ways of managing risk fall into multiple categories. Risk retention pools are technically retaining the risk for the group, but spreading it over the whole group involves transfer among individual members of the group. This is different from traditional insurance, in that no premium is exchanged between members of the group up front, but instead losses are assessed to all members of the group.

The planning by which the risks cannot occur is:

➢ Interact with all branches.
➢ Authorize carefully and creating new safe security system for that.
➢ Make new security for natural threats.

CHAPTER-4 SYSTEM DESIGN

4.1 Context Level Diagram:
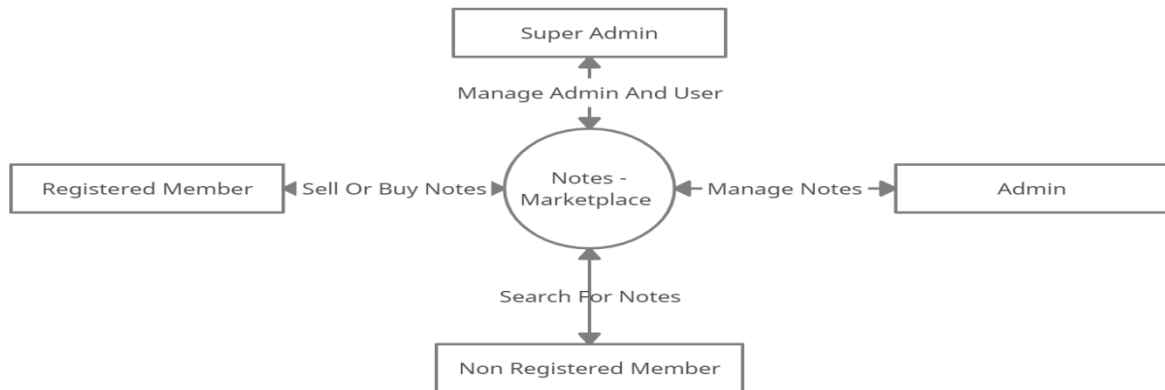
Figure: 4.1 Context Diagram

4.2 UML Diagram

4.2.1 Activity Diagram
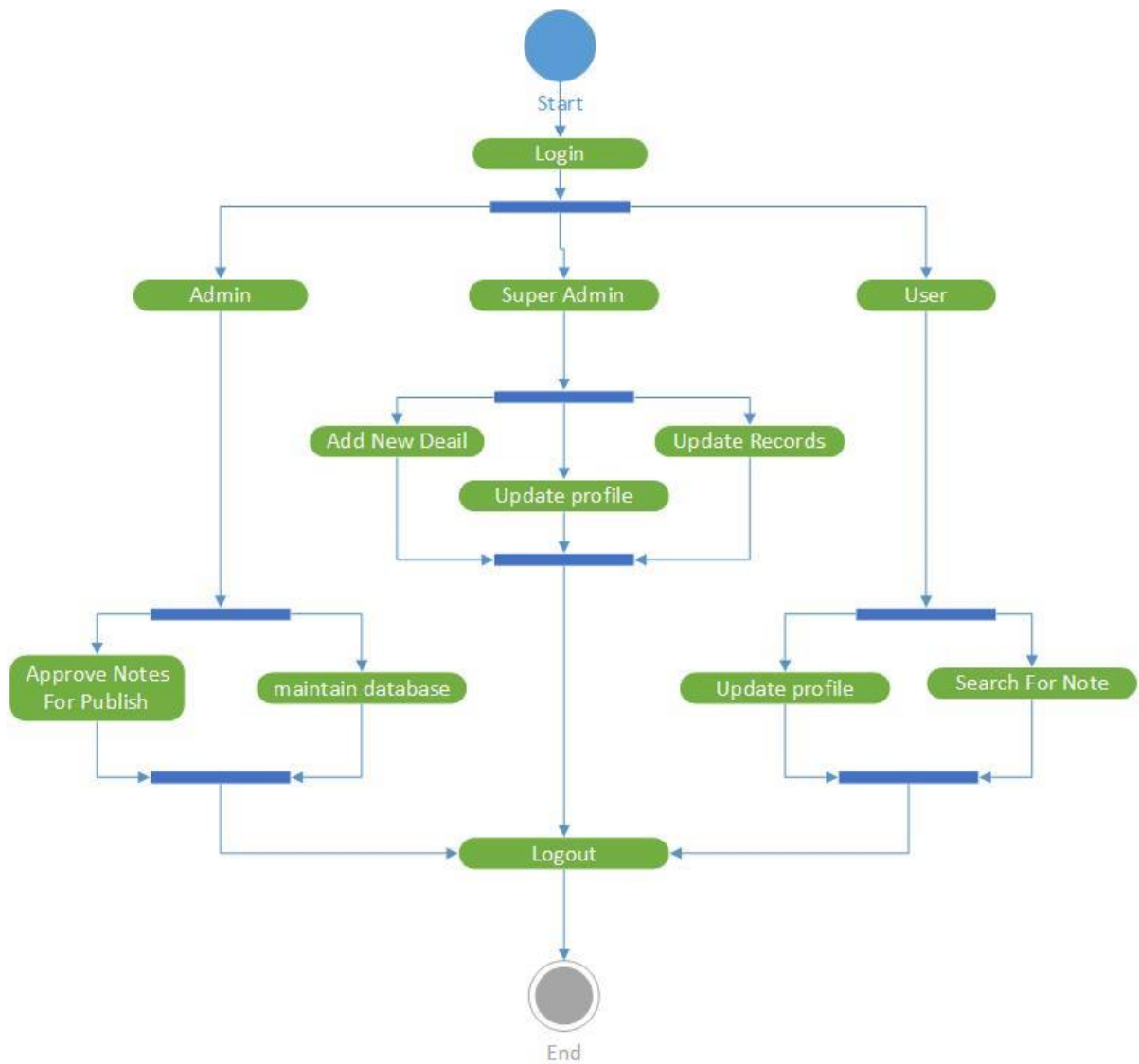
Figure: 4.2 Activity Diagram

4.2.2 Sequence Diagram
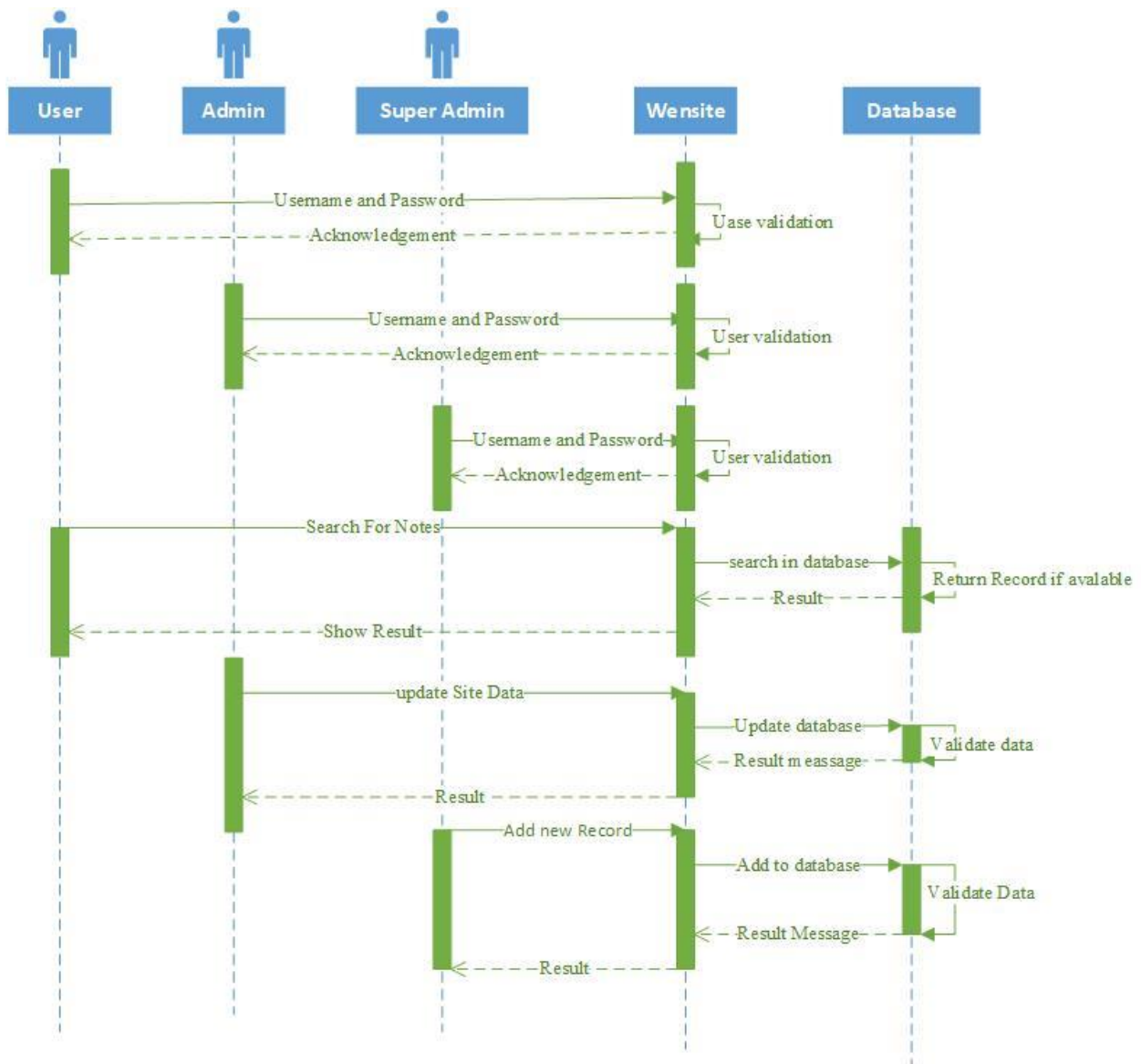
Figure: 4.3 Sequence Diagram

4.3 Data Dictionary

User Table

Field Name	Data Type	Length	Nullable
ID	int		NO
RoleID	int		NO
FirstName*	varchar	50	NO
LastName*	varchar	50	NO
EmailID*	varchar	100	NO
Password*	varchar	24	NO
IsEmailVerified	bit		NO
CreatedDate	datetime		YES
CreatedBy	int		YES
ModifiedDate	datetime		YES
ModifiedBy	int		YES
IsActive	bit		NO

User Profile Table

Field Name	Data Type	Length	Nullable
ID	int		NO
User ID	int		NO
DOB	datetime		YES
Gender	int		YES
SecondaryEmailAddress	varchar	100	
Phone number – Country Code *	varchar	5	NO
Phone number *	varchar	20	NO
Profile Picture	varchar	500	YES
Address Line 1*	varchar	100	NO
Address Line 2*	varchar	100	NO
City*	varchar	50	NO
State*	varchar	50	NO
Zip Code *	varchar	50	NO
Country*	varchar	50	NO
University	varchar	100	YES
College	varchar	100	YES
CreatedDate	datetime		YES
CreatedBy	INT		YES
ModifiedDate	datetime		YES
ModifiedBy	INT		YES

Seller Note Table

Field Name	Data Type	Length	Nullable
ID	INT		NO
SellerID	INT		NO
Status	INT		NO
ActionedBy	INT		YES
AdminRemarks	varchar	max	YES
PublishedDate	datetime		YES
Title*	varchar	100	NO
Category*	INT		NO
DisplayPicture	varchar	500	YES
NoteType	INT		YES
NumberofPages	INT		YES
Description*	varchar	max	NO
UniversityName	varchar	200	YES
Country	INT		YES
Course	varchar	100	YES
CourseCode	varchar	100	YES
Professor	varchar	100	YES
IsPaid	bit		NO
SellingPrice	decimal		YES
NotesPreview	nvarchar	max	YES
CreatedDate	datetime		YES
CreatedBy	INT		YES
ModifiedDate	datetime		YES
ModifiedBy	INT		YES
IsActive	bit		NO

Download Table

Field Name	Data Type	Length	Nullable
ID	INT		NO
NoteID	INT		NO
Seller	INT		NO
Downloader	INT		NO
IsSellerHasAllowedDownload	bit		NO
AttachmentPath	varchar	max	YES
IsAttachmentDownloaded	bit		NO
AttachmentDownloadedDate	datetime		YES
IsPaid	bit	100	NO
PurchasedPrice	decimal		YES
NoteTitle	varchar	100	NO
NoteCategory	varchar	100	NO
CreatedDate	datetime		YES
CreatedBy	INT		YES
ModifiedDate	datetime		YES
ModifiedBy	INT		YES

4.4 Use Case Diagram
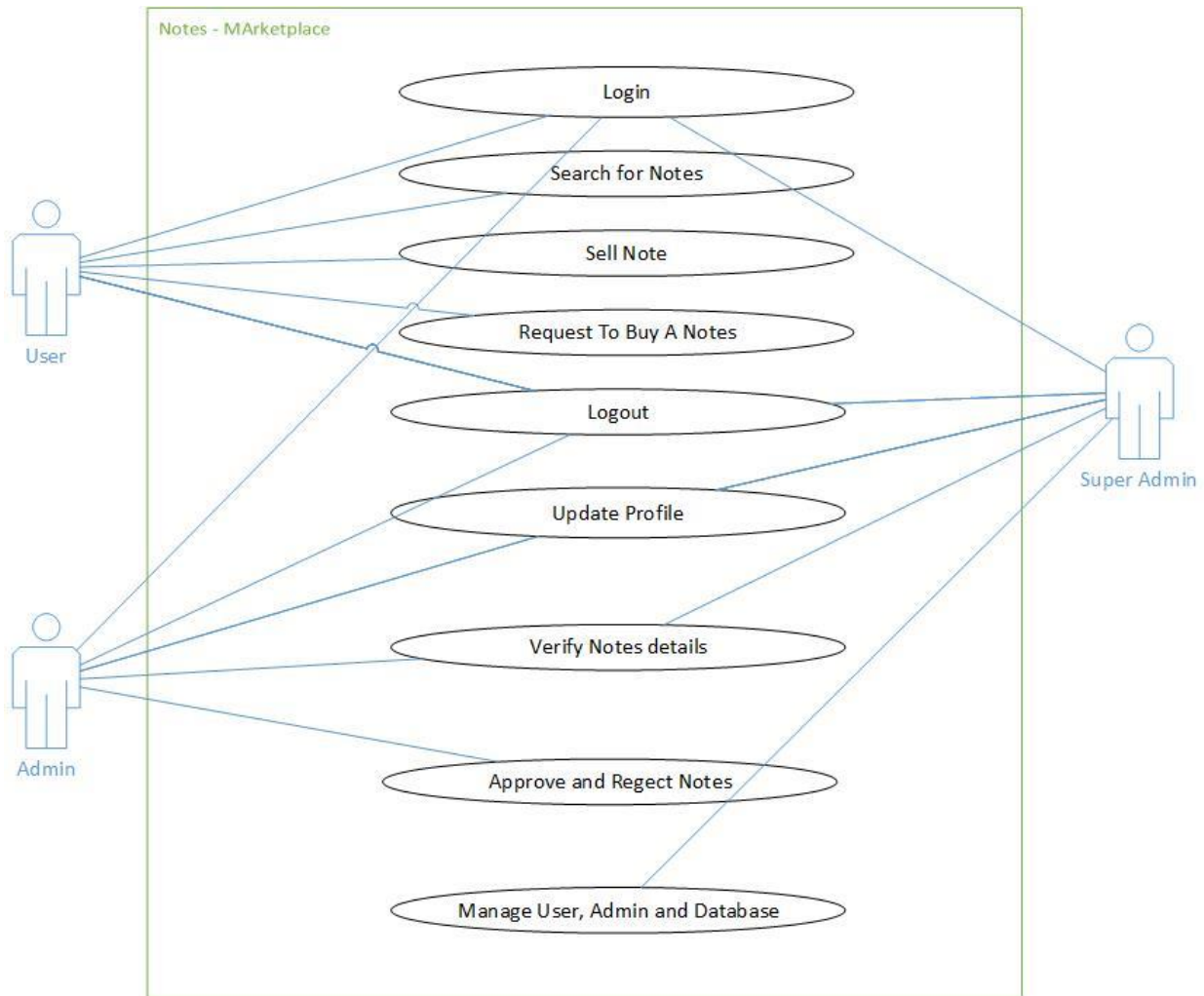
4.4 Use Case Diagram

CHAPTER-5 IMPLEMENTATION

5.1 Implementation Environment

The online Notes – Marketplace system is the web-based systems so the following web-based Software is used to develop the proposed system.

.Net Technology

- .NET technology helps in creating applications . The versatility of the technology has been the foremost factor that allows its widespread use. The .NET software is from Microsoft. Microsoft launched this technology stating it as an advanced new platform built on highly developed protocols with tools that help in faster and better communication and computing. software development company using this technology to make information and data available on any device at any time and something that can be accessed at any time. This technology is used to develop both desktop applications and web-based solutions.

- There are four main principles of .NET from the perspective of the user:
5. It erases the boundaries between applications and the Internet. Instead of interacting with an application or a single Web site, .NET will connect the user to an array of computers and services that will exchange and combine objects and data.

6. Software will be rented as a hosted service over the Internet instead of purchased on a store shelf. Essentially, the Internet will be housing all your applications and data.

7. Users will have access to their information on the Internet from any device, anytime, anywhere.

8. There will be new ways to interact with application data, such as speech and handwriting recognition.

- This is how Microsoft describes it: ".NET is the Microsoft Web services strategy to connect information, people, systems, and devices through software. Integrated across the Microsoft platform, .NET technology provides the ability to quickly build, deploy, manage, and use connected, security-enhanced solutions with Web services. .NET-connected solutions enable businesses to integrate their systems more rapidly and in a more agile manner and help them realize the promise of information anytime, anywhere, on any device."

- Microsoft views this technology as revolutionary, enabling Internet users to do things that were never before possible, such as integrate fax, email and phone services, centralize data storage and synchronize all of a users computing devices to be automatically updated.

- Microsoft .NET consists of four major components:
- 1,Common Language Specification (CLS)
- 2,Framework Class Library (FCL)
- 3,Common Language Runtime (CLR)
- 4,.NET Tools

- **<u>Development Tool</u>**

  Visual Studio 2019

- **<u>Database</u>**

  SQL Server 2017

5.2 Code Modules

5.2.1 Account Modules

5.2.1.1 Login

```
public class AccountController : Controller
{

    [Route("Login")]
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult Login(UserLogin user)
    {
      if (ModelState.IsValid)
      {
        //Encrypt Password and Save
        var newPassword = EncryptPassword.EncryptPasswordMd5(user.Password);

        bool isValid = objNoteMarketplaceEntities.Users.Any(x => x.EmailID ==
        user.EmailID && x.Password == newPassword && x.IsActive == true);
        if (isValid)
        {
            User userDetails = objNoteMarketplaceEntities.Users.Where(x => x.EmailID
           == user.EmailID && x.Password == newPassword).FirstOrDefault();
            if (userDetails.IsEmailVerified)
          {
                FormsAuthentication.SetAuthCookie(user.EmailID,
                user.RememberMe);

                if (userDetails.RoleID objNoteMarketplaceEntities.UserRoles.Where(x
                => x.Name.ToLower() == "admin" && x.IsActive == true).Select(x
                => x.ID).FirstOrDefault())
          {
                UserProfile userprofile =
                objNoteMarketplaceEntities.UserProfiles.Where(x => x.UserID ==
                userDetails.ID).FirstOrDefault();
            if (userprofile != null)
            {
              return RedirectToAction("Index", "Admin");
            }
            return RedirectToAction("MyProfile", "Admin");
          }
          else if (userDetails.RoleID ==
          objNoteMarketplaceEntities.UserRoles.Where(x => x.Name.ToLower() ==
          "superadmin" && x.IsActive == true).Select(x => x.ID).FirstOrDefault())
          {
                UserProfile userprofile =
                objNoteMarketplaceEntities.UserProfiles.Where(x => x.UserID ==
                userDetails.ID).FirstOrDefault();
            if (userprofile != null)
```

```
                {
                    return RedirectToAction("Index", "Admin");
                }
                return RedirectToAction("MyProfile", "Admin");
            }
            else
            {
            var FirstTime = objNoteMarketplaceEntities.UserProfiles.Any(x => x.UserID
            == userDetails.ID);
                if (FirstTime)
                {
                    return RedirectToAction("SearchNotes", "Home");
                }
                return RedirectToAction("UserProfile", "User");
            }
        }
        TempData["Error"] = "Email Address Is Not Verified";
        return View();
    }
    TempData["Error"] = "Invalid username or password";
    return View();
  }
  return View();
}
```

5.2.1.2 Register

```
[Route("Register")]
[HttpPost]
public ActionResult Register(UserRegistration objUser)
{
  if (ModelState.IsValid)
  {
        UserRole role = objNoteMarketplaceEntities.UserRoles.Where(x =>
        x.Name.ToLower() == "member").FirstOrDefault();
    User obj = new User
    {
      FirstName = objUser.FirstName,
      LastName = objUser.LastName,
      EmailID = objUser.EmailID,
      RoleID = role.ID,
      IsActive = true,
      CreatedDate = DateTime.Now,
      IsEmailVerified = false
    };

    string activationCode = Guid.NewGuid().ToString();

    //Encrypt Password and Save
    obj.VerificationCode = activationCode;
```

```csharp
obj.Password = EncryptPassword.EncryptPasswordMd5(objUser.Password);
var userName = objUser.FirstName.ToString();

// Adding User To DataBase
objNoteMarketplaceEntities.Users.Add(obj);
objNoteMarketplaceEntities.SaveChanges();

// Generating Email Verification Link
var verifyUrl = "/Account/VerifyAccount/?VerificationCode=" + activationCode;
var activationlink = Request.Url.AbsoluteUri.Replace(Request.Url.PathAndQuery
verifyUrl);

var SupportEmailAddress =
objNoteMarketplaceEntities.SystemConfigurations.Where(x => x.Key.ToLower()
== "supportemailaddress").Select(y => y.Value).FirstOrDefault();
var EmailPassword = objNoteMarketplaceEntities.SystemConfigurations.Where(x
=> x.Key.ToLower() == "emailpassword").Select(y => y.Value).FirstOrDefault();

// Sending Email
EmailVerification.SendVerificationLinkEmail(SupportEmailAddress,
EmailPassword, obj, activationlink);


//Redirect To VerifyEmail Page
TempData["userName"] = userName;
return new RedirectResult(@"~\Account\VerifyEmail\");

}
return View();
}
```

5.2.1.3 Verify Account

```csharp
[HttpGet]
public ActionResult VerifyAccount(string VerificationCode)
{
  using (NoteMarketplaceEntities dc = new NoteMarketplaceEntities())
  {
        dc.Configuration.ValidateOnSaveEnabled = false; // This line I have added
        here to avoid
        // Confirm password does not match issue on save changes
        var v = dc.Users.Where(x => x.VerificationCode ==
        VerificationCode).FirstOrDefault();
    if (v != null)
    {
      v.IsEmailVerified = true;
      v.IsActive = true;
      dc.SaveChanges();
      @TempData["Message"] = "Your Email Is Verified You Can Login Here";
      return RedirectToAction("Login", "Account");
```

```
        }
        else
        {
          @TempData["Error"] = "Invalid Request";
          return RedirectToAction("Login", "Account");
        }
      }
    }
```

5.2.2 Add Note Module

```
public class AddNoteController : Controller
  {
    private readonly NoteMarketplaceEntities dbobj = new NoteMarketplaceEntities();

    [Route("AddNote")]
    public ActionResult AddNote()
    {
      ViewBag.NotesCategory = dbobj.NoteCategories.Where(x => x.IsActive == true);
      ViewBag.NotesType = dbobj.NoteTypes.Where(x => x.IsActive == true);
      ViewBag.Country = dbobj.Countries.Where(x => x.IsActive == true);
      return View();
    }

    [Route("AddNote")]
    [HttpPost]
    public ActionResult AddNote(AddNote note)
    {
      if (ModelState.IsValid)
      {

        //Check UploadNote Is Selected Or Not
        if (note.UploadNotes[0] == null)
        {
          TempData["notice"] = "Select File to upload";
            ViewBag.NotesCategory = dbobj.NoteCategories.Where(x => x.IsActive ==
            true);
          ViewBag.NotesType = dbobj.NoteTypes.Where(x => x.IsActive == true);
          ViewBag.Country = dbobj.Countries.Where(x => x.IsActive == true);
          return View();
        }

        //Check SellingPrice is included for paid note or not
        if (note.IsPaid == true && note.SellingPrice == null)
        {
          TempData["noticeprice"] = "Enter The Price";
            ViewBag.NotesCategory = dbobj.NoteCategories.Where(x => x.IsActive ==
            true);
          ViewBag.NotesType = dbobj.NoteTypes.Where(x => x.IsActive == true);
          ViewBag.Country = dbobj.Countries.Where(x => x.IsActive == true);
```

```
            return View();
        }


        //Check NotePreView is included for paid note or not
        if (note.IsPaid == true && note.NotesPreview == null)
        {
            TempData["noticePreview"] = "Note Preview Is Required For Paid Notes";
                ViewBag.NotesCategory = dbobj.NoteCategories.Where(x => x.IsActive ==
                true);
            ViewBag.NotesType = dbobj.NoteTypes.Where(x => x.IsActive == true);
            ViewBag.Country = dbobj.Countries.Where(x => x.IsActive == true);
            return View();
        }

        //get userid of logedin user
        var EmailID = User.Identity.Name.ToString();
        User userObj = dbobj.Users.Where(x => x.EmailID == EmailID).FirstOrDefault();
                string path = Path.Combine(Server.MapPath("~/Members"),
                userObj.ID.ToString());

        // Check for Directory, If not exist, then create it
        if (!Directory.Exists(path))
        {
            Directory.CreateDirectory(path);
        }

        //Check User Submitted Publish button or Save Button
        var noteDataValue = "Draft";


        // get Note Status based on user clicked on Save or Publish
            ReferenceData referenceData = dbobj.ReferenceDatas.Where(x =>
            x.RefCategory == "Notes Status" && x.DataValue == noteDataValue &&
            x.IsActive == true).FirstOrDefault();

        //Save Book Details
        SellerNote objSellerNote = new SellerNote()
        {
            SellerID = userObj.ID,
            Status = referenceData.ID,
            Title = note.Title,
            Category = note.Category,
            Description = note.Description,
            IsPaid = note.IsPaid,
            NoteType = note.NoteType,
            NumberofPages = note.NumberofPages,
            UniversityName = note.UniversityName,
            Country = note.Country,
            Course = note.Course,
            CourseCode = note.CourseCode,
```

44

```csharp
            Professor = note.Professor,
            CreatedDate = DateTime.Now,
            SellingPrice = note.SellingPrice,
            CreatedBy = userObj.ID,
            IsActive = true
        };

    //Save Note To Database
    dbobj.SellerNotes.Add(objSellerNote);
    dbobj.SaveChanges();


    //Get Saved Notes ID

    var noteID = objSellerNote.ID;

    //Generate Path To Store Image
            string storepath = Path.Combine(Server.MapPath("~/Members/" +
userObj.ID), noteID.ToString());

    // Check for Directory, If not exist, then create it
    if (!Directory.Exists(storepath))
    {
        Directory.CreateDirectory(storepath);
    }

    //Store The DisplayPicture If Uploaded
    if (note.DisplayPicture != null && note.DisplayPicture.ContentLength > 0)
    {
                    string fileName =
            Path.GetFileNameWithoutExtension(note.DisplayPicture.FileName);
        string extension = Path.GetExtension(note.DisplayPicture.FileName);
        fileName = "DP_" + DateTime.Now.ToString("ddMMyyyy") + extension;
        string finalpath = Path.Combine(storepath, fileName);
        note.DisplayPicture.SaveAs(finalpath);

        objSellerNote.DisplayPicture = Path.Combine(("/Members/" + userObj.ID + "/"
        + noteID + "/"), fileName);
        dbobj.SaveChanges();
    }
    else
    {
                    SystemConfiguration systemConfiguration =
            dbobj.SystemConfigurations.Where(x => x.Key.ToLower() ==
            "defaultimagefornotes").FirstOrDefault();
        objSellerNote.DisplayPicture = systemConfiguration.Value;
        dbobj.SaveChanges();
    }

    //Store The NotesPreview If Uploaded
    if (note.NotesPreview != null && note.NotesPreview.ContentLength > 0)
```

```csharp
{
                string fileName =
        Path.GetFileNameWithoutExtension(note.NotesPreview.FileName);
    string extension = Path.GetExtension(note.NotesPreview.FileName);
    fileName = "Preview_" + DateTime.Now.ToString("ddMMyyyy") + extension;
    string finalpath = Path.Combine(storepath, fileName);
    note.NotesPreview.SaveAs(finalpath);

    objSellerNote.NotesPreview = Path.Combine(("/Members/" + userObj.ID + "/" +
    noteID + "/"), fileName);
    dbobj.SaveChanges();
}

//Create Path To Store Attachement
string attachementsstorepath = Path.Combine(storepath, "Attachements");

// Check for Directory, If not exist, then create it
if (!Directory.Exists(attachementsstorepath))
{
    Directory.CreateDirectory(attachementsstorepath);
}

//Create Object of SellerNotesAttachement Table and Store Data
SellerNotesAttachement sellerNotesAttachement = new SellerNotesAttachement
{
    NoteID = noteID,
    ISActive = true,
    CreatedBy = userObj.ID,
    CreatedDate = DateTime.Now
};

//Store The Attached File
int Count = 1;
var FilePath = "";
var FileName = "";
long FileSize = 0;
foreach (var file in note.UploadNotes)
{
    FileSize += ((file.ContentLength) / 1024);
    string fileName = Path.GetFileNameWithoutExtension(file.FileName);
    string extension = Path.GetExtension(file.FileName);
                fileName = "Attachement" + Count + "_" +
        DateTime.Now.ToString("ddMMyyyy") + extension;
    string finalpath = Path.Combine(attachementsstorepath, fileName);
    file.SaveAs(finalpath);

    FileName += fileName + ";";
                FilePath += Path.Combine(("/Members/" + userObj.ID + "/" +
        noteID + "/Attachements/"), fileName) + ";";

    Count++;
```

```
            }

            //Save the Detail To DataBase
            sellerNotesAttachement.AttachementSize = FileSize;
            sellerNotesAttachement.FileName = FileName;
            sellerNotesAttachement.FilePath = FilePath;
            dbobj.SellerNotesAttachements.Add(sellerNotesAttachement);
            dbobj.SaveChanges();

            TempData["success"] = userObj.FirstName+ " " + userObj.LastName;
            TempData["message"] = "Note has been added";
            return RedirectToAction("Index", "User");
        }
        ViewBag.NotesCategory = dbobj.NoteCategories.Where(x => x.IsActive == true);
        ViewBag.NotesType = dbobj.NoteTypes.Where(x => x.IsActive == true);
        ViewBag.Country = dbobj.Countries.Where(x => x.IsActive == true);
        return View();
    }
}
```

5.2.3 Admin Notes Under Review Module

5.2.3.1 Approve Notes

```
[Route("Approve/{id}")]
    [HttpGet]
    public ActionResult Approve(int? id)
    {

        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }

        var Emailid = User.Identity.Name.ToString();
        User user = dbobj.Users.Where(x => x.EmailID == Emailid).FirstOrDefault();

        SellerNote note = dbobj.SellerNotes.Find(id);

        if (note == null)
        {
            return RedirectToAction("Error", "Home");
        }

        note.Status = dbobj.ReferenceDatas.Where(x => x.RefCategory == "Notes Status"
        && x.Value == "Published").Select(x => x.ID).FirstOrDefault();
        note.ModifiedBy = user.ID;
        note.ModifiedDate = DateTime.Now;
        note.ActionedBy = user.ID;
        note.PubilshedDate = DateTime.Now;
```

```
        dbobj.Entry(note).State = EntityState.Modified;
        dbobj.SaveChanges();

        TempData["success"] = user.FirstName + " " + user.LastName;
        TempData["message"] = "Note has been Approved";
        return RedirectToAction("NotesUnderReview", "Admin");
    }
```

5.2.3.2 In Review Notes

```
    [Route("InReview/{id}")]
    [HttpGet]
    public ActionResult InReview(int? id)
    {

        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }

        var Emailid = User.Identity.Name.ToString();
        User user = dbobj.Users.Where(x => x.EmailID == Emailid).FirstOrDefault();

        SellerNote note = dbobj.SellerNotes.Find(id);

        if (note == null)
        {
            return RedirectToAction("Error", "Home");
        }

        note.Status = dbobj.ReferenceDatas.Where(x => x.RefCategory == "Notes Status"
        && x.Value == "In Review").Select(x => x.ID).FirstOrDefault();
        note.ModifiedBy = user.ID;
        note.ModifiedDate = DateTime.Now;
        note.ActionedBy = user.ID;
        dbobj.Entry(note).State = EntityState.Modified;
        dbobj.SaveChanges();

        TempData["success"] = user.FirstName + " " + user.LastName;
        TempData["message"] = "Note Status has been changed";
        return RedirectToAction("NotesUnderReview", "Admin");
    }
```

5.2.3.3 Reject Notes

```
    [Route("RejectedNote/{id}")]
    [HttpGet]
    public ActionResult RejectedNote(int? id, AdminRejectRemark adminremark)
    {
```

```
            var Emailid = User.Identity.Name.ToString();
            User user = dbobj.Users.Where(x => x.EmailID == Emailid).FirstOrDefault();

            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }

            SellerNote note = dbobj.SellerNotes.Find(id);

            if (note == null)
            {
                return RedirectToAction("Error", "Home");
            }

            note.Status = dbobj.ReferenceDatas.Where(x => x.RefCategory == "Notes Status"
            && x.Value.ToLower() == "rejected").Select(x => x.ID).FirstOrDefault();
            note.ModifiedBy = user.ID;
            note.ModifiedDate = DateTime.Now;
            note.AdminRemarks = adminremark.Remarks;
            note.ActionedBy = user.ID;
            dbobj.Entry(note).State = EntityState.Modified;
            dbobj.SaveChanges();


            TempData["success"] = user.FirstName + " " + user.LastName;
            TempData["message"] = "Note has been Rejected";
            return RedirectToAction("NotesUnderReview", "Admin");
        }
```

5.2.4 Add Review

```
[HttpPost]
    [Route("AddReview/{id}")]
    public ActionResult AddReview(int? id, ReportAsInappropriate review)
    {
        var EmailID = User.Identity.Name.ToString();
        User userObj = dbobj.Users.Where(x => x.EmailID == EmailID).FirstOrDefault();

        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }

        Download downloadrecord = dbobj.Downloads.Find(id);

        if (downloadrecord == null)
        {
            return RedirectToAction("Error", "Home");
        }
```

```csharp
        SellerNotesReview sellerNotesReview = new SellerNotesReview
        {
            NoteID = downloadrecord.NoteID,
            ReviewedBy = userObj.ID,
            AgainstDownloadsID = downloadrecord.ID,
            Ratings = review.Ratings,
            Comments = review.Remarks,
            CreatedDate = DateTime.Now,
            CreatedBy = userObj.ID,
            ModifiedDate = DateTime.Now,
            ModifiedBy = userObj.ID,
            Isactive = true
        };

        dbobj.SellerNotesReviews.Add(sellerNotesReview);
        dbobj.SaveChanges();

        return RedirectToAction("MyDownloads", "User");
    }
```

5.2.5 Download Attachment

```csharp
public ActionResult DownloadAttechedFile(int? id)
    {

        Download download = dbobj.Downloads.Find(id);

        download.IsAttachementDownloaded = true;
        download.AttacmentDownloadedDate = DateTime.Now;
        download.ModifiedDate = DateTime.Now;

        dbobj.Entry(download).State = EntityState.Modified;
        dbobj.SaveChanges();

        SellerNotesAttachement attachment = dbobj.SellerNotesAttachements.Where(x =>
        x.NoteID == download.NoteID).FirstOrDefault();

        var allFilesPath = attachment.FilePath.Split(';');
        using (var memoryStream = new MemoryStream())
        {
                using (var ziparchive = new ZipArchive(memoryStream,
        ZipArchiveMode.Create, true))
          {
            foreach (var FilePath in allFilesPath)
            {
              string FullPath = Path.Combine(Server.MapPath("~" + FilePath));
              string FileName = Path.GetFileName(FullPath);
              if (FileName == "")
              {
```

```
                    continue;
                }
                else
                {
                    ziparchive.CreateEntryFromFile(FullPath, FileName);
                }
            }
        }
        return File(memoryStream.ToArray(), "application/zip", "Attachments.zip");
    }
}
```

5.3 Security features

Feature-1: Email Verification

Once non-registered user does the signup – system need to send email verification link to registered email and then we need to redirect user to an Email Verification Page.

User need to click the link which system has sent over registered email to verify the email address. Until user does the verification of email address – User should not be able to login into the system. See 5.1 for identifying email template. Once Email Verification is done - We need to redirect user first time to My profile page after successful login so they can fill the remaining profile information (as user has not filled in that) instead Search Notes Screen. Until user do not verify the email address, every time user does login we need to redirect to this page and user should not able to do actual login.

Notes MarketPlace
Learning is a lifelong process...

Email Verification

Dear Smith,

Thanks for Signing up!
Simply click below for email verification.

VERIFY EMAIL ADDRESS

Feature-2: Forgot Password

There is another feature introduced here which is forgot password clause.

When user wants to log in and he forgot the password then he/she can retrieve the password.

By clicking on forgot password system will redirect to a page where we have to enter Email and system generate a random password and sent it to users registered email address.
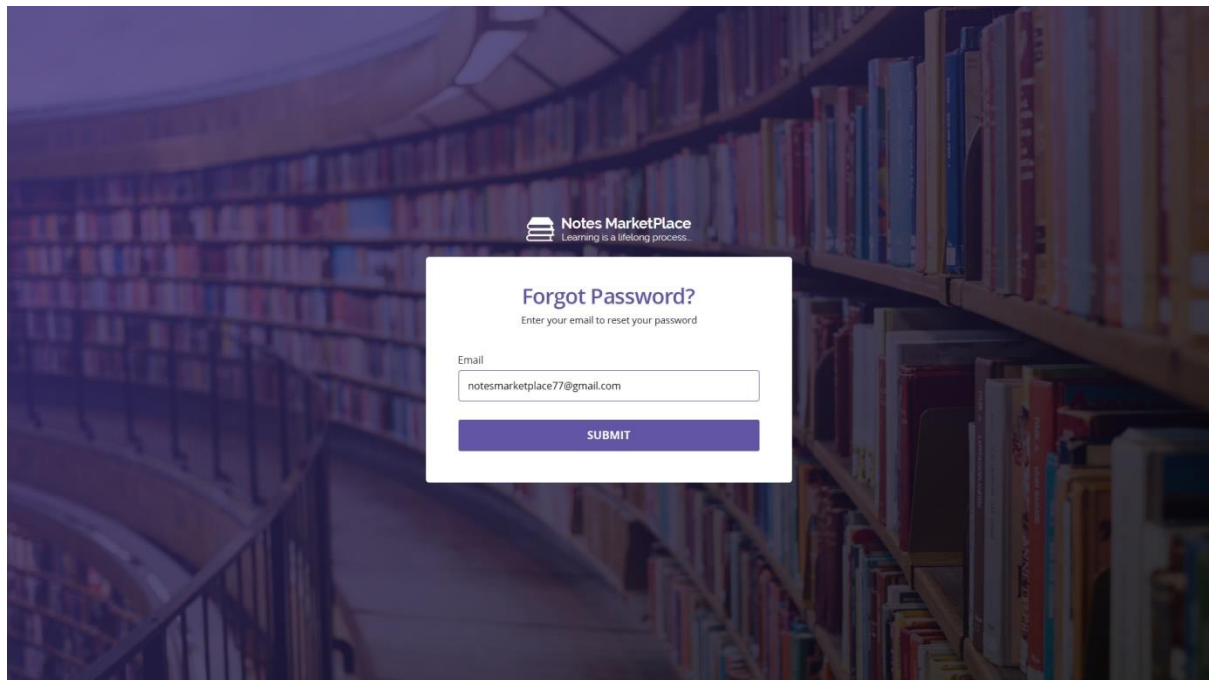
5.4 Important screenshots

Home Page

Signup Page

Login Page

Search Note Page

Notes MarketPlace
Learning is a lifelong process..

Search Notes

Search and Filter notes

🔍 Search notes here...

| Select type ∨ | Select category ∨ | Select university ∨ | Select course ∨ | Select country ∨ | Select rating ∨ |

Total 18 notes

Computer Operating System - Final Exam Book With Paper Solution
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★★★ 100 reviews

Computer Science
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★★☆ 100 reviews

Basic Computer Engineering Tech India Publication Series
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★☆☆ 100 reviews
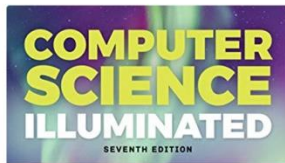
Computer Science Illuminted - Seventh Edition
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★★★ 100 reviews

The Principles of Computer Hardware - Oxford
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★★☆ 100 reviews
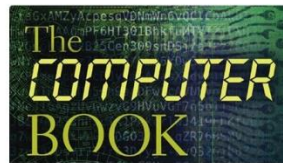
The Computer Book
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★☆☆ 100 reviews

Computer Operating System - Final Exam Book With Paper Solution
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★★★ 100 reviews

Computer Science
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★★☆ 100 reviews

Basic Computer Engineering Tech India Publication Series
🏛 University of California, US
📖 204 Pages
📅 Thu, Nov 26 2020
🏳 5 Users marked this note as inappropriate
★★★☆☆ 100 reviews

‹ 1 2 3 4 5 ›

Note Detail Page

User Dashboard

Add Notes Page

Add Notes

Basic Note Details

Title *

Enter your notes title

Category *

Select your category

Display Picture

Upload a picture

Upload Notes *

Upload your notes

Type

Select your note type

Institution Information

Country

Select your country

Institution Name

Enter your institution name

Course Details

Course Name

Enter your course name

Course Code

Enter your course code

Professor / Lecturer

Enter your professor name

Description *

Enter your description

Selling Information

Sell For *
○ Free   ● Paid

Note Preview

Upload a file

Sell Price *

Enter your price

ADD NOTES

Member Detail Page

Notes MarketPlace
Learning is a lifelong process..

Dashboard   Notes   Members   Reports   Settings   Logout

Member Details

First Name:	Richard	Address 1:	144-Diamond Height
Last Name:	Brown	Address 2:	Star Colony
Email:	richardbrown77@gmail.com	City:	New York
DOB:	13-08-1990	State:	New York State
Phone Number:	9988731221	Country:	United State
College/University:	University of California	Zip Code:	11004-05

Notes

SR NO.	NOTE TITLE	CATEGORY	STATUS	DOWNLOADED NOTES	TOTAL EARNINGS	DATE ADDED	PUBLISHED DATE	
1	Software Development	IT	Published	22	$177	09-10-2020, 10:10	09-10-2020, 10:10	⋮
2	Computer Basic	Computer	Published	4	$125	10-10-2020, 11:25	Download Notes	
3	Human Body	Science	InReview	17	$978	11-10-2020, 01:00	11-10-2020, 01:00	⋮
4	World War 2	History	Published	28	$15254	12-10-2020, 10:10	12-10-2020, 10:10	⋮
5	Accountng	Account	Published	0	$69	13-10-2020, 11:25	NA	⋮

‹ **1** 2 3 4 5 ›

Version : 1.1.24

Admin Dashboard

Notes MarketPlace
Learning is a lifelong process..

Dashboard   Notes   Members   Reports   Settings   Logout

Update Profile
Change Passsword
LOGOUT

Dashboard

20	103	223
Numbers of Notes in Review for Publish	Numbers of New Notes Downloaded (Last 7 days)	Numbers of New Registrations (Last 7 days)

Published Notes

Search   **SEARCH**   Select month

SR NO.	TITLE	CATEGORY	ATTACHMENT SIZE	SELL TYPE	PRICE	PUBLISHER	PUBLISHED DATE	NUMBER OF DOWNLOADS	
1	Data Science	Science	10 KB	Free	$0	Pritesh Panchal	09-10-2020, 10:10	10	⋮
2	Accounts	Commerce	23 MB	Paid	$22	Rahil Shah	10-10-2020, 12:30	Download Notes	
3	Social Studies	Social	3 MB	Paid	$56	Anish Patel	11-10-2020, 01:00	View More Details	
4	AI	IT	1 MB	Free	$0	Vijay Vaishnav	12-10-2020, 10:10	Unpublish	
								34	⋮
5	Lorem ipsum	Lorem	105 KB	Paid	$90	Mehul Patel	13-10-2020, 11:25	9	⋮

‹ **1** 2 3 4 5 ›

Version : 1.1.24

Admin In Progress Notes Dashboard

Notes MarketPlace
Learning is a lifelong process...

Dashboard   Notes   Members   Reports   Settings   Logout

Notes Under Review

Seller
Khayati

Search   SEARCH

SR NO.	NOTE TITLE	CATEGORY	SELLER		DATE ADDED	STATUS	ACTION		
1	Software Engineer	IT	Khyati Patel	👁	09-10-2020, 10:10	InReview	Approve	Reject	InReview ⋮
2	Computer Basic	Computer	Khyati Patel	👁	10-10-2020, 11:25	Sunmitted For Review	Approve	Reject	InReview ⋮
3	Human Body	Science	Khyati Patel	👁	11-10-2020, 01:00	InReview	Approve	Reject	InReview ⋮
4	world war 2	History	Khyati Patel	👁	12-10-2020, 10:10	InReview	Approve	Reject	InReview ⋮
5	Accounting	Account	Khyati Patel	👁	13-10-2020, 11:25	InReview	Approve	Reject	InReview ⋮

‹ 1 2 3 4 5 ›

Version : 1.1.24

Copyright © Tatva

60

CHAPTER-6 SYSTEM TESTING

6.1 Introduction

Testing presents an interesting anomaly for the software engineering activities, the engineer attempts to build software from an abstract concept to a tangible product. Now comes testing. The engineer creates a series of test case that are initiated to "demolish" the software that has been build. Infect, testing is the one step in the software process that could be viewed (psychologically, at least) as destructive rather than constructive.

6.2 Modules of Testing

There are different Models of testing. On the basis of testing methods there are two types of testing:

1. Black-box testing.
2. White-box testing

Black-box tests are used to demonstrate that software functions are operational, that input is properly accepted and output is correctly produced, and that integrity of external information is maintained.

White-box tests are used to examine the procedural details. It checks the logical paths by test case. It can also checks the conditions, loops used in the software coding. It checks that loops are working correctly on defined boundary value.

1) White-Box Testing:

- White-box testing some times called glass-box testing, is a test case design method that users the control structure of the procedural design to drive the test case.

- Always we are thinking that there is no necessary to execute or checks the loops and conditions. And so large number of errors is uncovered. With using white-box testing methods, we have checked that,

- All independent paths within a function have been executed at least once.

- A11 loops working correctly at their boundary values and within their specified conditions.

- In our coding we test that all the loops works truly in each module. The one technique of white-box testing is basis path testing. It contains two parts, one is flow graph notation and the second is cyclometer complexity. In flow graph notation we are checking logical control of flow. By using cyclometer complexity we find complexity of our project structure.

2) Block-box Testing

- Black-box testing focuses on the functional requirements of the software. That is black-box testing enables the software engineer to drive sets of input conditions that will fully exercise all functional Requirements for the program.

- Black-box testing is not an alternative to white-box testing techniques. Rather, it is a complementary approach that is likely to uncover a different class of errors than white-box methods.

- We use in our coding to find errors in the following categories:

- Incorrect or missing functions

- Interface errors

- Errors in database

- Performance errors

- Initialization and termination errors.

- Unlike white-box testing, which is performed earlier in the testing process, black-box testing tends to be applied during later stages of testing. Because black-box testing purposely disregards control structure, attention is focused on the information domain.

- By applying black-box techniques, we derive a set of test cases that satisfy following criteria
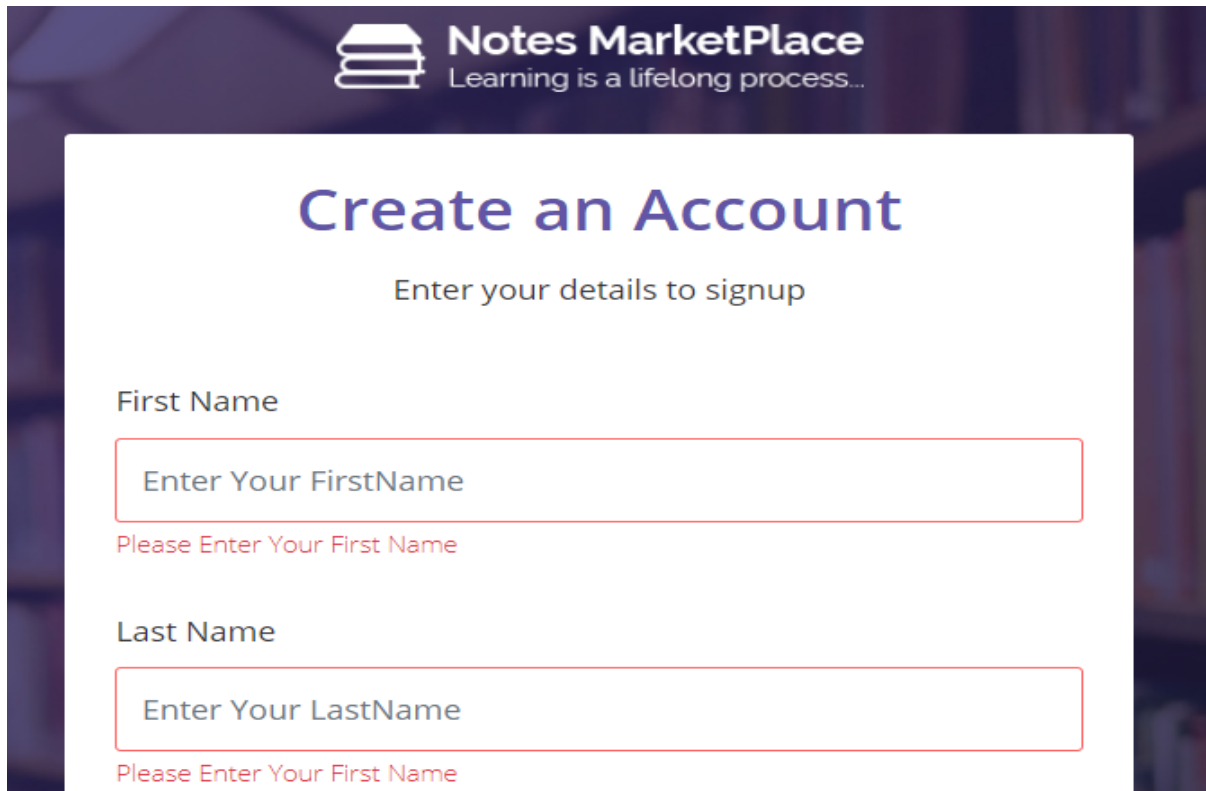
Test cases that reduce, by a count that is greater then one, the number of additional test cases must be designed to achieve reasonable testing.

Test cases that tell us something about the present or absence of classes of errors, rather then an error associated only with the specific test at hand.

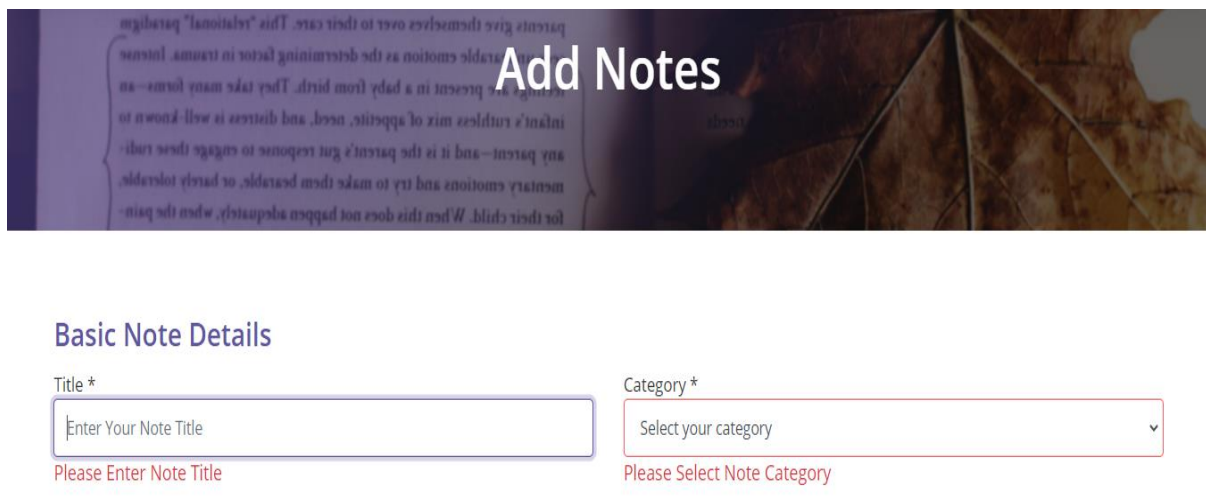## 6.3 Testing of The System

6.3.1 Register

      It will give some validation error for register if user does not provide the details like enter the first name and last name.
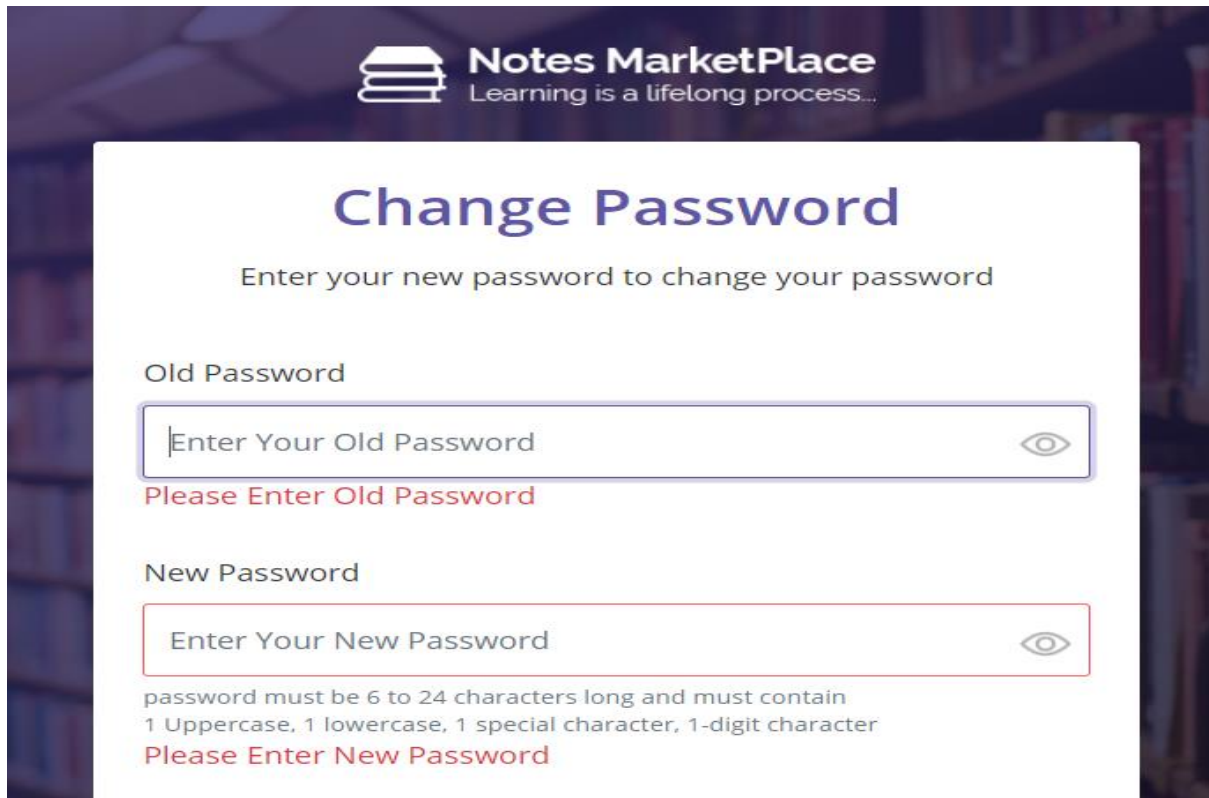

6.3.2 Add Note

      It will give some validation error for Add note if user does not provide the details for it shown below.

6.3.3 Change Password

It will give some validation error for change Password if user does not provide the details like old password and new password.
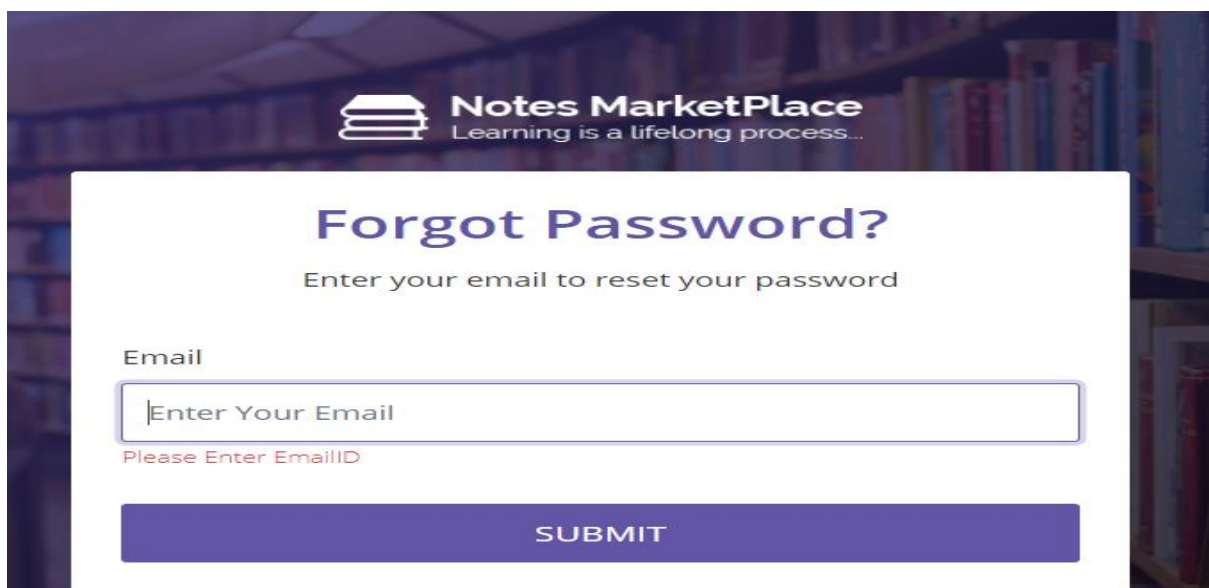
6.3.4 Forgot Password

6.3.5 404 Error Page

Oops. The page you're looking for doesn't exist.

Back Home

CONCLUSION

- This System denotes the interaction with the user. Here we conclude that we have completed the System which is working smoothly without any bugs.
- Using this system user can search for particular notes and request to buy that note to the seller, seller can approve and reject the request of the buyer. After approve the request user will be able to download the book from his/her my download tab in the navigation bar.
- Here the main role is of admin, who can log in and approve the note to publish on the portal he can also manage the user activity on the portal admin can unpublished the particular book also for some reason.