ONLINE BOOK STORE

A PROJECT REPORT

SUBMITTED BY

MUSKAN TOMAR

(University Roll No:2000290140074)

A Thesis Submitted
in Partial Fulfillment of the Requirements
for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of

Mr. Ankit Verma

ASSISTANT PROFESSOR
KIET Group of Institution, Ghaziabad

DR. APJ ABDUL KALAM TECHNICAL UNIVERSITY,
(Formerly Uttar Pradesh Technical University) LUCKNOW
2021-2022

DECLARATION

We hereby declare that the work presented in this report entitled “ONLINE BOOK STORE ", was carried out by us. We have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute.

We have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. We have used quotation marks to identify verbatim sentences and given credit to the original authors/sources.

Name : **MUSKAN TOMAR**

Roll. No. : **University Roll No. 2000290140074**

Field : **M.C.A. 3rd Semester**

Muskan Tomar

(Candidate Signature)

ACKNOWLEGMENT

Here, I gladly present this project report on **"Online Book Store"** as part of the 3rd semester MCA Master in Computer Applications. I take this occasion to thank God, almighty for blessing me with his grace and taking our endeavor to a successful culmination. I extend my sincere and heartfelt thanks to me esteemed guide, for providing me with the right guidance and advice at the crucial junctures and for showing me the right way. I have many understanding friends, who have helped me a lot on many critical conditions. Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Muskan Tomar

CETIFICATE

Certified that **Muskan Tomar** (**University Roll No.2000290140074),** have carried out the project work having "Online Book Store" for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Muskan Tomar
(University Roll No. 2000290140074)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date :

Mr. Ankit Verma
Associate Professor
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Signature of Internal Examiner **Signature of External Examiner**

Dr. Ajay Shrivastava
Head, Department of Computer Applications
KIET Group of Institutions, Ghaziabad

TABLE OF CONTENTS

Declaration.....2

Certificate 3

Abstract.....4

Acknowledge 5

Abstract….....08

CHAPTER- I

Introduction.....09-10

CHAPTER II

Literature View. 11

CHAPTER-III

Advantages to both end user and developer 12

System Analysis 13-15

Existing System… 13

Proposed System 13-14

Advantages of Proposed System… 14-15

CHAPTER-IV

Feasibility Study..... 15-16

Economic Feasibility 15

Technical Feasibility.....16

Operational Feasibility.....16

CHAPTER-V

Problem Definition… 17-18

Result and Output Analysis 19-21

Hardware Requirement…..... 19

Software Requirement… 20

Design Constraints 21

System Features…..... 21

CHAPTER-VI

System Overview 22-30

Use Case Diagram.....22

Class Diagram… 23

Sequence Diagram..... 24-25

ER Diagram…..... 26

Use Case Definition and Diagram… 27-30

CHAPTER-VII

Module Description..... 30-31

Administrator Module… 30

Doctor Module… 31

Appointment and Feedback Module 31

CHAPTER-VIII

Database Structure..... 32-34

Admin Login… 32

Appointment…..... 32

Doctor… 33

Feedback… 33

Patients….....34

CHAPTER-IX

System Implementation and Testing 34-37

System Implementation… 34-35

System Testing.....35

Unit Testing 36

Integration Testing.....36

Validation Testing… 37

CHAPTER-X

Code Snippets… 38-39

CHAPTER-XI

Screenshots… 40-46

Home 40

Admin Homepage 41

Doctor Details… 42

Doctor Login… 42

Doctor Homepage 43

Doctor List 43

Patient Homepage 44

Book an Appointment.....44

Patient Details 45

About.....46

Common Disease 46

CHAPTER- XII

Conclusion… 47-49

Future Scope 41

References.....53

LIST OF TABLES

- TABLE 11.1 ADMIN LOGIN 34
- TABLE 11.2 APPOINMNET 34
- TABLE 11.3 DOCTOR 35
- TABLE 11.4 FEEDBACK. 35
- TABLE 11.5 PATIENTS. 26

LIST OF FIGURES

- FIGURE 9.1 USE CASE DIAGRAM 24
- FIGURE 9.2 CLASS OF DIAGRAM..... 25
- FIGURE 9.3 SEQUENCE DIAGRAM. 26
- FIGURE 9.4 SEQUENCE DIAGRAM. 27
- FIGURE 9.5 E-R DIAGRAM. 28

Abstract

The main objective of the project is to create an online book store that allows users to search and purchase a book online based on title, author and subject. The selected books are displayed in a tabular format and the user can order their books online through credit card payment. Using this Website the user can purchase a book online instead of going out to a book store and wasting time.

There are many online book stores like Powell's, Amazon which were designed using Html. I want to develop a similar website using Node JS , SQL Server.

Online Book store is an online web application where the customer can purchase books online. Through a web browser the customers can search for a book by its title or author, later can add to the shopping cart and finally purchase using credit card transaction. The user can login using his account details or new customers can set up an account very quickly. They should give the details of their name, contact number and shipping address. The user can also give feedback to a book by giving ratings on a score of five. The books are divided into many categories based on subject like Software, Database, English, Architecture etc. The Online Book Store Website provides customers with online shopping through a web browser. A customer can, create, sign in to his account, place items into a shopping cart and purchase using his credit card details. The Administrator will have additional functionalities when compared to the common user. He can add, delete and update the book details, book categories, member information and also confirm a placed order. This application is developed using HTML, CSS , Node JS programming language.

The Master page, data sets, data grids, user controls are used to develop the Online Book store.

CHAPTER :- I

Introduction

1.1 Purpose and Motivation: The main objective of the project is to create an online book store that allows users to search and purchase a book based on title, author and subject. The selected books are displayed in a tabular format and the user can order their books online through credit card payment. The Administrator will have additional functionalities when compared to the common user. The motivation to create this project has many sources:

• Interest to develop a good user friendly website with many online transactions using a database.

• To increase my knowledge horizon in technologies like Node JS, SQL, CSS, HTML.

• To gain good experience in .NET before joining in a full time job.

• To gain expertise using Data Grid, Data Set, Data Table, Data Adapter and Data

Project Overview

2.1 Background : There are many online book stores like Powell's, Amazon which were designed using Html. I want to develop a similar website using Node JS, SQL Server.

Online Book store is an online web application where the customer can purchase books online. Through a web browser the customers can search for a book by its title or author, later can add to the shopping cart and finally purchase using credit card transaction. The user can login using his account details or new customers can set up an account very quickly. They should give the details of their name, contact number and shipping address. The user can also give feedback to a book by giving ratings on a score of five. The books are divided into many categories based on subject Like Software, Database, English, Architecture etc.

This project has the following functionalities:

1) **A Home page with product catalog**
   This is the page where the user will be navigated after a successful login. It will display all the book categories and will have a search keyword option to search for the required book. It also includes some special sections like recommended titles, weekly special books.
2) **Search**
   A search by keyword option is provided to the user using a textbox .The keyword to be entered should be

the book title.

3) Advanced Search

Advanced search helps the user to search for a book based on Title, Author, Category and price range. All the books which match the particular search criteria and their total count will be displayed .From here the user can select a book and add to the shopping cart.

4) Book Description

If the user would like to know details about a book he can click on the title from where he will be directed to a Book description page. It includes the notes on the book content and also a link to Amazon.com to get the book review.

5) User Voting

The user can give rating to a book based on his interest. He can rate it by giving a score of five as Excellent, four for very good, three for good, two for regular and one for deficient. The final rating of a book will depend on all the individual user rating.

6) Shopping Cart

The user can manage a shopping cart which will include all the books he selected. The user can edit, delete and update his shopping cart. A final shopping cart summary is displayed which includes all the items the user selected and the final total cost.

7**) Managing user accounts**

Each user should have an account to access all the functionalities of website. User can login using login page and logout using the logout page. All the user sessions will be saved in the database.

8) **Administration**

The Administrator will be provided with special functionalities like

- Add or delete a book category
- Add or delete a member
- Manage member orders.
- Add or delete a Credit Card type.

CHAPTER:- III

ADVANTAGES TO BOTH END USERS & DEVELOPERS

The system is useful in various ways as the information about the patients who are taking the free services from the health center all the details are already stored in the database, so the service is done in no time. All the information about the books are also maintained in the database

CHAPTER:- IV

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations involved in the feasibility analysis are:

• **Economic Feasibility**:

This study is carried out to check the economic impact will have on the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products have to be purchased.

• **Technical Feasibility:**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes for the implementing this system.

• **Operational Feasibility:**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity.

PROBLEM DEFINITION

Use Cases:

Search for a Book

• **Purpose:** A user can search for a book of his choice by selecting category and title. Then a select query is used to retrieve data from the database and display the selected information.

• **Actor:** User

• **Input:** The user will select a category and enter title in a text box provided.

• **Output:** The system will display the books which matches the selected search criteria.

A dataset is created as a result of select query. Later the dataset is binded to the data repeater to display the selected data.

Perform Advanced Search

• **Purpose**: If the user wants to perform an advanced search he can search for a book of his choice by selecting category, title, author and price range. Then a select query is used to retrieve data from the database and display the selected information.

• **Actor:** User

• **Input**: The user will select a category and enter title, author, and price range in a text box provided.

• **Output**: The system will display the books which matches the selected search criteria.

A dataset is created as a result of select query. Later the dataset is binded to the data repeater to display the selected data.

Give rating to a book

• **Purpose:** If the user wants to give rating according to his opinion for a book he can select either Excellent, Very good, good, regular or deficient. The final rating of a book will depend on all the individual user rating.

• **Actor**: User

• **Input**: The user will select a rating based on his opinion.

• **Output:** The system will display the rating of a book and the total number of votes received. Below is the display for various rating.

***** Excellent

**** Very Good

*** Good

** Regular

* Deficient

Maintain Account

1) **Register**

• **Purpose:** If the user doesn't have an account then he will be asked to register.

• **Actor:** User

• **Input:** The user will enter details in the registration form according to the required fields. The fields include

1. Username
2. Password
3. confirm password
4. first name
5. last name
6. email
7. Address
8. Phone
9. CC details

• **Output:** After registration the user will be directed to the main home page.

2) **Login**

• **Purpose:** If the user wants to get access to all the functionalities of Online Book Store he should login using his username and password.

• **Actor:** User

• **Input:** The user will enter his username and password.

• **Output:** If it is a successful login the user will be directed to the main home page. Else if the user enters invalid information he will be asked to check the entered information.

3) **Update Profile**

• **Purpose**: If the user wants to change his personal account information then he can update his selected fields and the entire data will be updated in the data base through an update query.

• **Actor**: User

• **Input**: The user will update his account information.

• **Output**: The system will update the entered information in the database using an update query.

4) **Logout**

• **Purpose**: If the user wants to end his session and sign out of the website then he can use the logout option.

• **Actor:** User 12

• **Input**: The user will click the logout button.

• **Output**: The user's account session comes to an end and he should login again if he wants to enter into the website.

Manage Shopping Cart

1) **Place an order**

• **Purpose**: If the user wants to purchase a book then he can place an order by selecting the add to shopping cart button and entering the quantity required under the book description.

• **Actor:** User

• **Input**: The user will enter the quantity required and click the add to shopping cart button.

• **Output**: The order will be added to the user's shopping cart.

2) **Update Shopping Cart**

• **Purpose**: If the user wants to change the quantity of a book or change a book then he can update his shopping cart.

• **Actor**: User

• **Input:** The user will click the details button in the shopping cart summary to edit and update his order details..

• **Output**: The updated order details are reflected in the shopping cart summary.

3) **View Shopping Cart**

• **Purpose**: If the user wants to view the items he added to the shopping cart then he can click the shopping cart link at the top of the page.

• **Actor**: User

• **Input**: The user will click the shopping cart link at the top of every page.

• **Output**: The user's shopping cart summary will be displayed in the form of a tabular format with all the books and their quantity. A total cost of all the items is also displayed at the bottom.

Administrator

1) **Login**

• **Purpose**: If the Administrator wants to get access to all the functionalities of Online Book Store he should login using his username and password.

• **Actor**: Administrator

• **Input**: The Administrator will enter his username and password. 13

• **Output:** If it is a successful login the Administrator will be directed to his menu page.

Else if the Administrator enters invalid information he will be asked to check the entered information.

2) **Add or Delete Category**

• **Purpose**: If the Administrator wants to add or delete a book category then he can insert or delete a book category using his administration rights and the category table will be updated in the database.

• **Actor**: Administrator

• **Input:** If the Administrator wants to add a book category the he should click the insert link button in the category page else he can delete a particular selected book category.

• **Output**: The updated categories list will be displayed in the main home page.

3) **Add or Delete Book**

• **Purpose**: If the Administrator wants to add or delete a book then he can insert or delete a book using his administration rights and the book table will be updated in the database.

• **Actor**: Administrator

• **Input**: If the Administrator wants to add a book the he should click the insert link button in the book page and fill the following fields related to the book.

1. Title
2. Author
3. Price
4. Category
5. Notes
6. Product url

If he wants to delete a book he can click the delete button to remove it from the database.

• **Output:** The updated books list will be displayed in the main home page under their particular category.

4) **Manage Orders**

• **Purpose**: If the Administrator wants to add or delete an order then he can insert or delete an order using his administration rights.

• **Actor**: Administrator

• **Input**: If the Administrator wants to add an order the he should click the insert link button in the orders page else he can delete a particular selected order

Output: The updated orders list will be processed to the users

.

5) **Add or Delete CC (Credit Card)**

• **Purpose**: If the Administrator wants to add or delete a CC type then he can insert or delete a CC type using his administration rights and the CC table will be updated in the database.

• **Actor:** Administrator

• **Input**: If the Administrator wants to add a CC type the he should click the insert link button in the CC page else he can delete a particular selected CC type..

• **Output**: The updated CC list will be displayed in registration page where the user will select it for his future transactions.

6) **Add or Delete Member**

• **Purpose**: If the Administrator wants to add or delete a book category then he can insert or delete a book category using his administration rights and the category table will be updated in the database.

• **Actor**: Administrator

• **Input**: If the Administrator wants to add a book category the he should click the insert link button in the category page else he can delete a particular selected book category.

• **Output:** The updated categories list will be displayed in the main home page.

7) **Logout**

• **Purpose**: If the Administrator wants to end his session and sign out of the website then he can use the logout option.

• **Actor**: Administrator

• **Input:** The Administrator will click the logout button.

• **Output**: The Administrator's account session comes to an end and he should login again if he wants to enter into the website

CHAPTER:- V

SYSTEM REQUIREMENTS

• HARDWARE REQUIREMENTS:

The section of hardware configuration is an important task related to the software development insufficient random-access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

Processor	: Dual core and above
Processor speed	: 1.4 GHz Onwards
Cache size	: 512 KB
RAM	: 4 GB (Minimum)
Hard disk	: Minimum (50 GB)
Monitor	: SVGA Color 15”

• SOFTWARE REQUIREMENTS

A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system. This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. It allow the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

Front end tool	: JSP, HTML, CSS as scripting language
Backend	: JAVA(Servlet), MY-SQL
Operating system	: Windows 7-10
Client Side	: HTML, CSS, JSP
IDE	: Eclipse, NetBeans

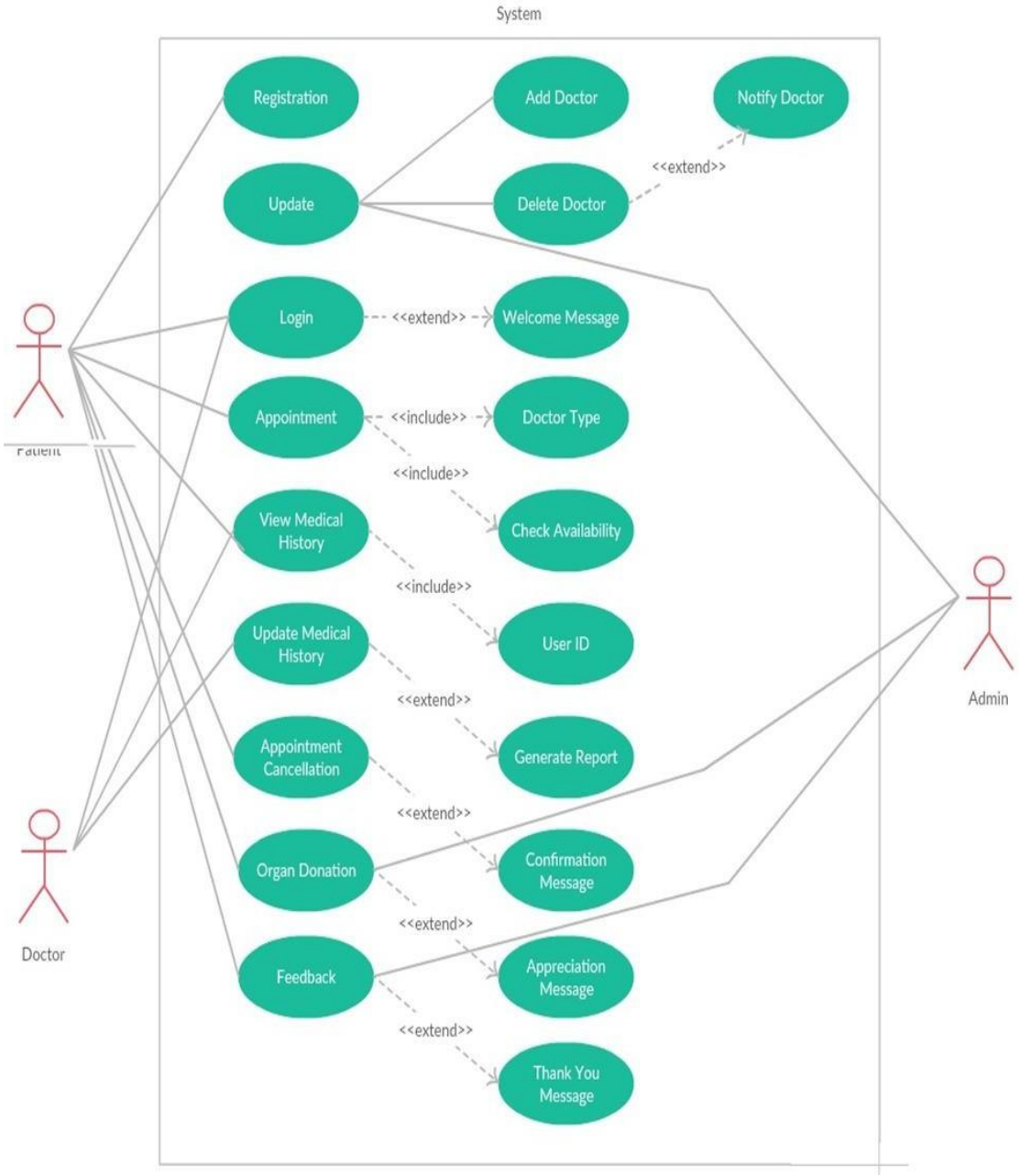
• **DESIGN CONSTRAINTS:**

This Online Book Store require huge resources as Hundreds of the customers will require the services instantly, quick response time are needed. The database should also be very large and robust to maintain very huge customers and books data.

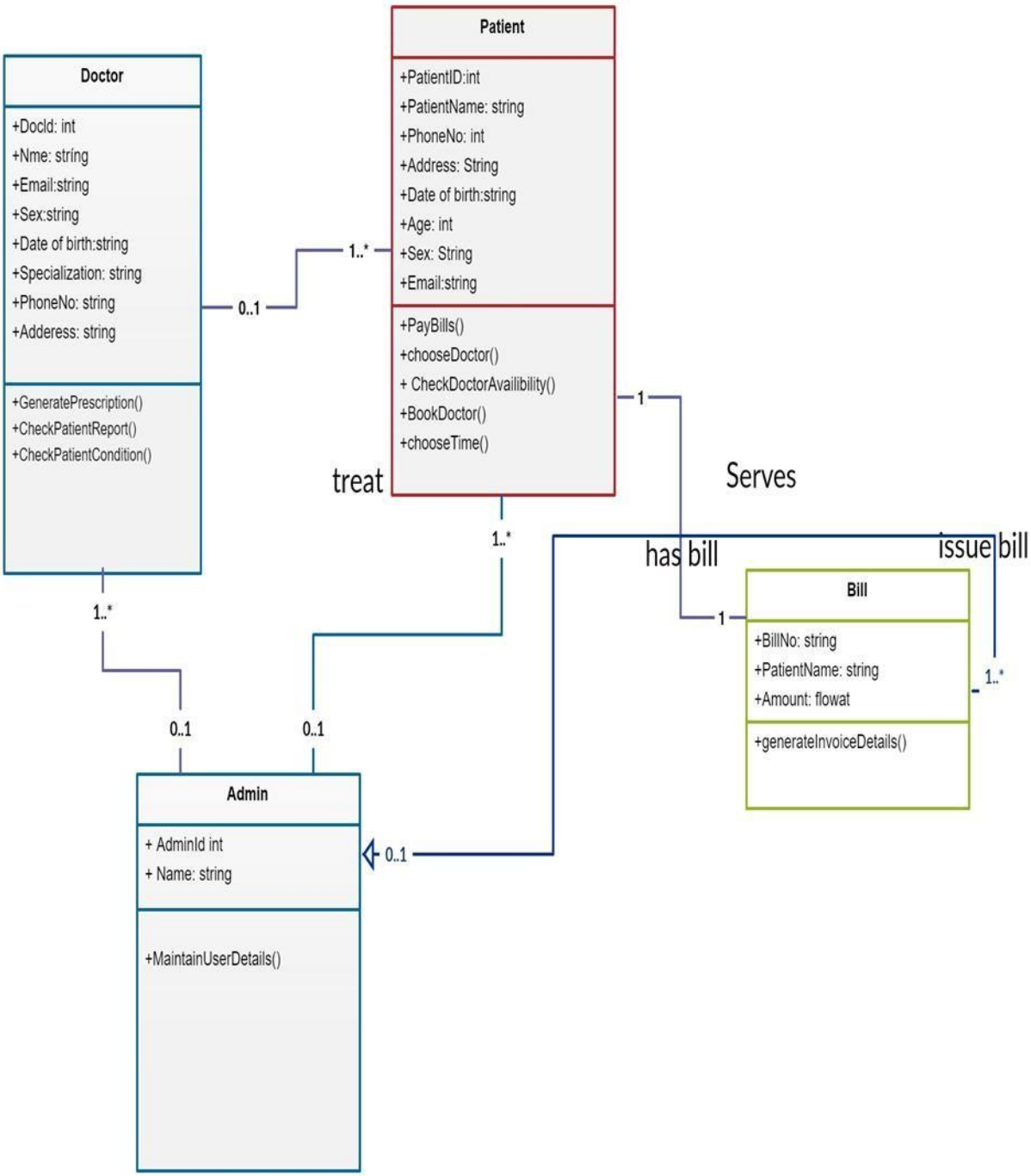
CHAPTER:- VI

SYSTEM OVERVIEW

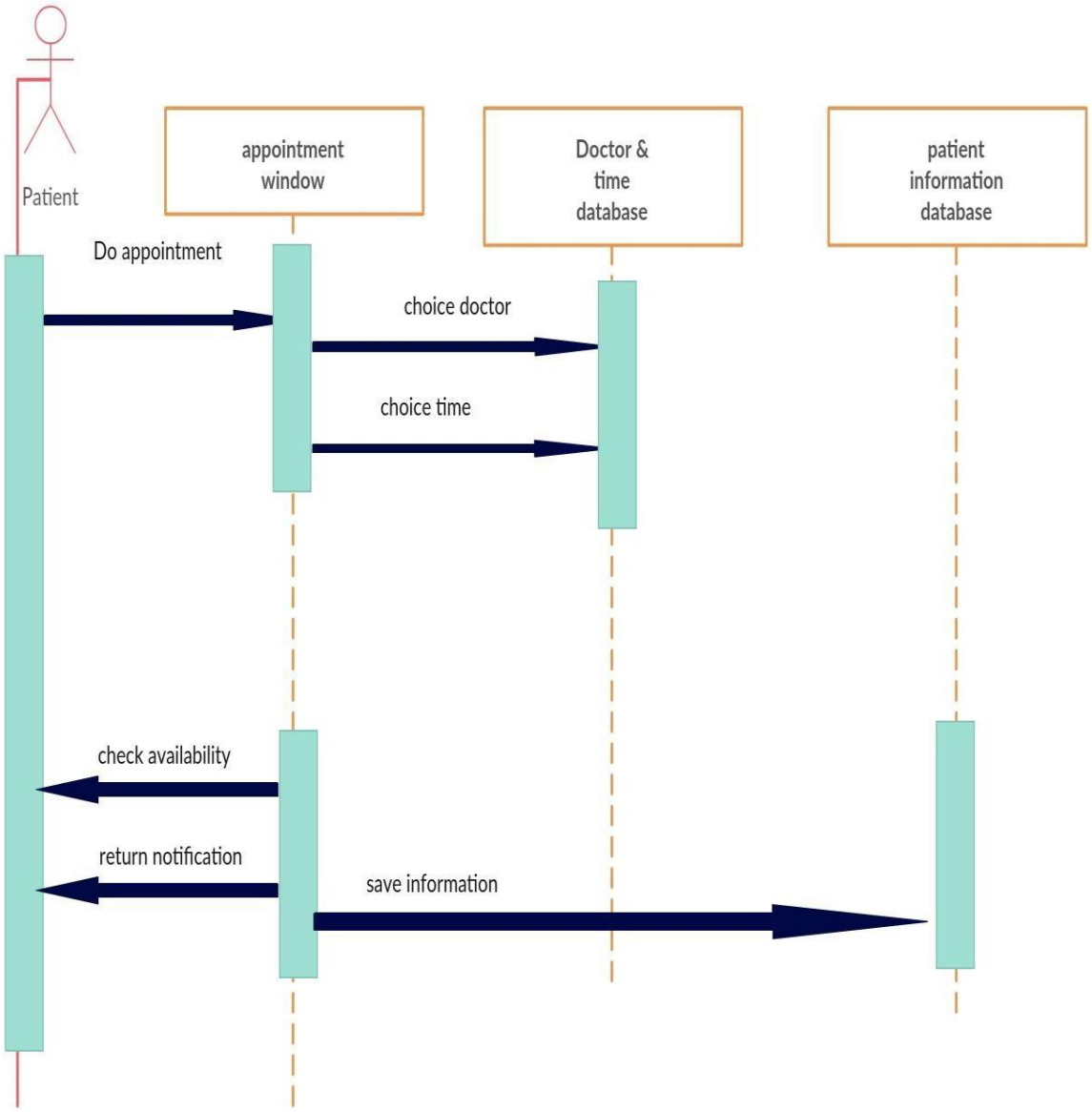
- **USE CASE DIAGRAM:**



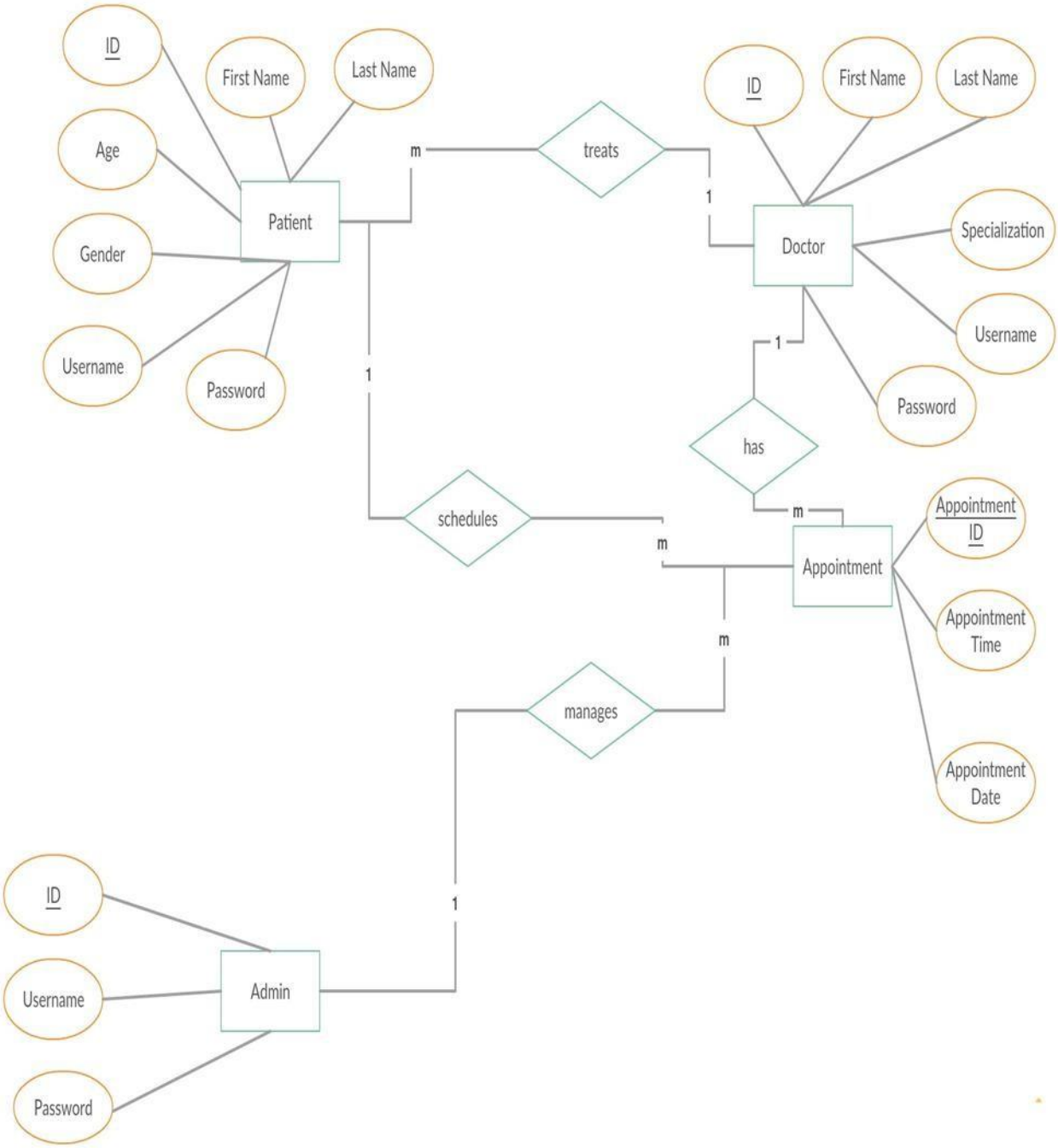
- **CLASS DIAGRAM:**



- **SEQUENCE DIAGRAM (2):**



- **ER DIAGRAM:**


ID
First Name
Last Name
Age
Gender
Username
Password
Patient
m
treats
1
ID
First Name
Last Name
Doctor
Specialization
Username
Password
1
has
m
1
schedules
m
Appointment
Appointment ID
Appointment Time
Appointment Date
m
manages
1
ID
Username
Password
Admin

• USE CASE DEFINITION AND DIAGRAM:

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams. Use case diagrams are used to gather the requirements of a system including internal and external aspects. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified. When the initial task is complete, use case diagrams are modelled to present the outside view.

• Purpose of Use Case Diagrams:

The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and state chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

When the initial task is complete, use case diagrams are modelled to present the outside view.

In brief, the purposes of use case diagrams can be said to be as follows –

- Used to gather the requirements of a system.
- Used to get an outside view of a system.
- Identify the external and internal factors influencing the system.

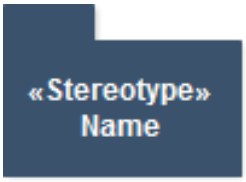
• **Use case Symbols:**

❖ Shows the interaction among the actors of system

Actor specifies a role played by a user or any other system that interacts with the subject.

❖ Use case is a list of steps, typically defining interactions between an actor and a system, to achieve a goal.

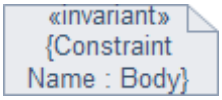
❖ Package is used to group elements, and to provide a namespace for the grouped elements.



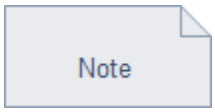
❖ Objects are model elements that represent instances of a class or of classes.

- Interfaces are model elements that define sets of operations that other model elements, such as classes, or components must implement.

- Constraint is an extension mechanism that enables you to refine the semantics of a UML model element.



- **Note:** contains comments or textual information.



CHAPTER:-VII

• MODULE DESCRIPTION:

The system is proposed to have the following modules along with the functional requirements.

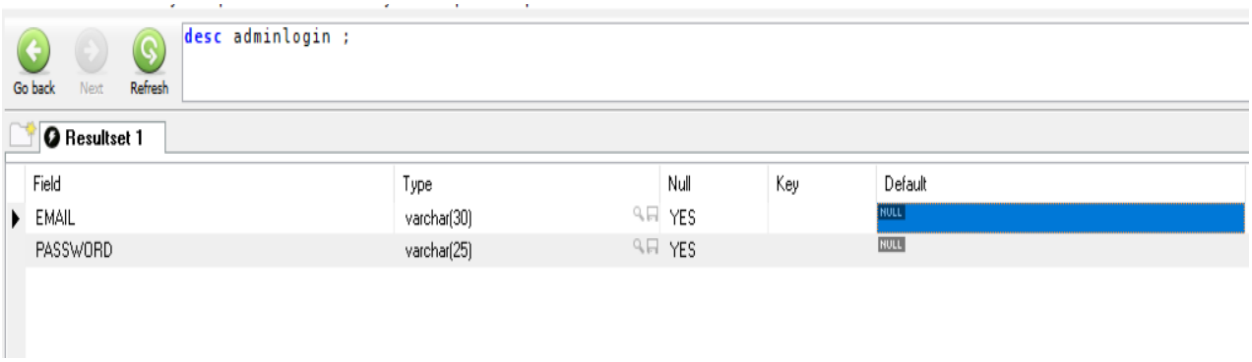
• ADMINISTRATOR MODULE:

Administrator should login with email id and password to maintain and update all details of system. Administrator can add or delete doctors in system. Administrator is responsible for verifying cusmoter credentials before adding them in system. Administrator is also responsible for maintaining all customer records. Administrator can view all customers and also feedback provided by the patients using the system.

CHAPTER:- VIII

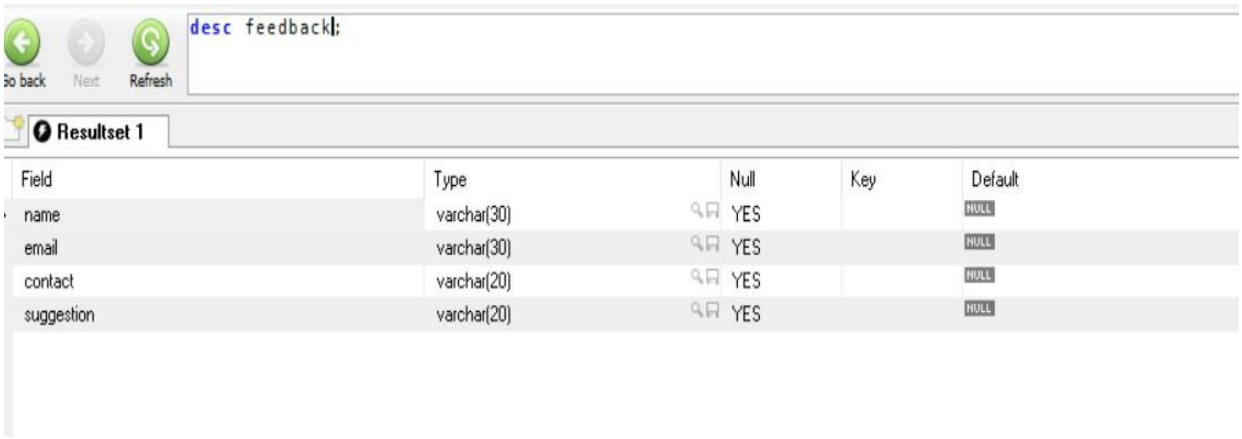
DATABASE STRUCTURE

• ADMIN LOGIN:

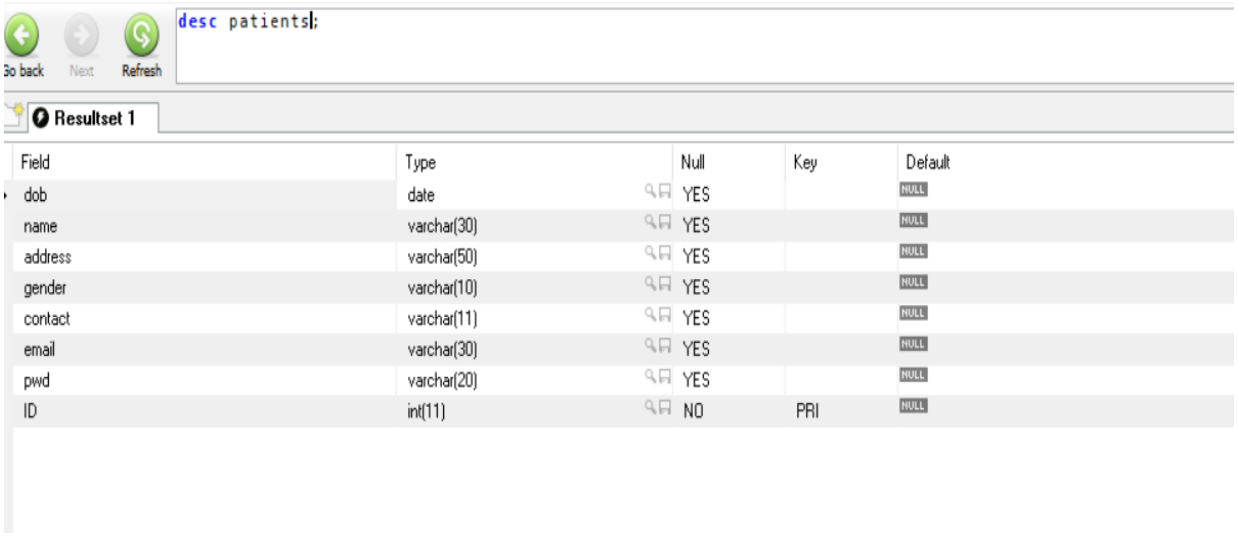


Field	Type	Null	Key	Default
EMAIL	varchar(30)	YES		NULL
PASSWORD	varchar(25)	YES		NULL

• **FEEDBACK:**


desc feedback;
Go back
Next
Refresh
Resultset 1
Field
Type
Null
Key
Default
name
varchar(30)
YES
NULL
email
varchar(30)
YES
NULL
contact
varchar(20)
YES
NULL
suggestion
varchar(20)
YES
NULL

CUSTOMERS:



Field	Type	Null	Key	Default
dob	date	YES		NULL
name	varchar(30)	YES		NULL
address	varchar(50)	YES		NULL
gender	varchar(10)	YES		NULL
contact	varchar(11)	YES		NULL
email	varchar(30)	YES		NULL
pwd	varchar(20)	YES		NULL
ID	int(11)	NO	PRI	NULL

CHAPTER:- IX

SYSTEM IMPLEMENTATION AND TESTING

• SYSTEM IMPLEMENTATION:

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation of change over methods.

Apart from planning major task of preparing the implementation are education and training of users. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. In network backup system no additional resources are needed. Implementation is the final and the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it is found to be working according to the specification. This method also offers the greatest security since the old system can take over the errors are found or inability to handle certain type of transactions while using the new system.

• SYSTEM TESTING:

As the part of system testing, we execute the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. The ultimate aim is quality assurance. Tests are carried out and the results are compared with the expected document. In the case of erroneous results, debugging is done. Using detailed testing strategies, a test plan is carried out on each module. The various tests performed are unit testing, integration testing and user acceptance testing

• **UNIT TESTING:**

. The software units in the system are modules and routines that are assembled and integrated to perform a specific function. As a part of unit testing, we executed the program for individual modules independently. This enables, to detect errors in coding and logic that are contained within each of the three modules. This testing includes entering data that is filling forms and ascertaining if the value matches to the type and entered into the database. The various controls are tested to ensure that each performs its action as required

• **INTEGRATION TESTING:**

Data can be lost across any interface, one module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integration testing is a systematic testing to discover errors associated within 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 the admin module, doctor module and patient module options are integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data.

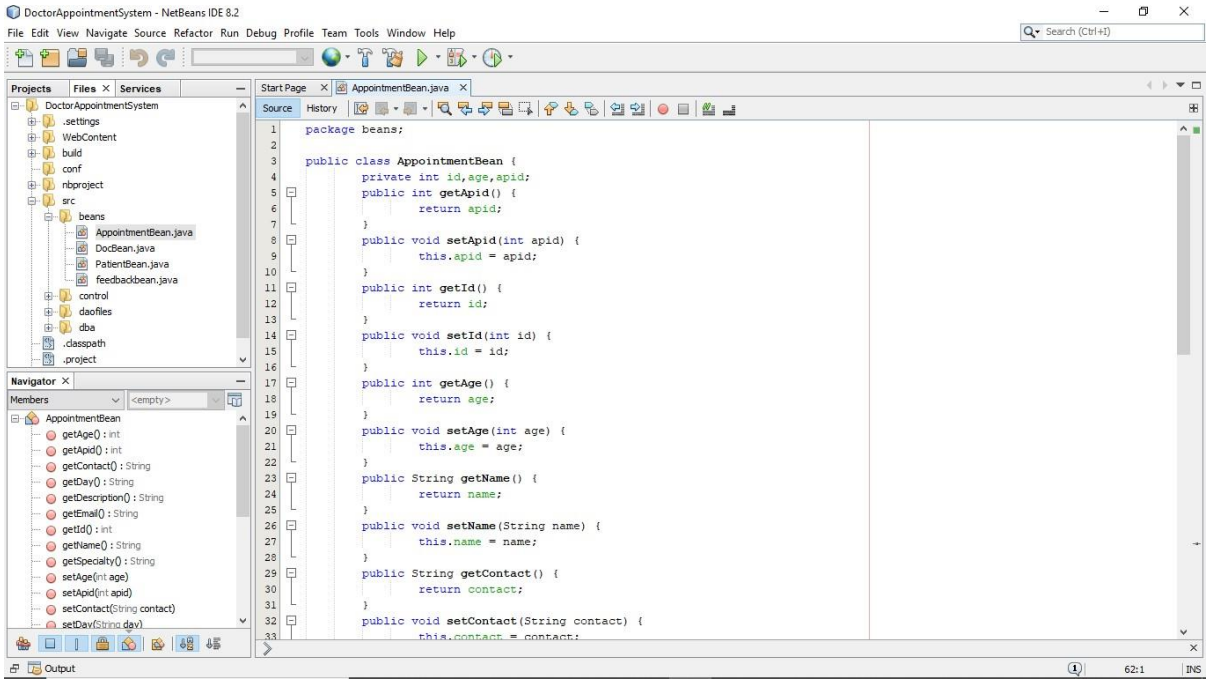
• **VALIDATION TESTING:**

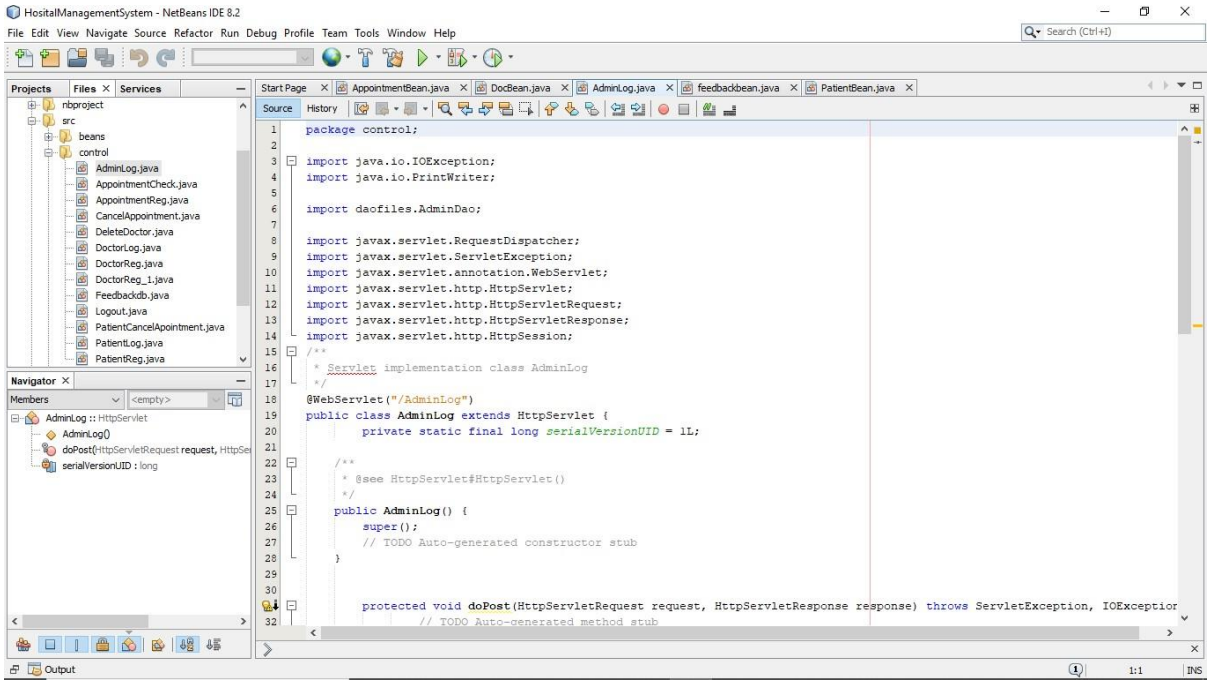
The process of evaluating software during the development process or at the end of the development process to determine whether it satisfies specified business requirements. Testing ensures that the product actually meets the client's needs. It can also be defined as to demonstrate that the product fulfills its intended use when deployed on appropriate environment.

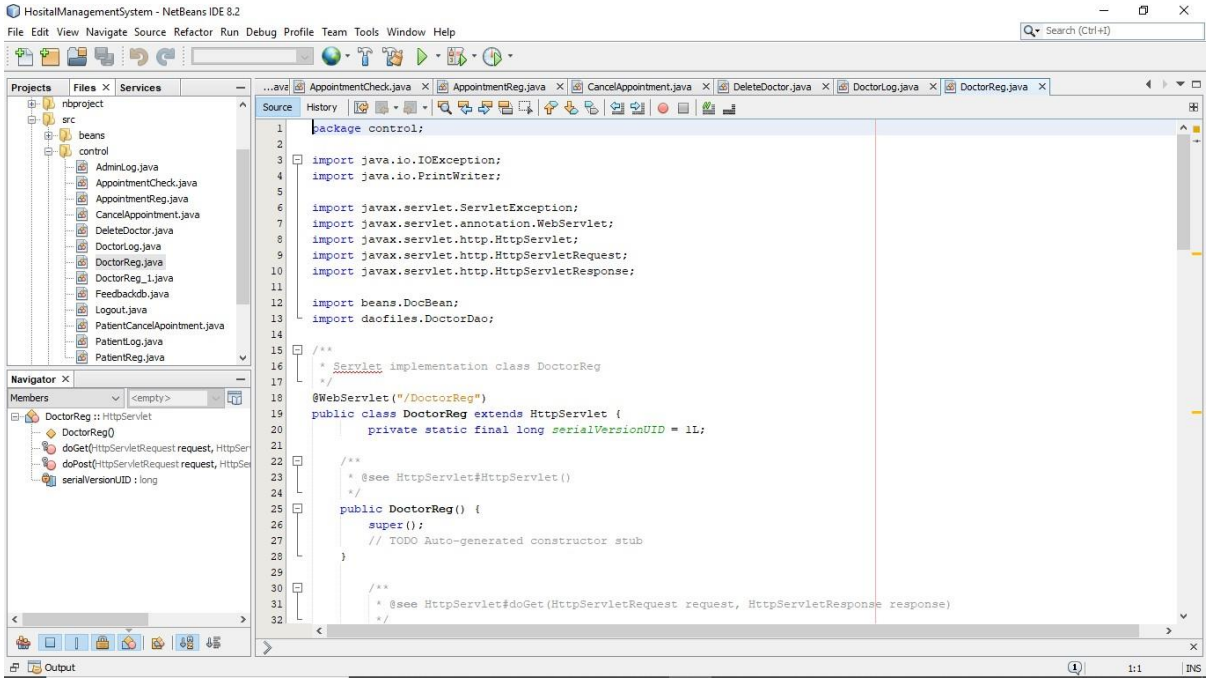
- Validation is the process of evaluating software at the end of the development process to determine whether software meets the customer expectations and requirements.
- Execution of code is coming under Validation.
- Validation activity is carried out just after the Verification.
- It determines whether the software is fit for use and satisfies the business need.
- Includes all the dynamic testing techniques.
- It is basically checking of developed program based on the requirement specifications documents & files.

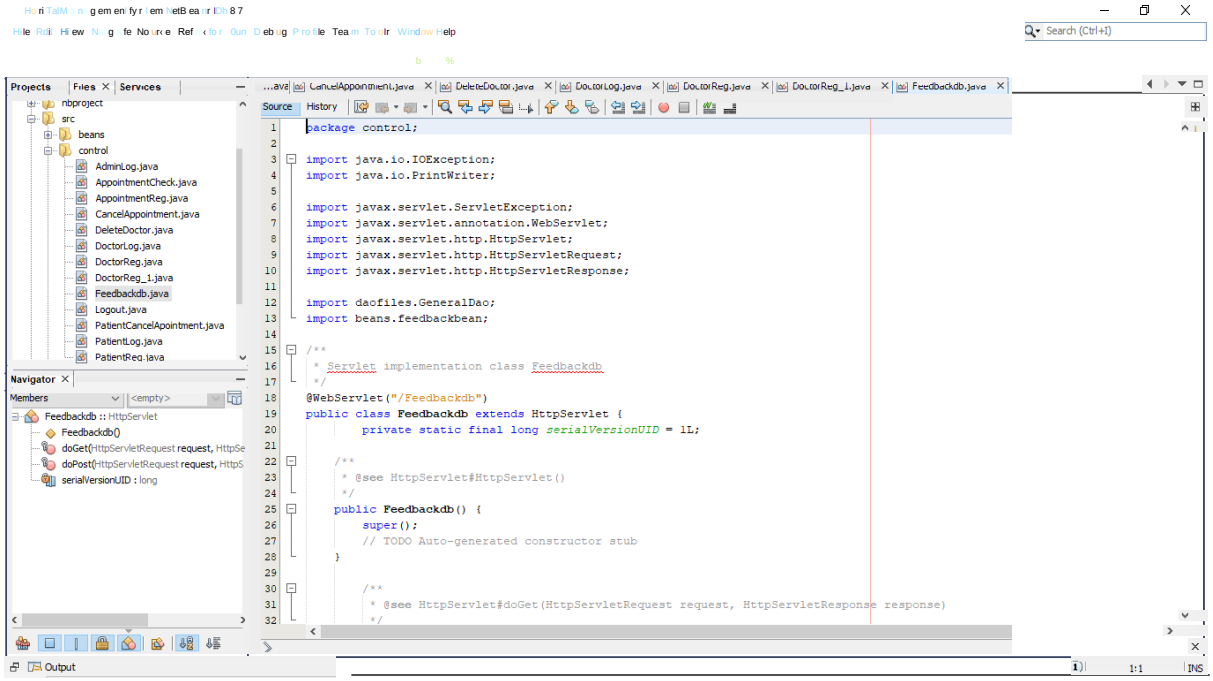
CHAPTER:- X

12. CODE SNIIPPETS




HositalManagementSystem - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
Projects
Files
Services
nbproject
src
beans
control
AdminLog.java
AppointmentCheck.java
AppointmentReg.java
CancelAppointment.java
DeleteDoctor.java
DoctorLog.java
DoctorReg.java
DoctorReg_1.java
Feedbackdb.java
Logout.java
PatientCancelApointment.java
PatientLog.java
PatientReg.java
Navigator
Members
<empty>
AdminLog :: HttpServlet
AdminLog()
doPost(HttpServletRequest request, HttpSe
serialVersionUID : long
Start Page
AppointmentBean.java
DocBean.java
AdminLog.java
feedbackbean.java
PatientBean.java
Source
History
1 package control;
2
3 import java.io.IOException;
4 import java.io.PrintWriter;
5
6 import daofiles.AdminDao;
7
8 import javax.servlet.RequestDispatcher;
9 import javax.servlet.ServletException;
10 import javax.servlet.annotation.WebServlet;
11 import javax.servlet.http.HttpServlet;
12 import javax.servlet.http.HttpServletRequest;
13 import javax.servlet.http.HttpServletResponse;
14 import javax.servlet.http.HttpSession;
15 /**
16 * Servlet implementation class AdminLog
17 */
18 @WebServlet("/AdminLog")
19 public class AdminLog extends HttpServlet {
20 private static final long serialVersionUID = 1L;
21
22 /**
23 * @see HttpServlet#HttpServlet()
24 */
25 public AdminLog() {
26 super();
27 // TODO Auto-generated constructor stub
28 }
29
30
32 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOExceptior
// TODO Auto-generated method stub
Output
1:1
INS


HositalManagementSystem - NetBeans IDE 8.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
Projects
Files
Services
nbproject
src
beans
control
AdminLog.java
AppointmentCheck.java
AppointmentReg.java
CancelAppointment.java
DeleteDoctor.java
DoctorLog.java
DoctorReg.java
DoctorReg_1.java
Feedbackdb.java
Logout.java
PatientCancelApointment.java
PatientLog.java
PatientReg.java
Navigator
Members
<empty>
DoctorReg :: HttpServlet
DoctorReg()
doGet(HttpServletRequest request, HttpSer
doPost(HttpServletRequest request, HttpSe
serialVersionUID : long
...ava
AppointmentCheck.java
AppointmentReg.java
CancelAppointment.java
DeleteDoctor.java
DoctorLog.java
DoctorReg.java
Source
History
1 package control;
2
3 import java.io.IOException;
4 import java.io.PrintWriter;
5
6 import javax.servlet.ServletException;
7 import javax.servlet.annotation.WebServlet;
8 import javax.servlet.http.HttpServlet;
9 import javax.servlet.http.HttpServletRequest;
10 import javax.servlet.http.HttpServletResponse;
11
12 import beans.DocBean;
13 import daofiles.DoctorDao;
14
15 /**
16 * Servlet implementation class DoctorReg
17 */
18 @WebServlet("/DoctorReg")
19 public class DoctorReg extends HttpServlet {
20 private static final long serialVersionUID = 1L;
21
22 /**
23 * @see HttpServlet#HttpServlet()
24 */
25 public DoctorReg() {
26 super();
27 // TODO Auto-generated constructor stub
28 }
29
30 /**
31 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
32 */
Output
1:1
INS


Search (Ctrl+I)
Projects
Files
Services
nbproject
src
beans
control
AdminLog.java
AppointmentCheck.java
AppointmentReg.java
CancelAppointment.java
DeleteDoctor.java
DoctorLog.java
DoctorReg.java
DoctorReg_1.java
Feedbackdb.java
Logout.java
PatientCancelApointment.java
PatientLog.java
PatientReg.java
Navigator
Members
<empty>
Feedbackdb :: HttpServlet
Feedbackdb()
doGet(HttpServletRequest request, HttpSe
doPost(HttpServletRequest request, HttpS
serialVersionUID : long
CancelAppointment.java
DeleteDoctor.java
DoctorLog.java
DoctorReg.java
DoctorReg_1.java
Feedbackdb.java
Source
History
1 package control;
2
3 import java.io.IOException;
4 import java.io.PrintWriter;
5
6 import javax.servlet.ServletException;
7 import javax.servlet.annotation.WebServlet;
8 import javax.servlet.http.HttpServlet;
9 import javax.servlet.http.HttpServletRequest;
10 import javax.servlet.http.HttpServletResponse;
11
12 import daofiles.GeneralDao;
13 import beans.feedbackbean;
14
15 /**
16 * Servlet implementation class Feedbackdb
17 */
18 @WebServlet("/Feedbackdb")
19 public class Feedbackdb extends HttpServlet {
20 private static final long serialVersionUID = 1L;
21
22 /**
23 * @see HttpServlet#HttpServlet()
24 */
25 public Feedbackdb() {
26 super();
27 // TODO Auto-generated constructor stub
28 }
29
30 /**
31 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
32 */
Output
1:1
INS