

Ex. No:**ONLINE COURSE RESERVATION SYSTEM****Date:****AIM:**

To create a system through which students can register to the courses desired by them.

(I) PROBLEM STATEMENT

The system is built to be used by students and managed by an administrator. The student and employee have to login to the system before any processing can be done. The student can see the courses available to him/her and register to the course he/she wants. The administrator can maintain the course details and view all the students who have registered to any course.

(II) SOFTWARE REQUIREMENT SPECIFICATION**1.0. INTRODUCTION**

Course Reservation System is an interface between the Student and the Registrar responsible for the issue of Course. It aims at improving the efficiency in the issue of Course and reduces the complexities involved in it to the maximum possible extent.

1.1 PURPOSE

If the entire process of 'Issue of Course' is done in a manual manner then it would takes several months for the course to reach the applicant. Considering the fact that the number of applicants for course is increasing every year, an Automated System becomes essential to meet the demand. So this system uses several programming and database techniques to elucidate the work involved in this process.

1.2 SCOPE

- The System provides an online interface to the user where they can fill in their personal details and submit the necessary documents (may be by scanning).
- The Registrar concerned with the issue of course can use this system to reduce his workload and process the application in a speedy manner.
- Provide a communication platform between the Student and the Registrar.

1.3 DEFINITIONS, ACRONYMS AND THE ABBREVIATIONS

- **Registrar**
Refers to the super user with the privilege to manage the entire system.
- **Applicant**
One who wishes to register the Course
- **OCRS**
Refers to online Course Reservation System.
- **HTML**
Markup Language used for creating web pages.
- **J2EE**
Java 2 Enterprise Edition is a programming platform java platform for developing and running distributed java applications.
- **HTTP**
Hyper Text Transfer Protocol.
- **TCP/IP**
Transmission Control Protocol/Internet Protocol is the communication protocol used to connect hosts on the Internet.

1.4 REFERENCES

IEEE Software Requirement Specification format.

1.5 TECHNOLOGIES TO BE USED

- HTML
- JSP
- Javascript
- Java

1.6 TOOLS TO BE USED

- Eclipse IDE (Integrated Development Environment)
- Rational Rose tool (for developing UML Patterns)

1.7 OVERVIEW

SRS includes two sections overall description and specific requirements

Overall Description will describe major role of the system components and inter-connections.

Specific Requirements will describe roles & functions of the actors.

2.0 OVERALL DESCRIPTION

2.1 PRODUCT PERSPECTIVE

The OCRS acts as an interface between the 'Student' and the 'Registrar'. This system tries to make the interface as simple as possible and at the same time not risking the security of data stored in. This minimizes the time duration in which the user receives the course.

2.2 SOFTWARE INTERFACE

- **Front End Client** - The Student and Registrar online interface is built using JSP and HTML. The Administrators's local interface is built using Java.
- **Web Server** – Tomcat Apache application server (Oracle Corporation).
- **Back End** – Oracle 11g database.

2.3 HARDWARE INTERFACE

The server is directly connected to the client systems. The client systems have access to the database in the server.

2.4 SYSTEM FUNCTIONS

- Secure Reservation of information by the Students.
- SMS and Mail updates to the students by the Registrar
- Registrar can generate reports from the information and is the only authorized personnel to add the eligible application information to the database.

2.5 USER CHARACTERISTICS

- **Applicant** - They are the person who desires to obtain the course and submit the information to the database.
- **Administrator** - He has the certain privileges to add the course status and to approve the issue of course. He may contain a group of persons under him to verify the documents and give suggestion whether or not to approve the dispatch of course.

2.6 CONSTRAINTS

- The applicants require a computer to submit their information.
- Although the security is given high importance, there is always a chance of intrusion in the web world which requires constant monitoring.
- The user has to be careful while submitting the information. Much care is required.

2.7 ASSUMPTIONS AND DEPENDENCIES

- The Applicants and Administrator must have basic knowledge of computers and English Language.
- The applicants may be required to scan the documents and send

(III)USE-CASE DIAGRAM:

The course registration system has the following use-cases

1. Login
2. View course details
3. Reserve for course
4. Pay fee
5. Check status

ACTORS INVOLVED:

1. Student
2. Registrar

USE-CASE NAME: LOGIN

The user enters the username and password and chooses if the user is student or Registrar. If entered details are valid, the user's account becomes available. If it is invalid, an appropriate message is displayed to the user.

USE-CASE NAME: VIEW COURSE DETAILS

In this use case, a student can search all the courses available to him and choose the best course he wants. The student can view the course duration, faculty and department of the courses he may choose.

USE-CASE NAME: RESERVE FOR COURSE

When a student has successfully chosen a course, he can register to that course. Upon registration, the student's details are stored in the database.

USE-CASE NAME: PAY FEE

After registration to any course, the student may see the details of his current course. He may wish to know details about fees and other information.

USE-CASE NAME: CHECK STATUS

The student tries to check the status in which category applied. The system displays the status information to the student.

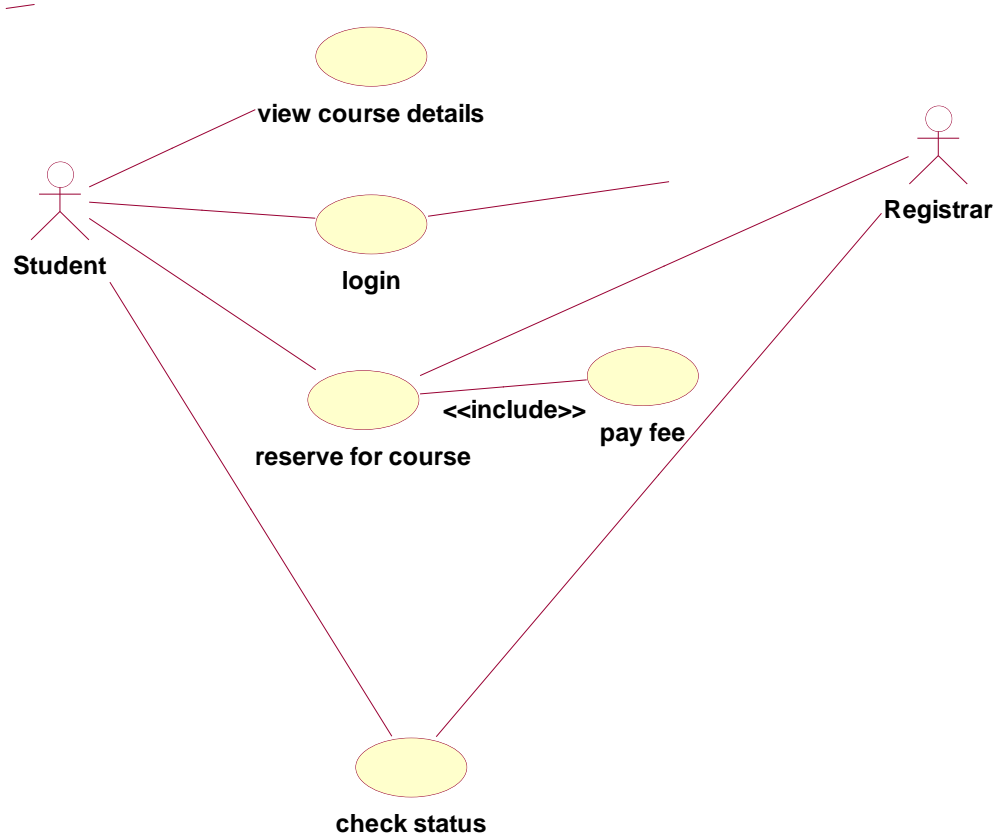
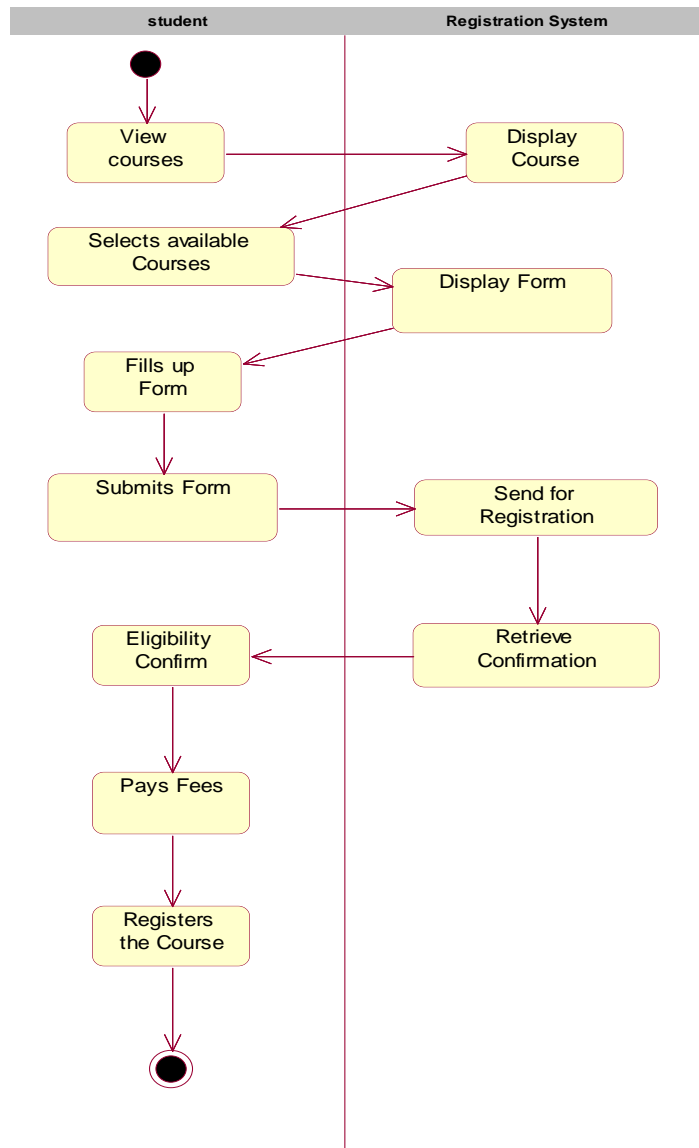


Fig.3.USE-CASE DIAGRAM

(IV) ACTIVITY DIAGRAM:**Fig.4. ACTIVITY DIAGRAM**

(V) CLASS DIAGRAM:

The class diagram is a graphical representation of all the classes used in the system and their operations, attributes and relationships.

The course registration system makes use of the following classes:

1. Student
2. Course Catalog
3. Reserve Course

1) STUDENT:

It consists of the details of all the students present in the database. The attributes present in this class are student id, student name, student qualification, student address1, student address2, student address3, student mobile no, student emailed,, student dob, student sex. The object of this class is created as soon as the student registers to a course. The operations available to this class are add details (), modify details (), del details (), reserve course().

2) COURSE CATALOG:

The course catalog class consist of course id, course name, course duration course fee, course eligibility, total no of seat, course avail seat. The operations are add course(), update course(), del course().

3) RESERVE COURSE:

The reserve catalog class consists of student id, course id, date, amt paid, reg id, DD no. the operation are get course details(), check eligibility(), confirm registration().

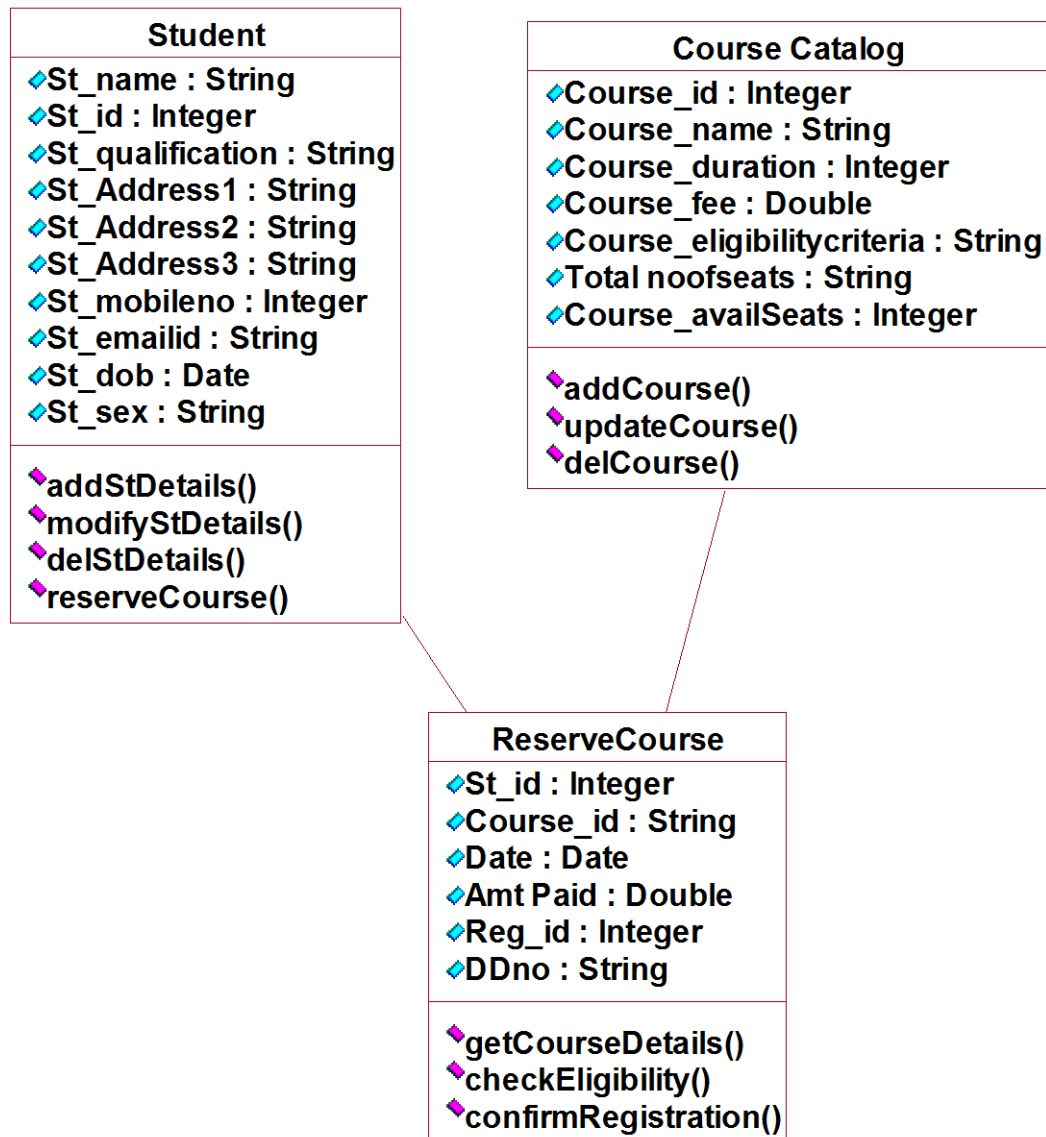


Fig.5. CLASS DIAGRAM

(VI) INTERACTION DIAGRAM:

- A sequence diagram represents the sequence and interactions of a given USE-CASE or scenario. Sequence diagrams can capture most of the information about the system. Most object to object interactions and operations are considered events and events include signals, inputs, decisions, interrupts, transitions and actions to or from users or external devices.
- An event also is considered to be any action by an object that sends information. The event line represents a message sent from one object to another, in which the “from” object is requesting an operation be performed by the “to” object. The “to” object performs the operation using a method that the class contains.
- It is also represented by the order in which things occur and how the objects in the system send message to one another.
- The sequence diagram for each USE-CASE that exists when a user administrator, check status and new registration about course registration system are given.
- Users have to first login to the system before performing any operation. The user has to provide the necessary details to the system for login.

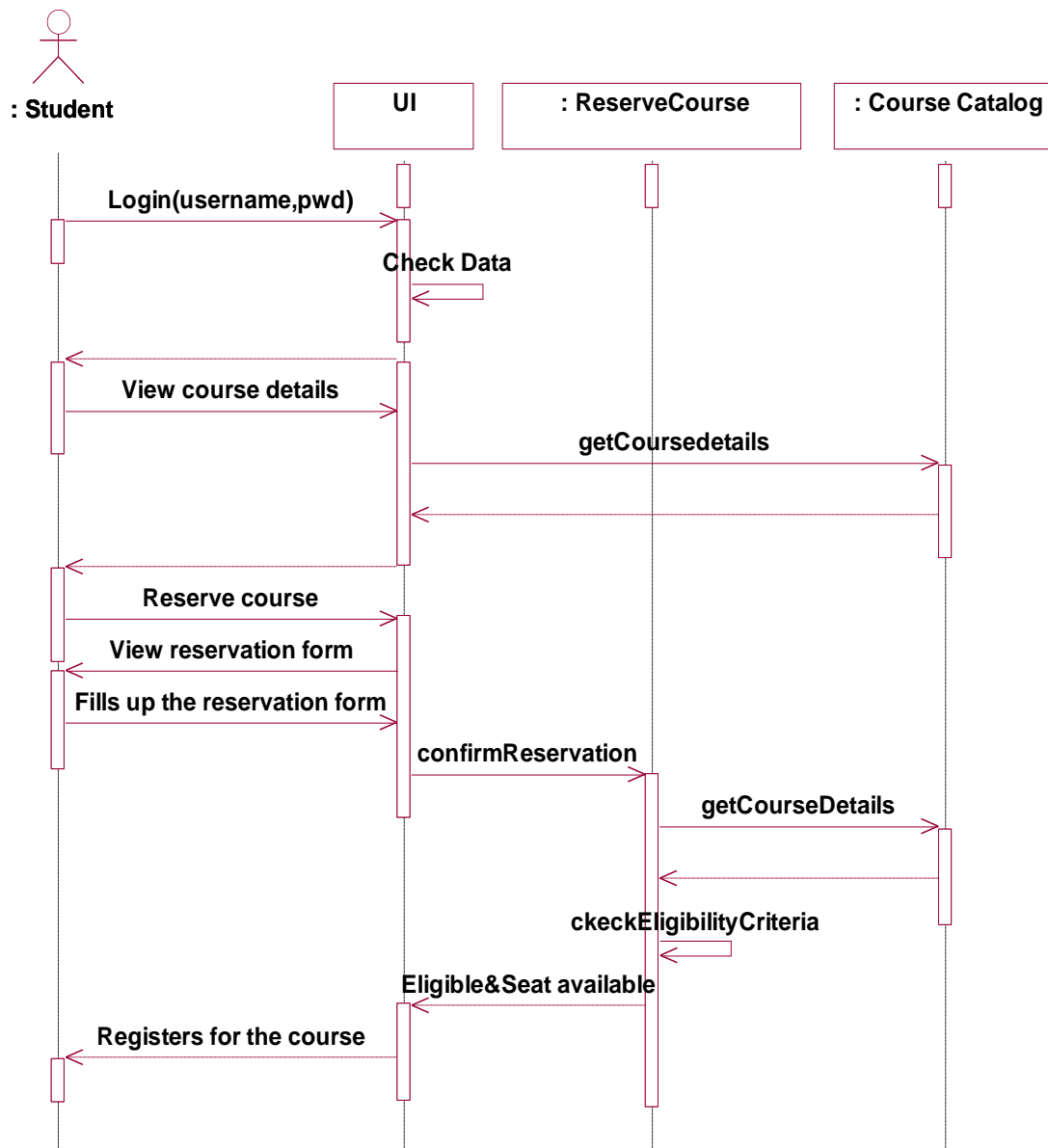


Fig.6.1.SEQUENCE DIAGRAM

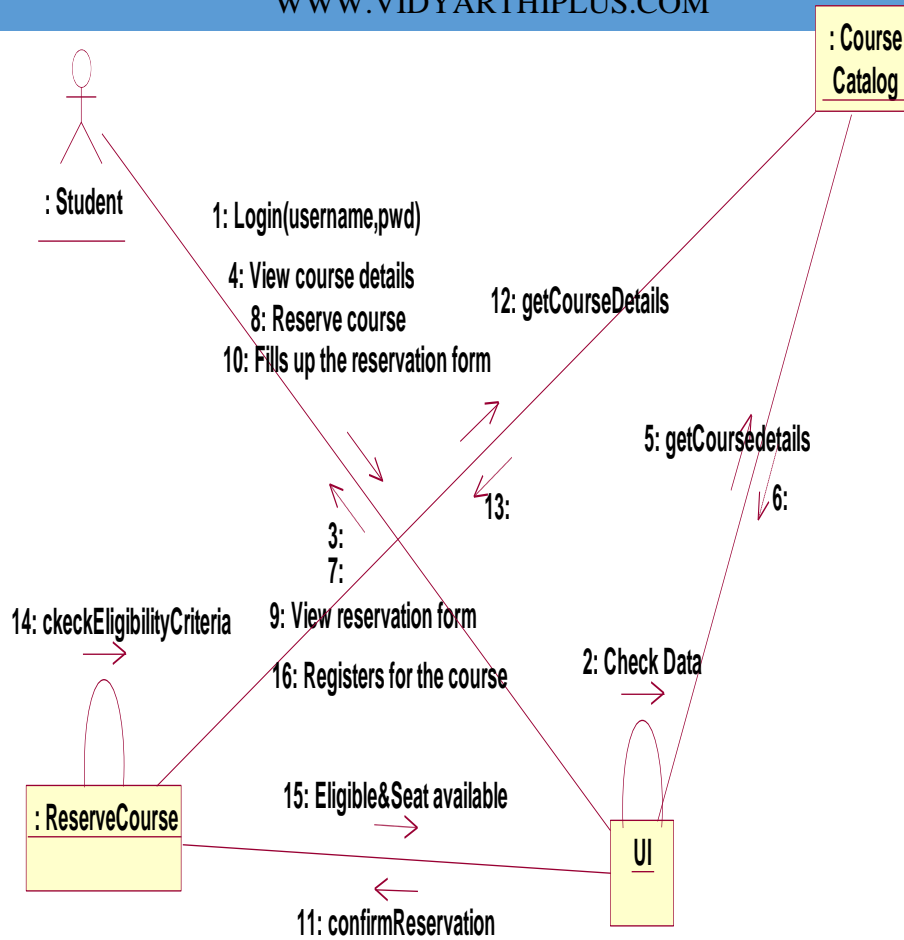


Fig.6.2.COLLABORATION DIAGRAM

- After login, the student has to register to a course of his choice. The student can view all the courses available to him and register to a course suitable to him. The student may view the course details before registration.
- A student may wish to view course details before registration. For this, the student has to first login and select the course details he wishes to see.

(VII) STATE CHART DIAGRAM:

Every object undergoes through some state and on receiving some event the state gets changed. This transition of the state can be represented by the state transition diagram.

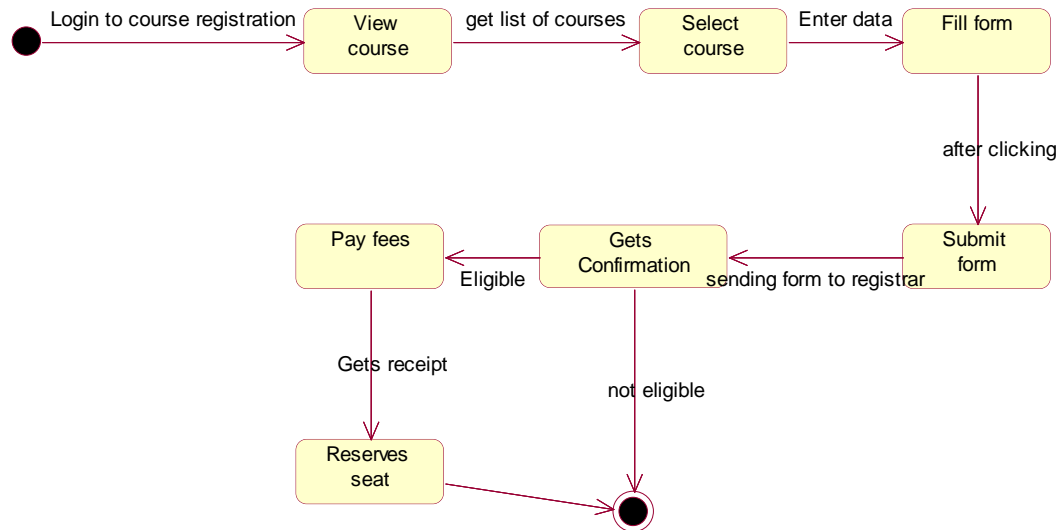


Fig.7. STATE CHART DIAGRAM

(VIII) DEPLOYMENT DIAGRAM AND COMPONENT DIAGRAM

Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed.

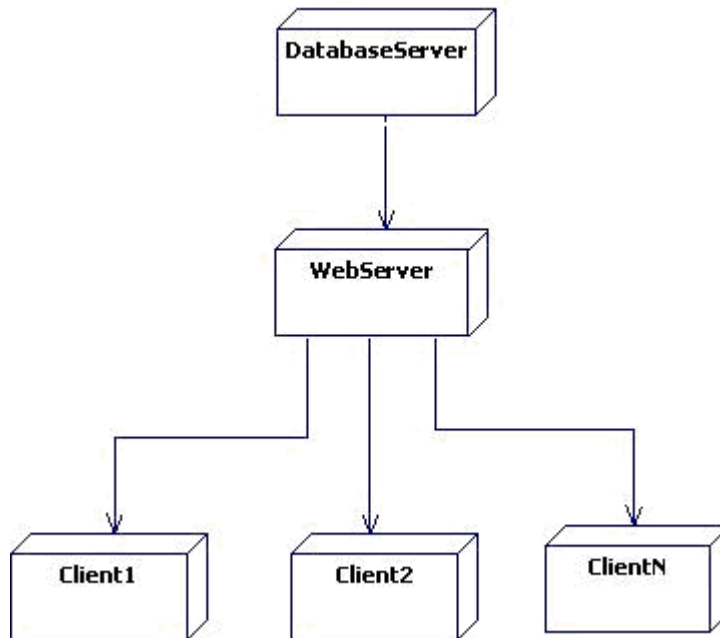


Fig.8.1.DEPLOYMENT DIAGRAM

COMPONENT DIAGRAM:

Component diagrams are used to visualize the organization and relationships among components in a system.

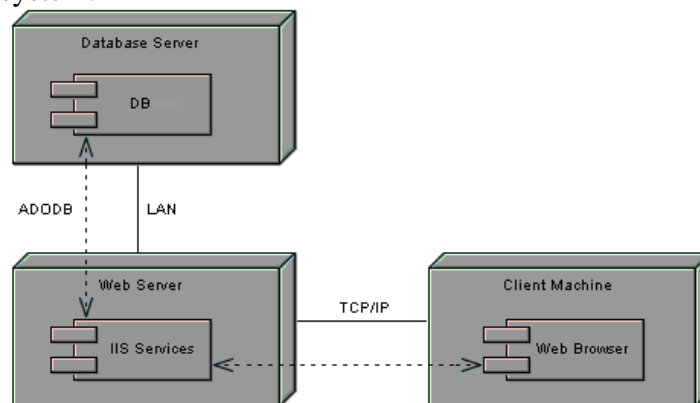


Fig.8.2.COMPONENT DIAGRAM

(IX) IMPLEMENTATION OF DOMAIN OBJECTS LAYER AND TECHNICAL SERVICE LAYER

//Source file: coursecatalog.java

```

public class coursecatalog
{
    private integer courseid;
    private integer coursename;
    private integer courseduration;
    private double coursefees;
    private string courseeligibilitycritiria;
    private string totalnoofseats;
    private integer courseavailable;
    public reservecourse theReservecourse;

    /**
    @roseuid 512350660128
    */
    public coursecatalog()
    {

    }

    /**
    @roseuid 51234FBD005D
    */
    public void addcourse()
    {

    }

    /**
    @roseuid 51234FC80138
    */
    public void updatecourse()
    {

    }

    /**
    @roseuid 51234FD20251
    */
    public void deletecourse()
    {

    }
}

```

//Source file: reservecourse.java

```

public class reservecourse
{
    private integer studentid;

```

```
private integer courseid;
private date date;
private double amountpaid;
private integer registerid;
private string DDno;

/**
@roseuid 512350660167
*/
public reservecourse()
{

}

/**
@roseuid 51234EE5007D
*/
public void getcoursedetails()
{

}

/**
@roseuid 51234EF3003E
*/
public void checkeligibility()
{

}

/**
@roseuid 51234F0102AF
*/
public void confirmreg()
{

}
}
```

(X) IMPLEMENTATION OF USER INTERFACE LAYER

Home Page



Login Page

MEC INFO TECH

LOGIN HERE!!!	
User Name	<input type="text"/>
Password	<input type="password"/>
Re-type Password	<input type="password"/>
<input type="button" value="GET REGISTER"/>	

Course Catalog

MEC INFO TECH

Courses Offered

JAVA

Course Name : Java Full Pack

Fee: 12,000

Duration : 3 Months

[Register Now!!!](#)

.NET FRAMEWORK

Course Name: VS .NET

Fee: 12,000

Duration: 3 Months

[Register Now!!!](#)

ORACLE 11g

Course Name: Oracle

Fee: 10,000

Duration: 3 Months

[Register Now!!!](#)

LOGIN HERE!!!

User Name	<input type="text" value="malathi"/>
Password	<input type="password" value="*****"/>
GET REGISTER	
New User?	

Registration Form

MEC INFO TECH

REGISTRATION FORM

Name	<input type="text" value="Mani"/>
Mobile Number	<input type="text" value="989856230"/>
Qualification	<input type="text" value="UG"/>
Course Name	<input type="text" value=".NET"/>
Course Fee	<input type="text" value="12000"/>
Submit	Reset

ThanU Page

MEC INFO TECH

Congratulation!!!!

You have successfully registered

Your registration id is

ID:3

RESULT:

Thus the mini project for Course Reservation system has been successfully executed and codes are generated.