Ex no:1 **PASSPORT AUTOMATION SYSTEM**

AIM:

To create an automated system to perform the Passport Process.

(I) PROBLEM STATEMENT:

Passport Automation System is used in the effective dispatch of passport to all of the applicants. This system adopts a comprehensive approach to minimize the manual work and schedule resources, time in a cogent manner. The core of the system is to get the online registration form (with details such as name, address etc.,) filled by the applicant whose testament is verified for its genuineness by the Passport Automation System with respect to the already existing information in the database.

(II)SOFTWARE REQUIREMENT SPECIFICATION:

2.1SOFTWARE INTERFACE

- **Front End Client** - The applicant and Administrator online interface is built using

  JSP and HTML. The Administrators's local interface is built using Java.

- **Web Server** - Glassfish application server(Oracle Corporation).
- **Back End** - Oracle database.

2.2HARDWARE INTERFACE

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

II) USECASE DIAGRAM :

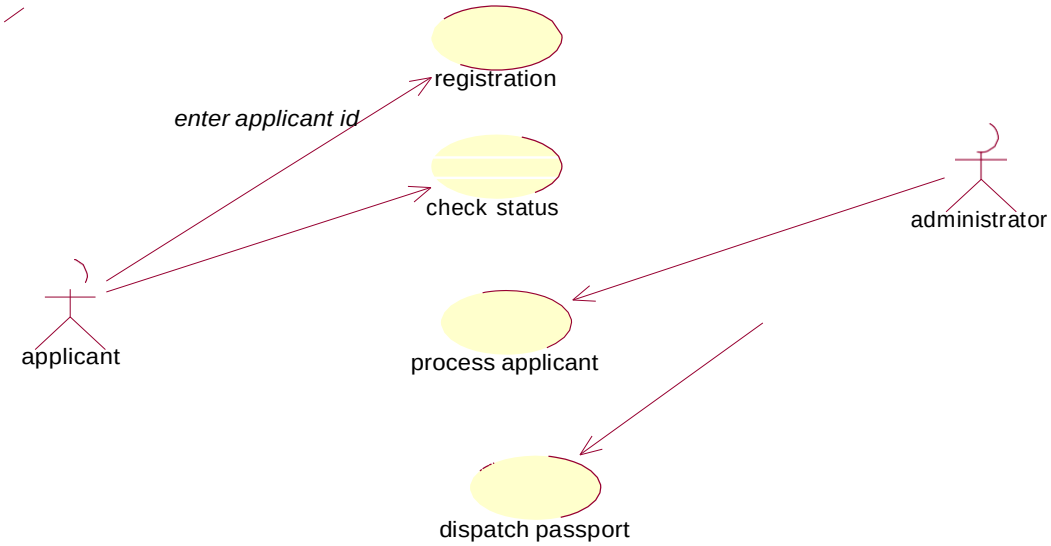


Fig.3. USECASE DIAGRAM FOR PASSPORT AUTOMATION SYSTEM

(IV) ACTIVITY DIAGRAM:

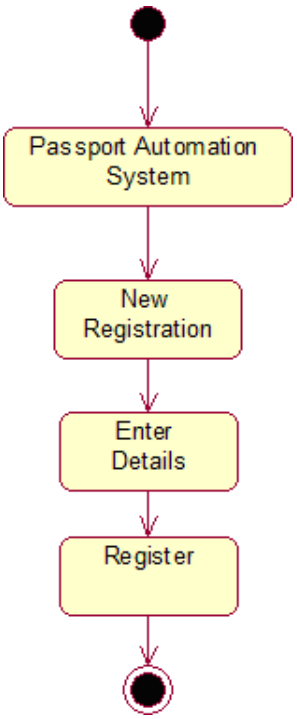


Fig.4.1. ACTIVITY DIAGRAM FOR REGISTER

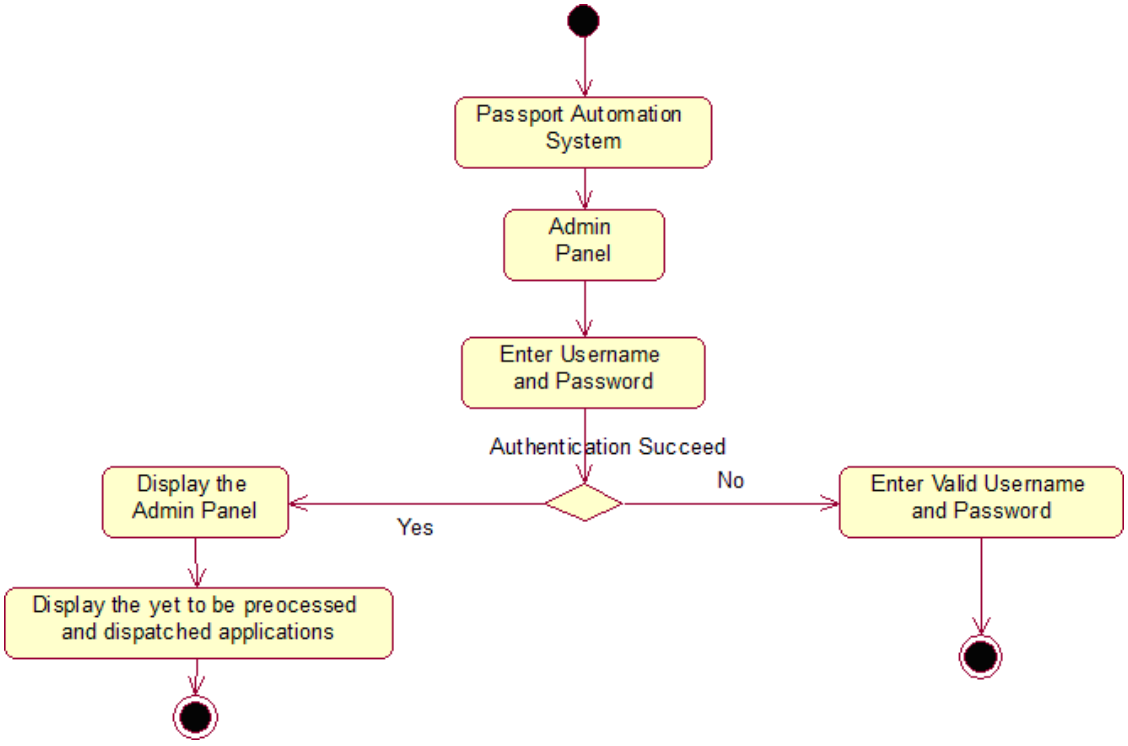


Fig.4.2. ACTIVITY DIAGRAM FOR ADMINISTRATION

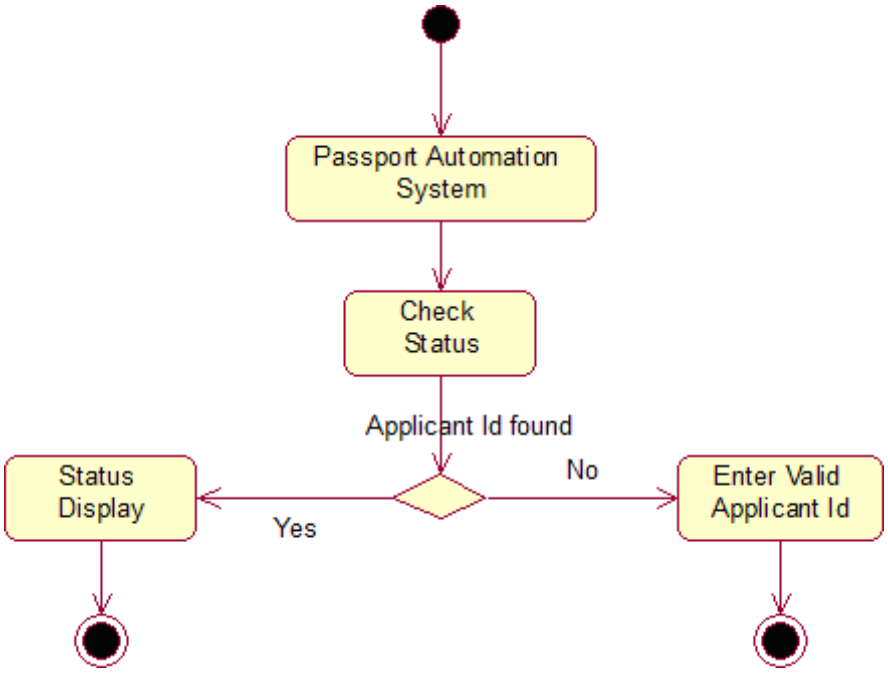


Fig.4.3. ACTIVITY DIAGRAM FOR CHECKING STATUS

(V) CLASS DIAGRAM:

The class diagram, also referred to as object modeling is the main static analysis diagram. The main task of object modeling is to graphically show what each object will do in the problem domain. The problem domain describes the structure and the relationships among objects.

The Passport Automation system class diagram consists of four classesPassport Automation System

1. New registration
2. Gender
3. Application Status
4. Admin authentication
5. Admin Panel

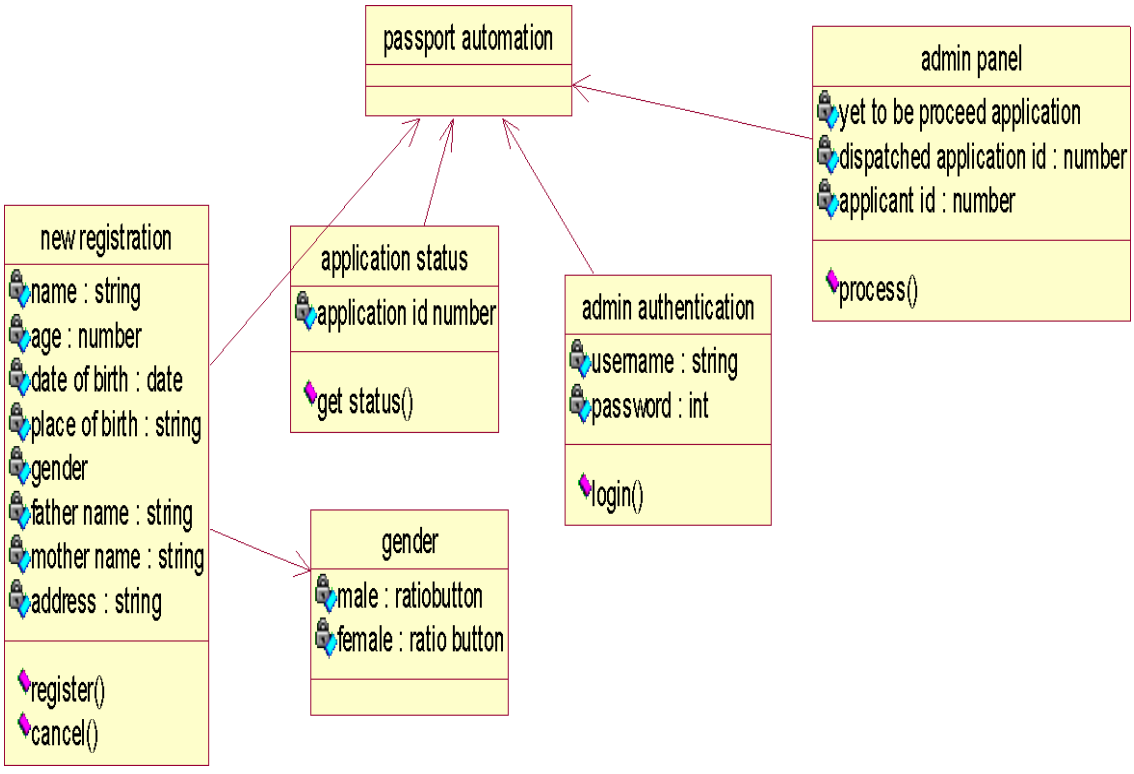


Fig.5. CLASS DIAGRAM FOR PASSPORT AUTOMATION SYSTEM

(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 “form” 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 passport automation system are given.

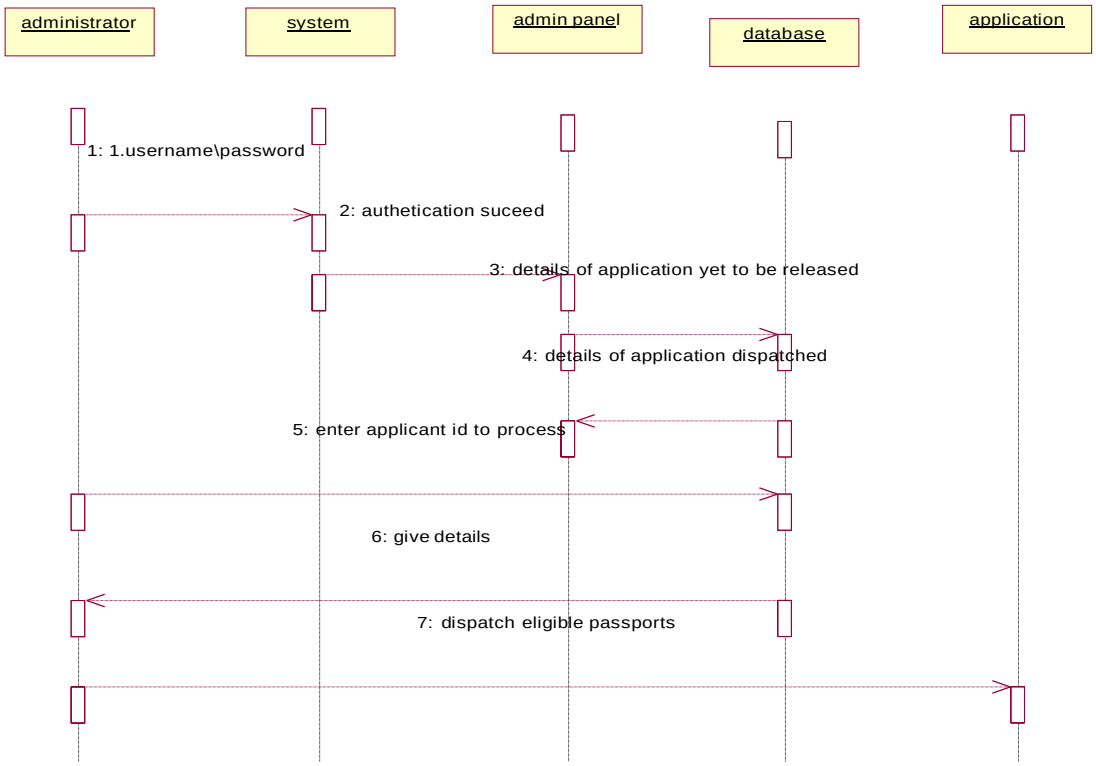


Fig.6.1.SEQUENCE DIAGRAM FOR ADMINISTRATOR

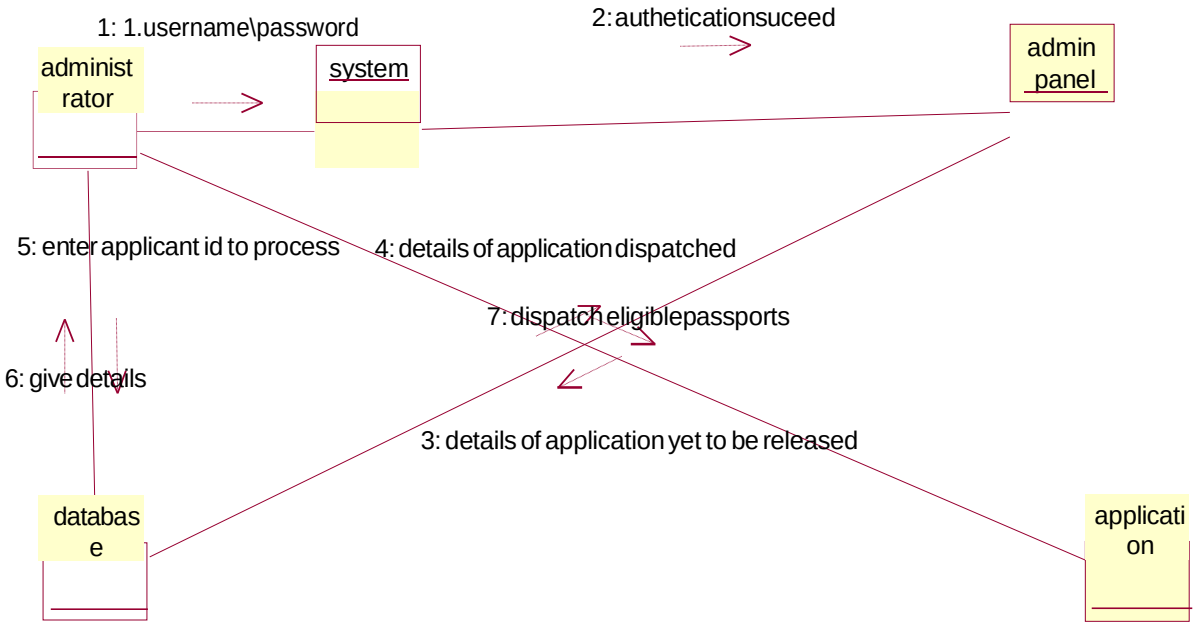


Fig.6.2.COLLABORATION DIAGRAM FOR ADMINISTRATOR

The diagrams show the process done by the administrator to the Passport Automation system. The applicant has to enter his details. The

details entered are verified by the administrator and the applicant is approved if the details match then the passport is dispatch, otherwise an appropriate error message is displayed.

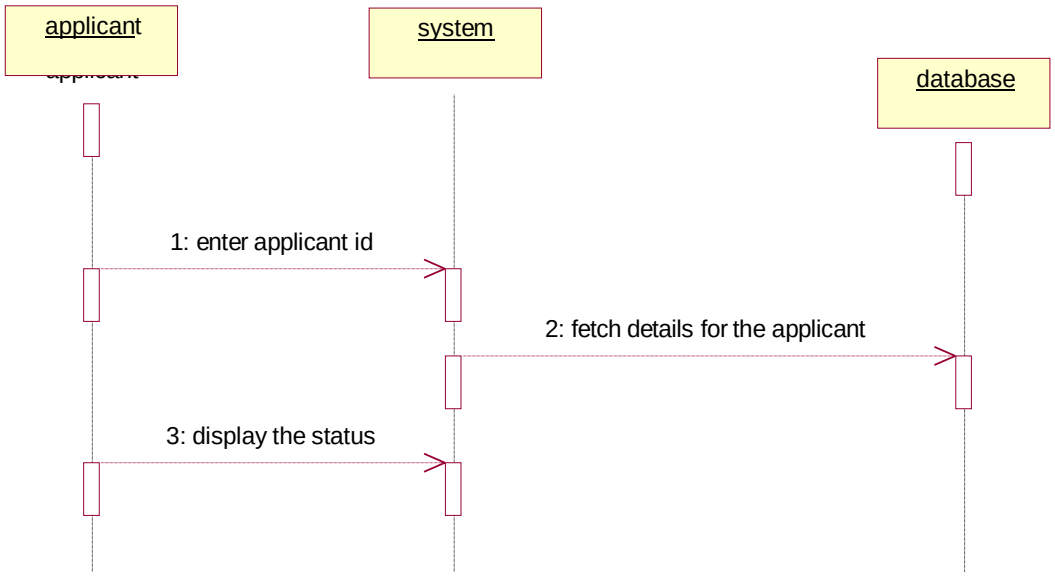


Fig.6.3.SEQUENCE DIAGRAM FOR CHECKING STATUS

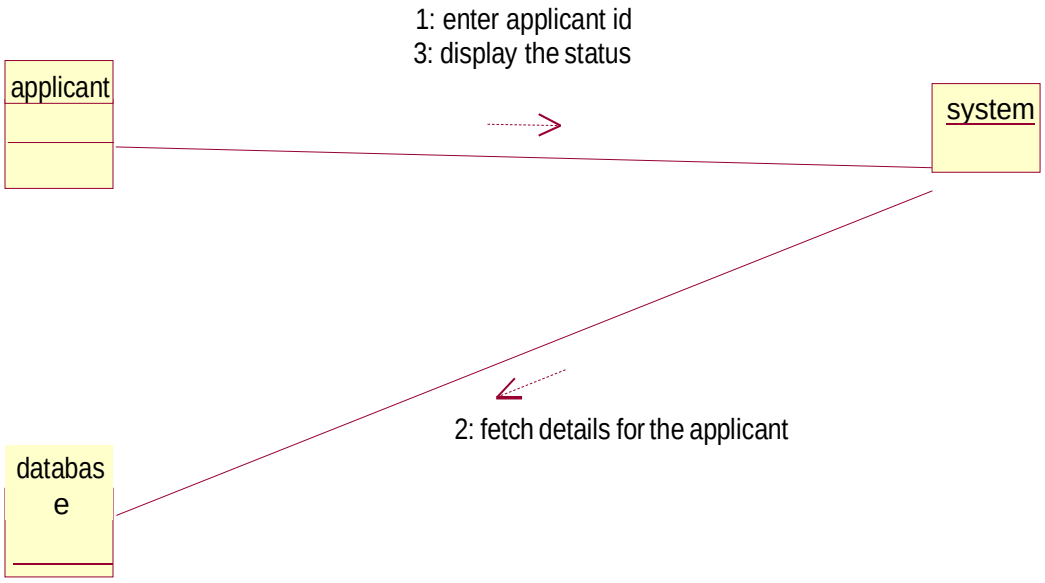


Fig.6.4.COLLABORATION DIAGRAM FOR CHECKING STATUS

The diagrams show the applicant enters his id and the system fetch the details from the database and display the status.

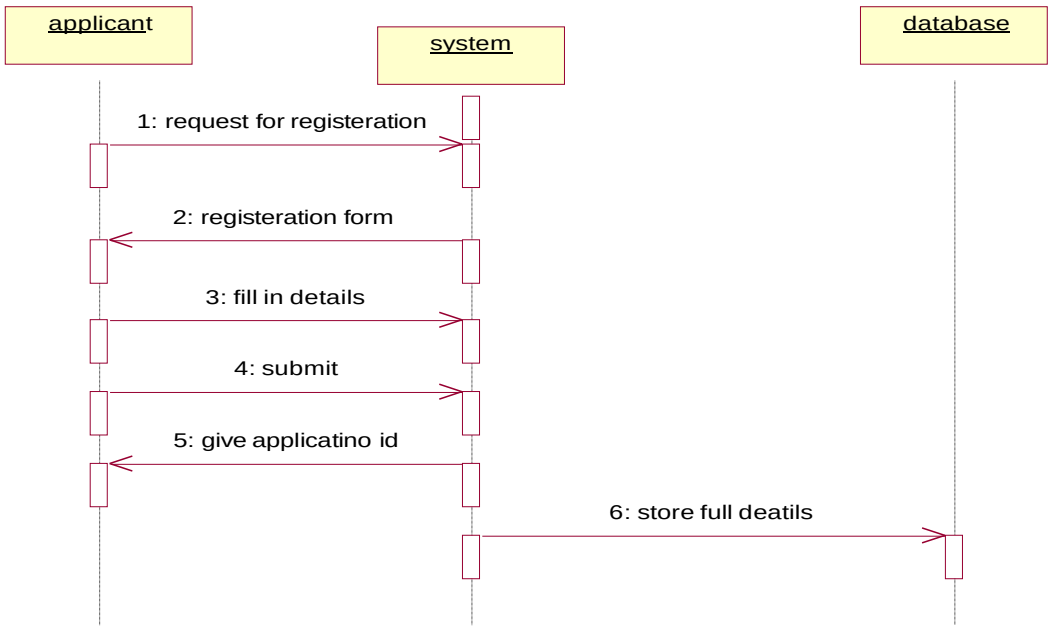


Fig.6.5.SEQUENCE DIAGRAM FOR NEW REGISTRATION

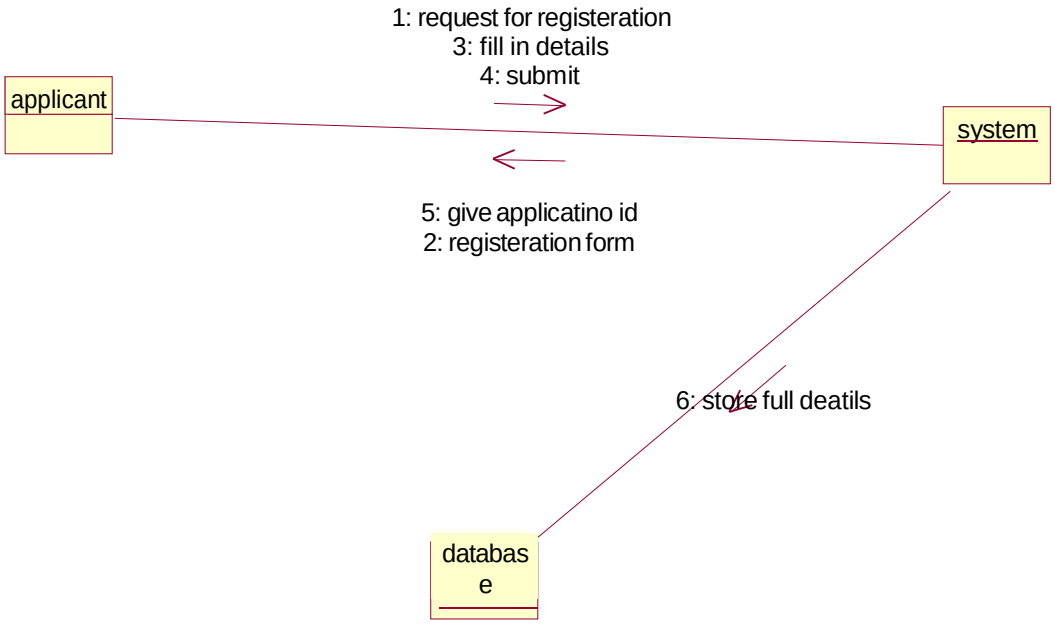
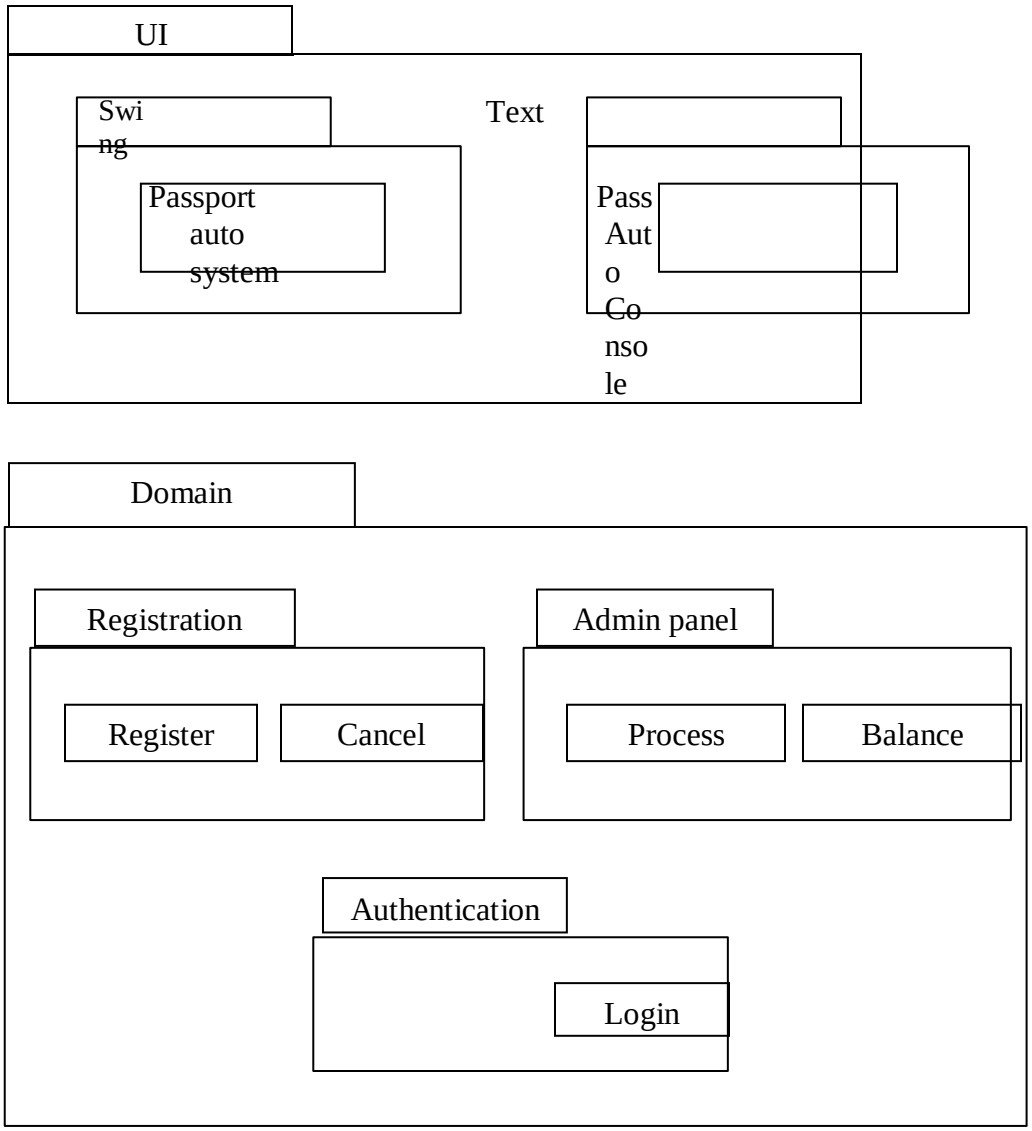
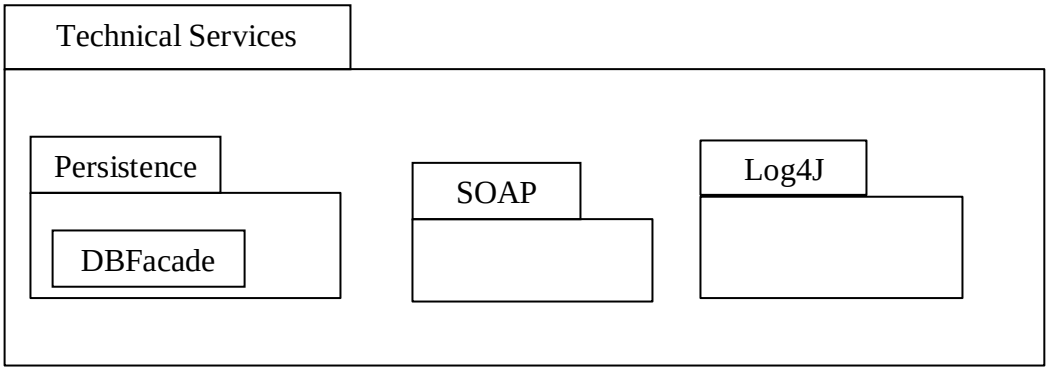


Fig.6.6.COLLABORATION DIAGRAM FOR NEW REGISTRATION

The diagrams show the applicant request the system for registration and the system provide the register form and applicant fill the form and submit and the system give the applicant id. The database stores the full details.

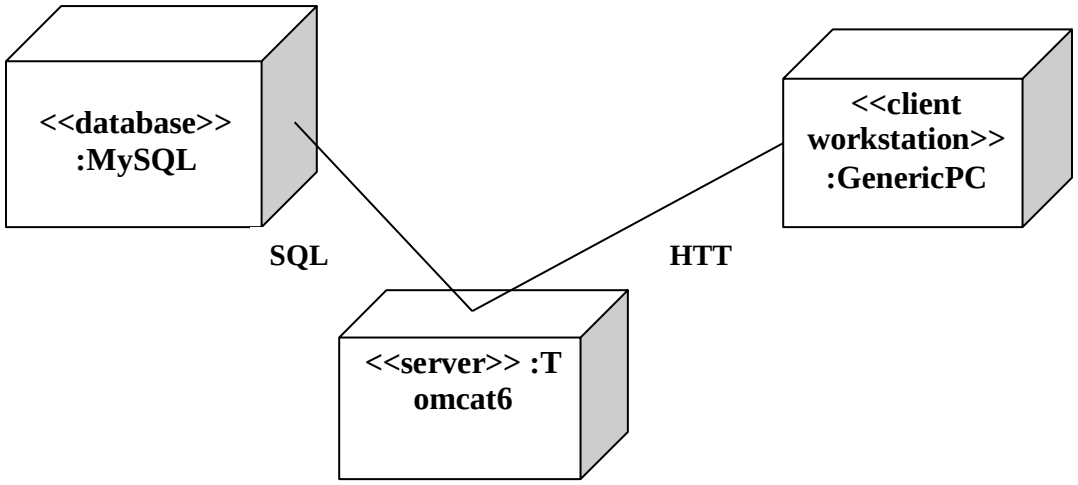
(VII) PARTIAL LAYERD LOGICAL ARCHITECTURE DIAGRAM




Technical Services
Persistence
DBFacade
SOAP
Log4J

(VIII) DEPLOYMENT DIAGRAM AND COMPONENT DIAGRAM

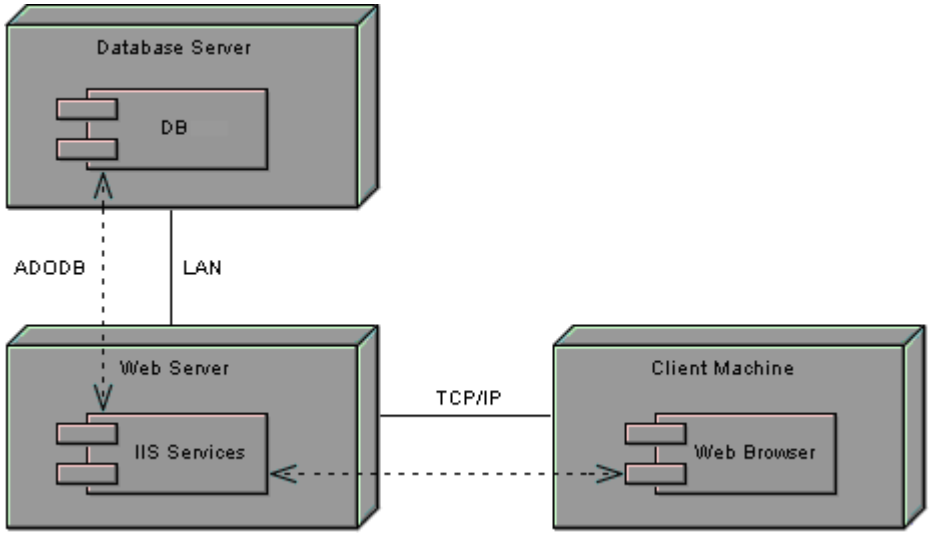
Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed.



DEPLOYMENT DIAGRAM

COMPONENT DIAGRAM

Component diagrams are used to visualize the organization and relationship among components in system



RESULT:

Thus the mini project for passport automation system has been successfully executed and codes are generated.

Ex no:2

BOOK BANK SYSTEM

Date:

AIM:

To create a system to perform book bank operation

(I) PROBLEM STATEMENT:

A Book Bank lends books and magazines to member, who is registered in the system. Also it handles the purchase of new titles for the Book Bank. Popular titles are brought into multiple copies. Old books and magazines are removed when they are out or date or poor in condition. A member can reserve a book or magazine that is not currently available in the book bank, so that when it is returned or purchased by the book bank, that person is notified. The book bank can easily create, replace and delete information about the tiles, members, loans and reservations from the system.

(II) SOFTWARE REQUIREMENTS SPECIFICATION:

2.1SOFTWARE INTERFACE

- **Front End Client** - The Student and Librarian online interface is built using JSP and HTML. The Librarians local interface is built using Java.
- **Web Server** - Glassfish application server (Oracle Corporation).
- **Back End** - Oracle database

2.2HARDWARE INTERFACE

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

(III)USE-CASE DIAGRAM:

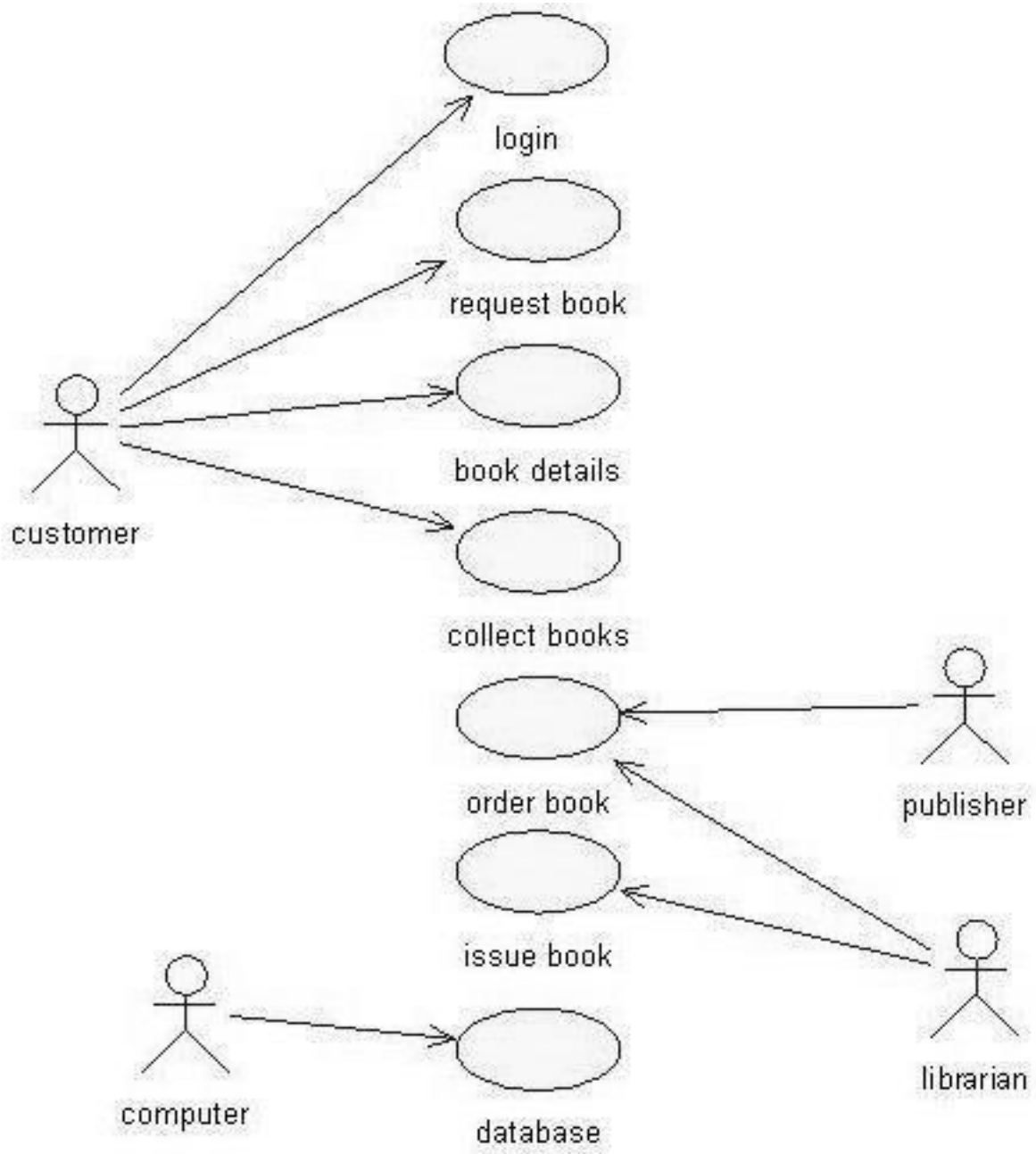


Fig 3. USE-CASE DIAGRAM FOR BOOK BANK SYSTEM

(IV) ACTIVITY DIAGRAM:

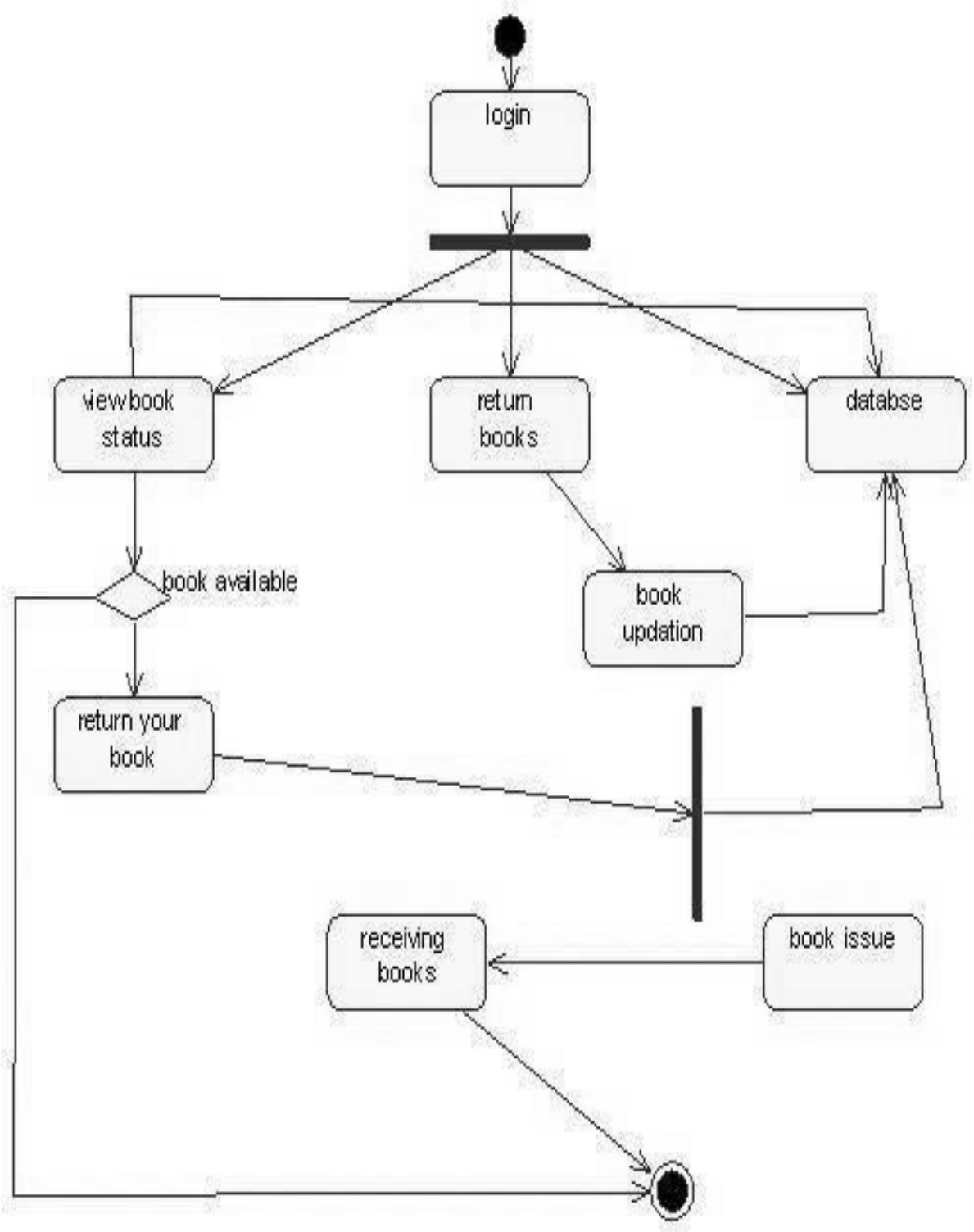


Fig.4. ACTIVITY DIAGRAM

(V) CLASS DIAGRAM:

The class diagram, also referred to as object modeling is the main static analysis diagram. The main task of object modeling is to graphically show what each object will do in the problem domain. The problem domain describes the structure and the relationships among objects.

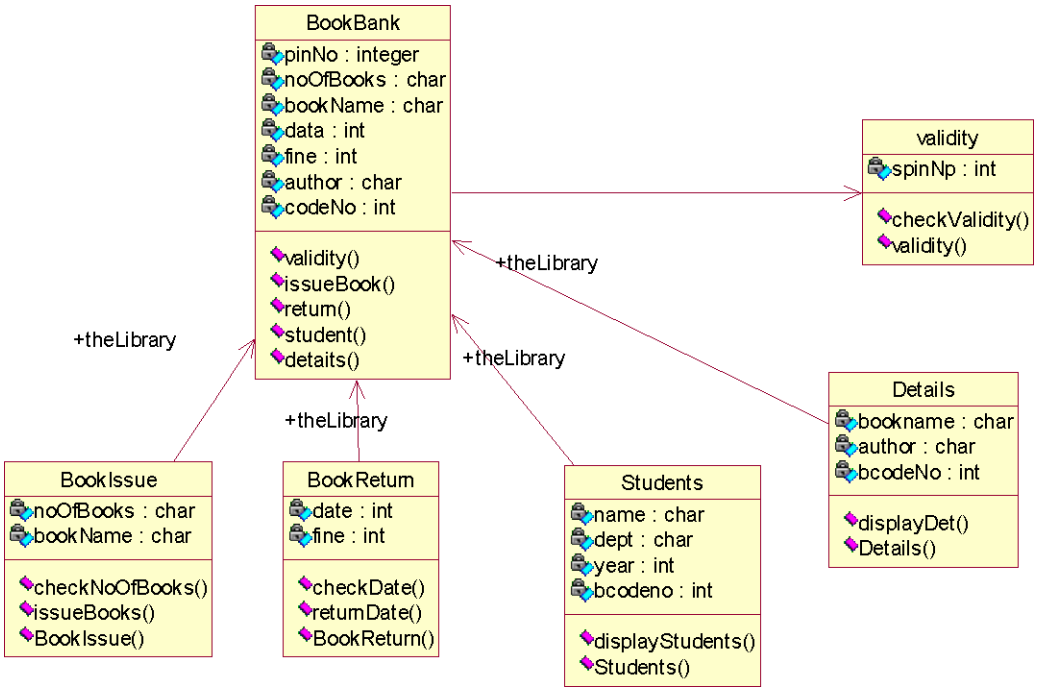


Fig.5. CLASS DIAGRAM FOR BOOK BANK SYSTEM

(VI)) SEQUENCE 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 “form” 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.

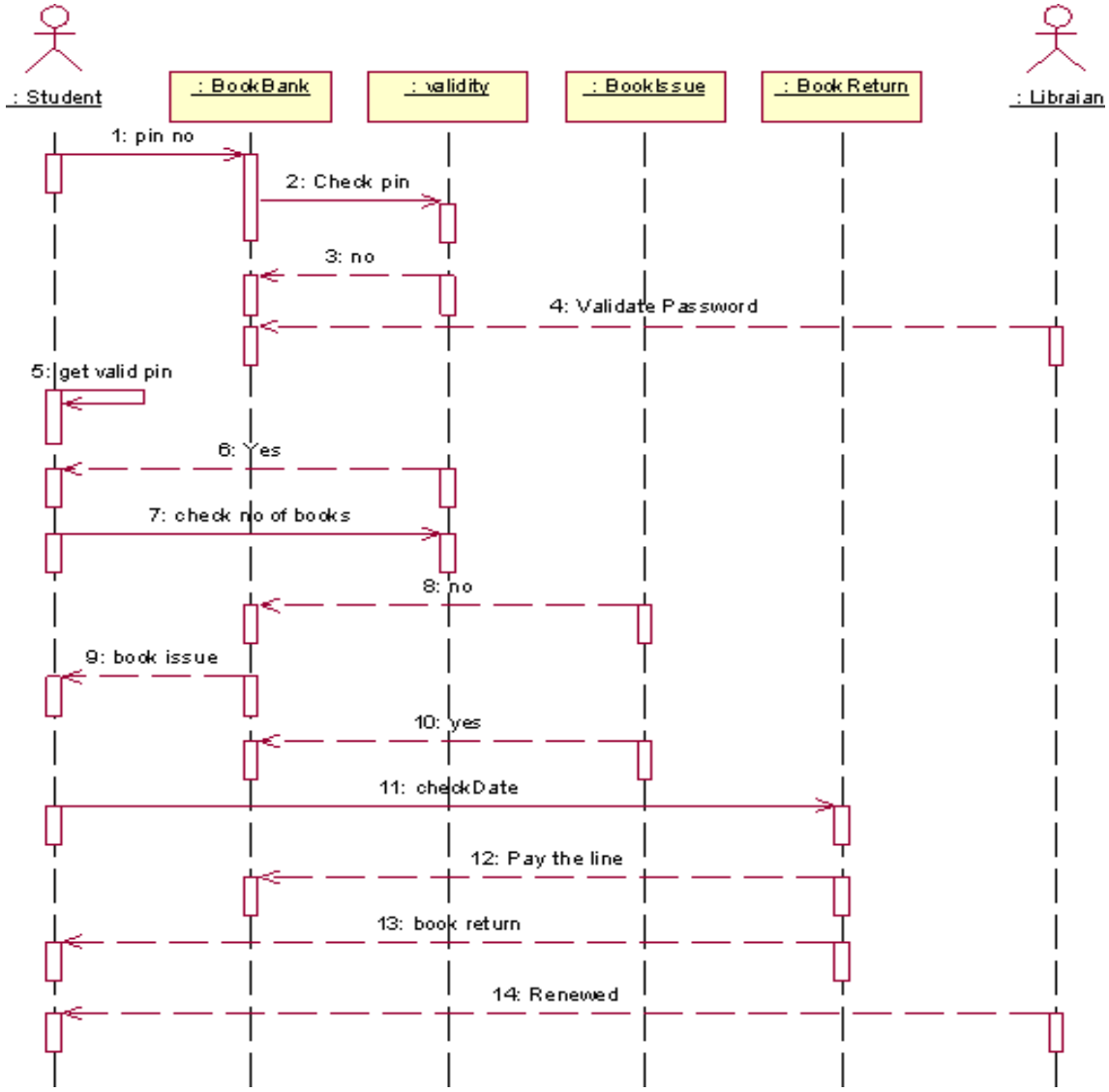


Fig. 6.1. SEQUENCE DIAGRAM FOR DEPOSIT PROCESS

The diagrams show the pin no is entered and check the pin .Get no and validate password check the condition based on condition book issue and return are done. Pay the online and renewed.

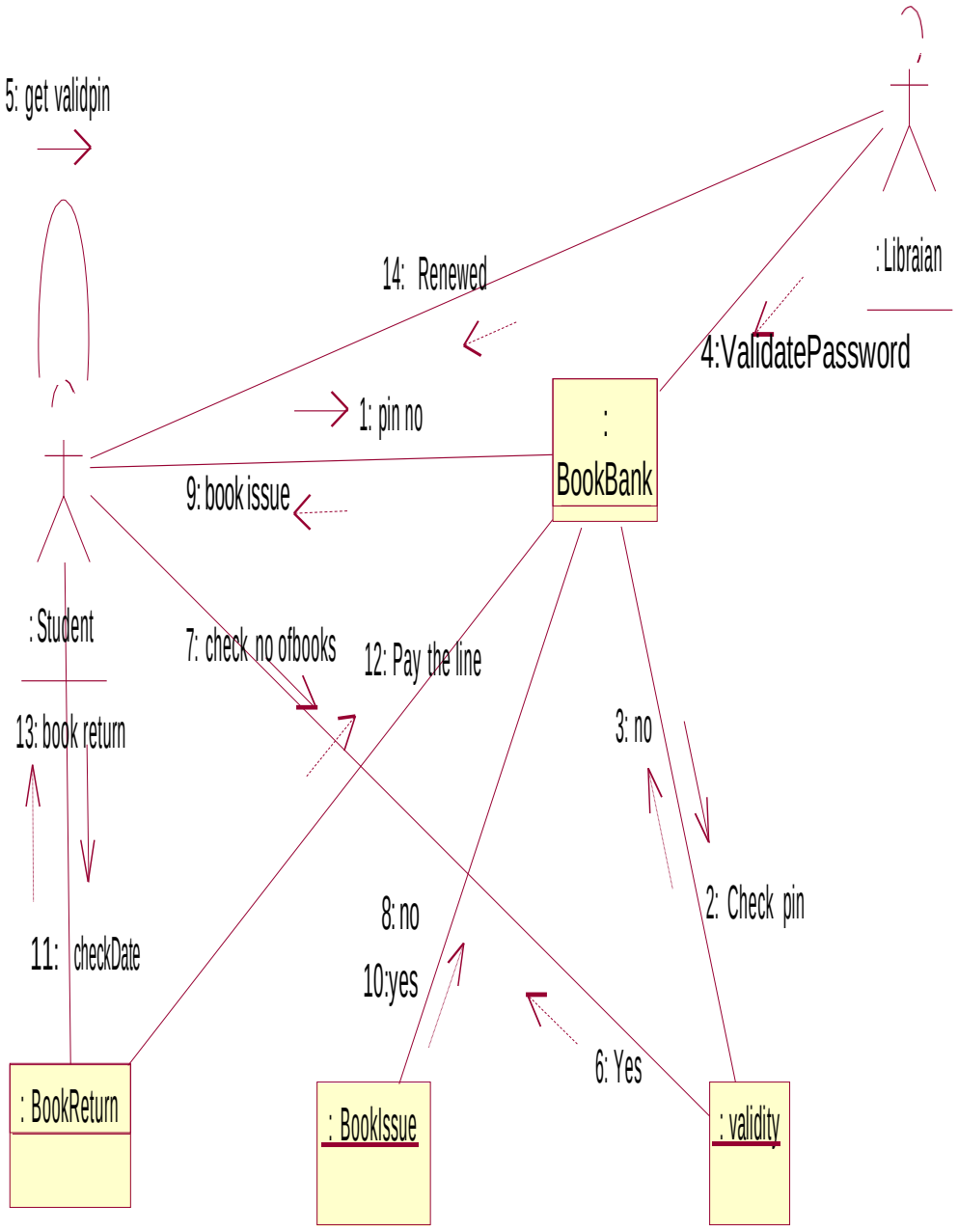
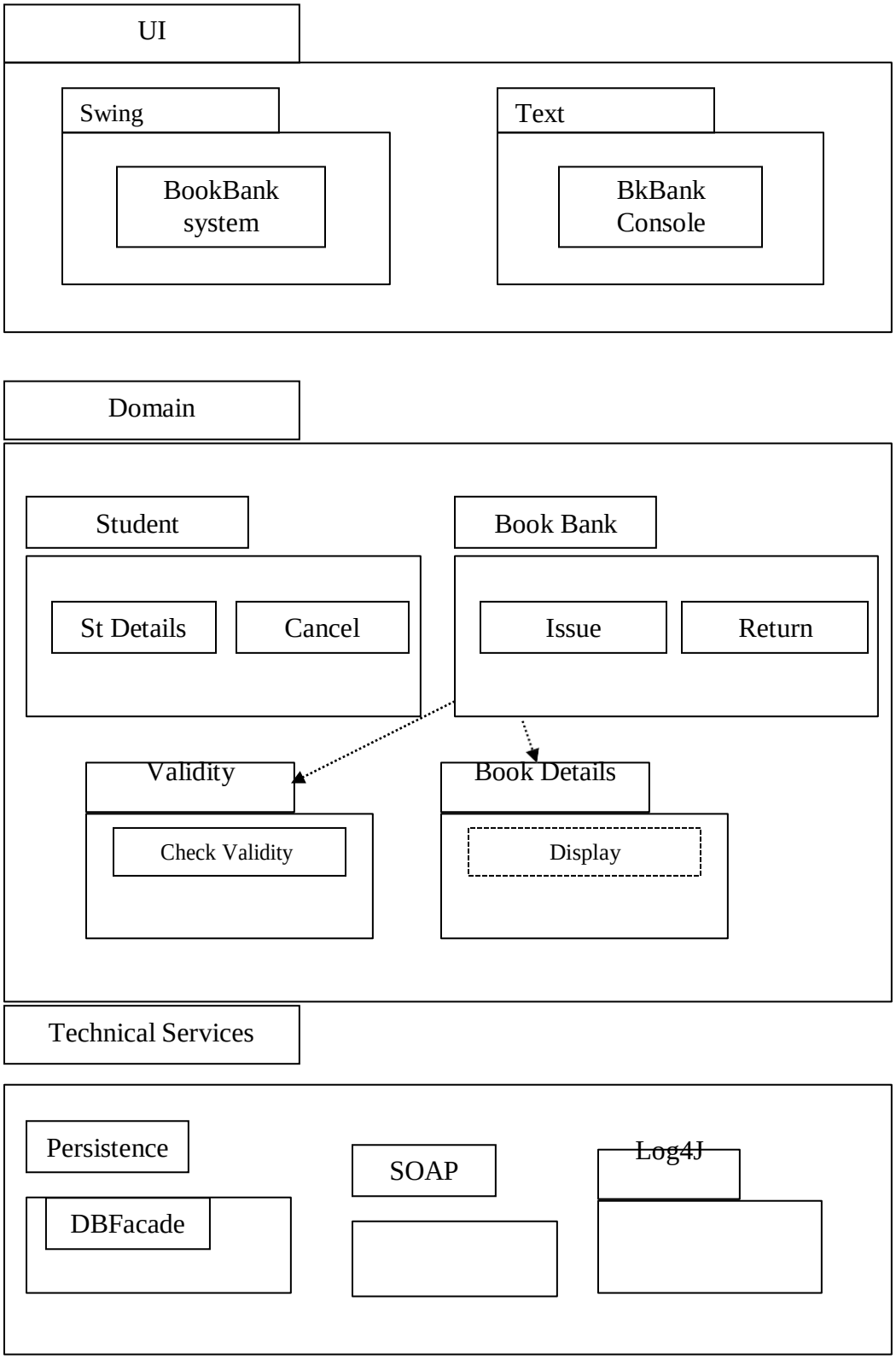


Fig. 6.2. COLLABORATION DIAGRAM FOR DEPOSIT PROCESS

(VII) PARTIAL LAYERD LOGICAL ARCHITECTURE 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.

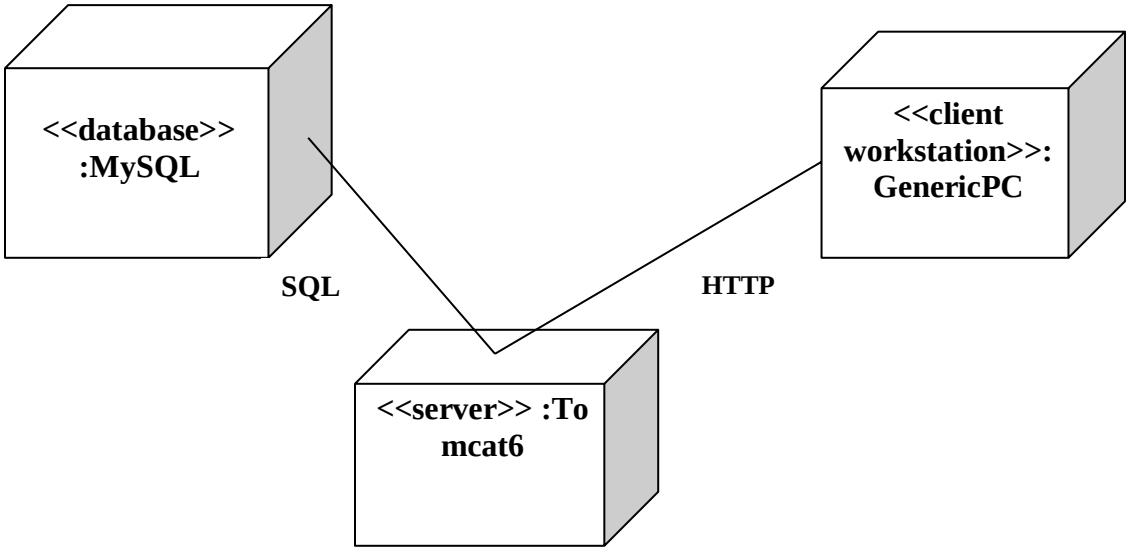


Fig.8.1.DEPLOYMENT DIAGRAM

COMPONENT DIAGRAM

Component diagrams are used to visualize the organization and relationships

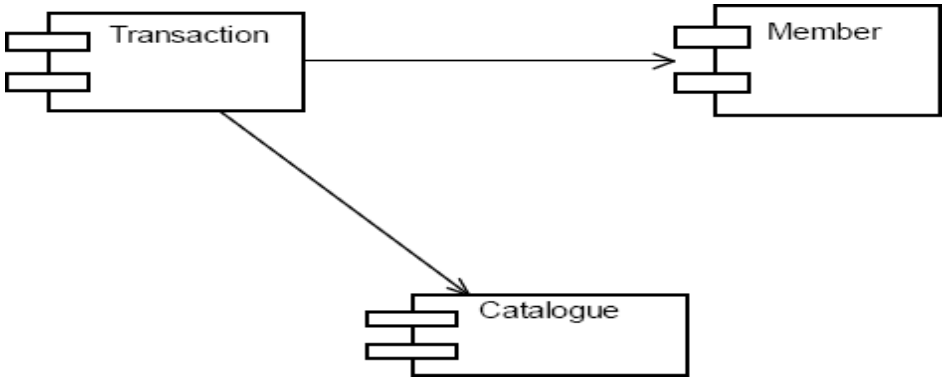


Fig.8.2.COMPONENT DIAGRAM

RESULT:

Thus the mini project for Book Bank System has been successfully executed and codes are generated.

EXAM REGISTRATION SYSTEM

Exp :03

AIM:

To create a system to perform the Exam Registration system

(I) PROBLEM STATEMENT:

Exam Registration system.is used in the effective dispatch of registration form to all of the students. This system adopts a comprehensive approach to minimize the manual work and schedule resources, time in a cogent manner. The core of the system is to get the online registration form (with details such as name, reg.no etc.,) filled by the student whose testament is verified for its genuineness by the Exam Registration System with respect to the already existing information in the database.

(II)SOFTWARE REQUIREMENT SPECIFICATION:

2.1SOFTWARE INTERFACE

- **Front End Client** - The student and Controller online interface is built using
  JSP and HTML. The Exam Controller's local interface is built using Java.
- **Web Server** - Glassfish application server(SQlCorporation).
- **Back End** - SQL database.

2.2HARDWARE INTERFACE

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

(III)USECASE DIAGRAM:

The Exam Registration use cases in our system are:

1. Login
2. View exam details
3. View fees details

4. Pay fee
5. Display details
6. Logout

USECASE DIAGRAM :

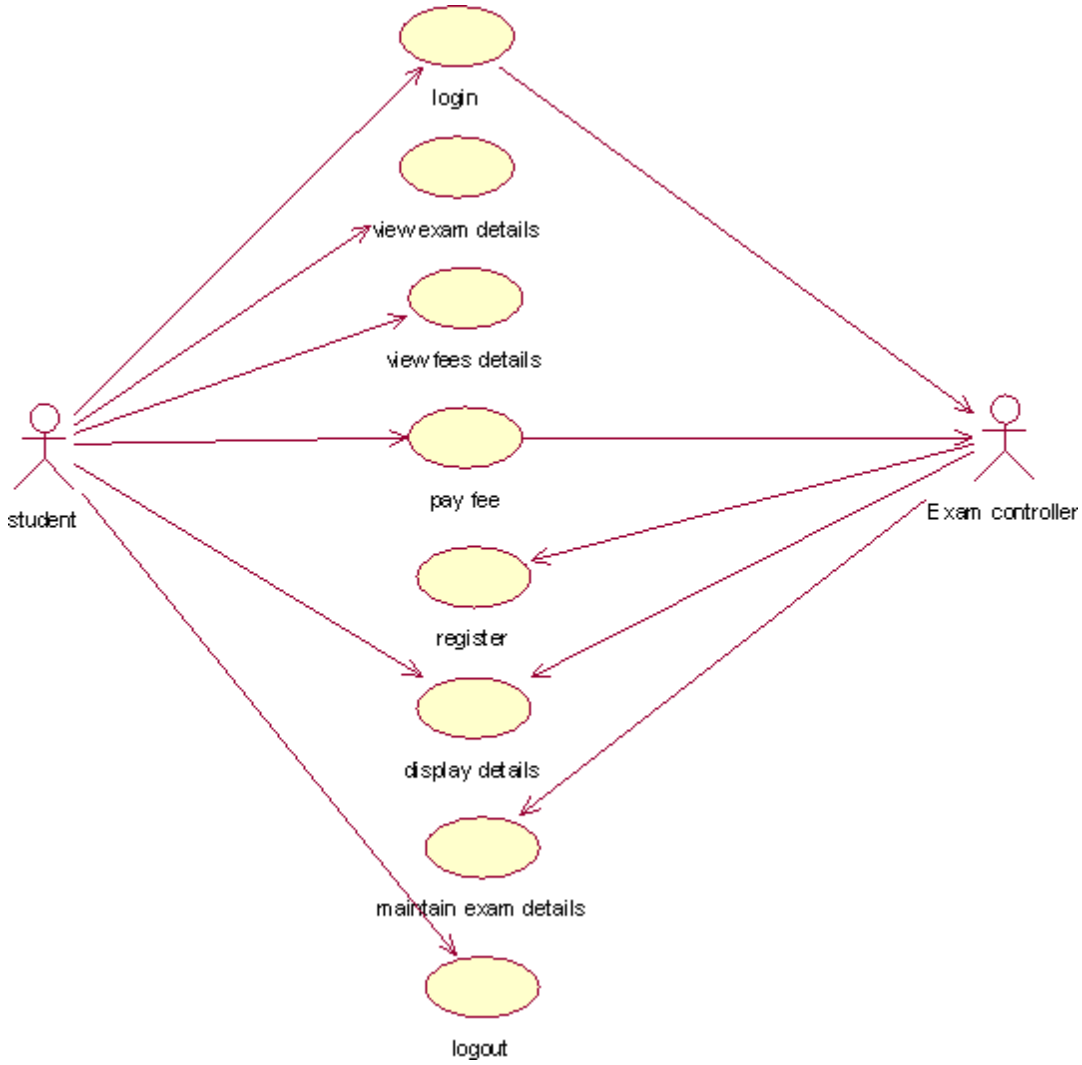


Fig. 3.USECASE DIAGRAM FOR EXAM REGISTRATION SYSTEM

(IV) ACTIVITY DIAGRAM:

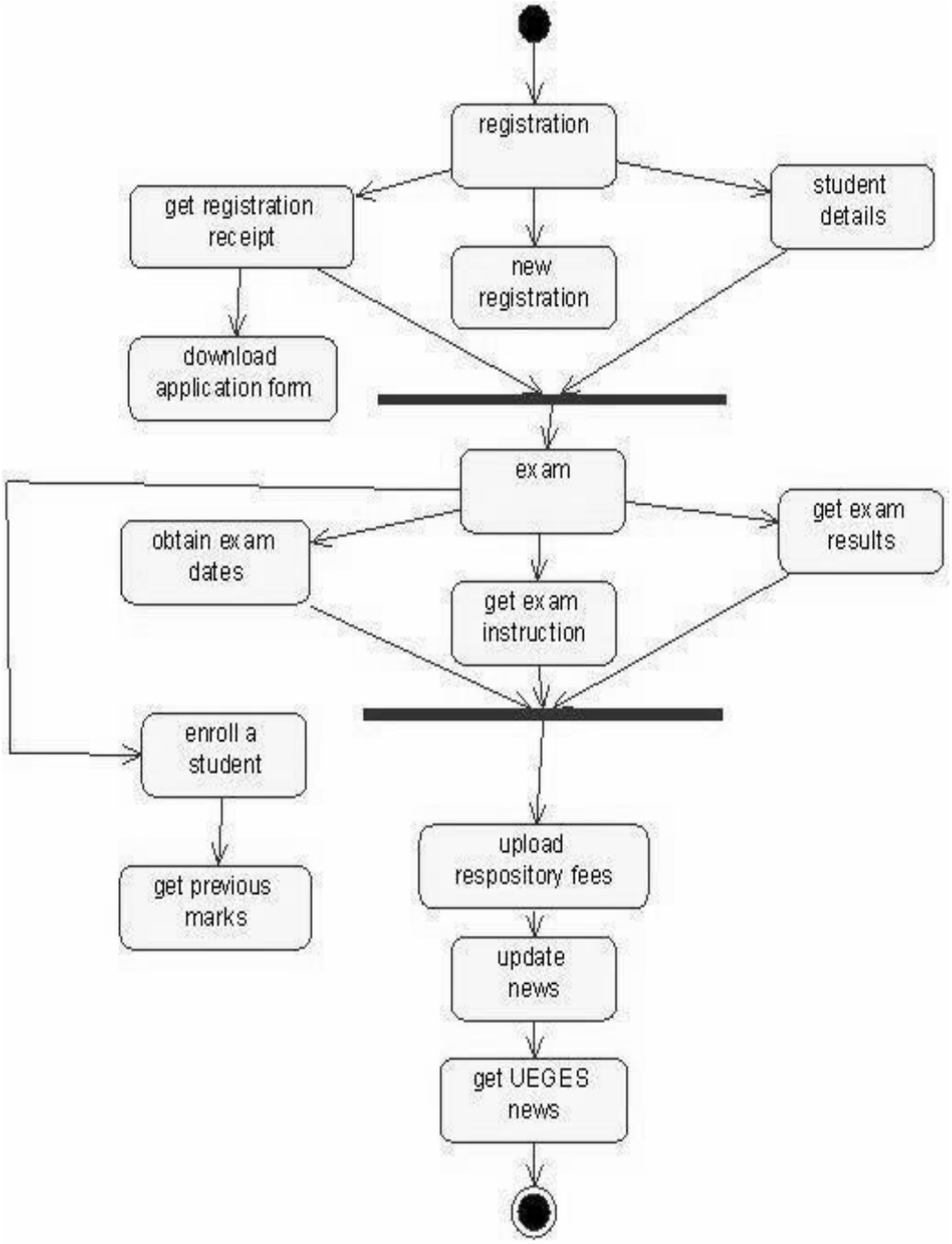


Fig. 4.USECASE DIAGRAM FOR EXAM REGISTRATION SYSTEM

(V)CLASS DIAGRAM:

The class diagram, also referred to as object modeling is the main static analysis diagram. The main task of object modeling is to graphically show what each object will do in the problem domain. The problem domain describes the structure and the relationships among objects.

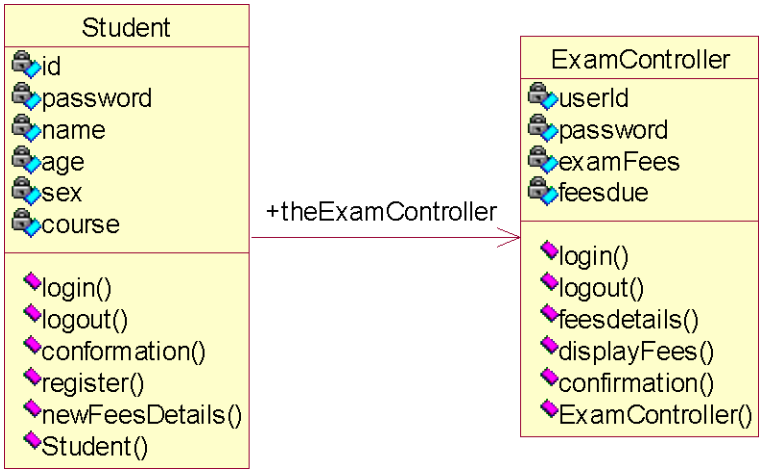


Fig.5. CLASS DIAGRAM FOR EXAM REGISTRATION SYSTEM

(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

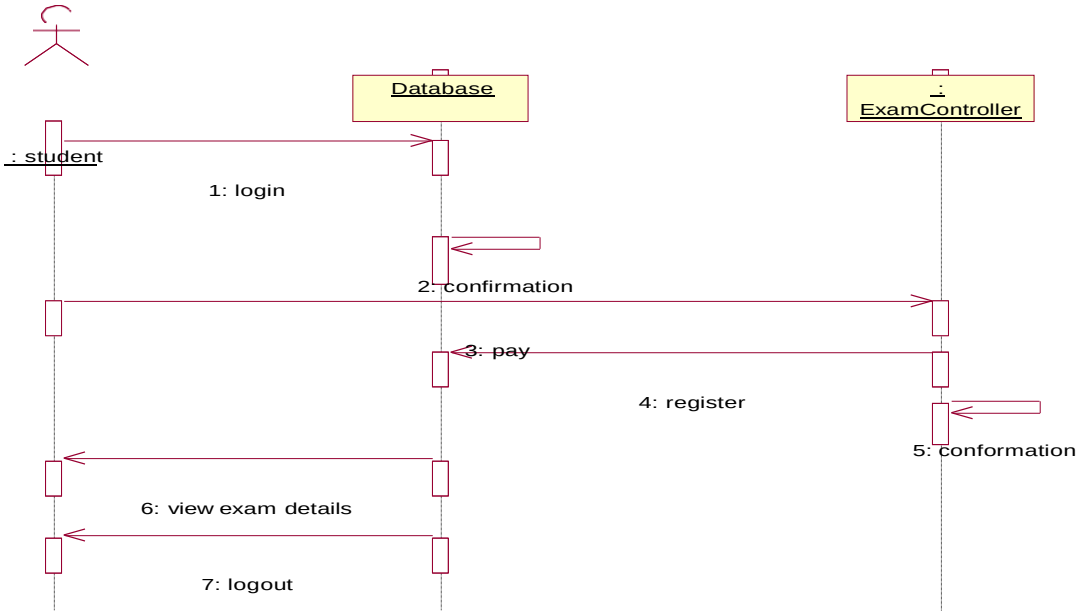


Fig. 6.1. SEQUENCE DIAGRAM FOR REGISTRATION SYSTEM

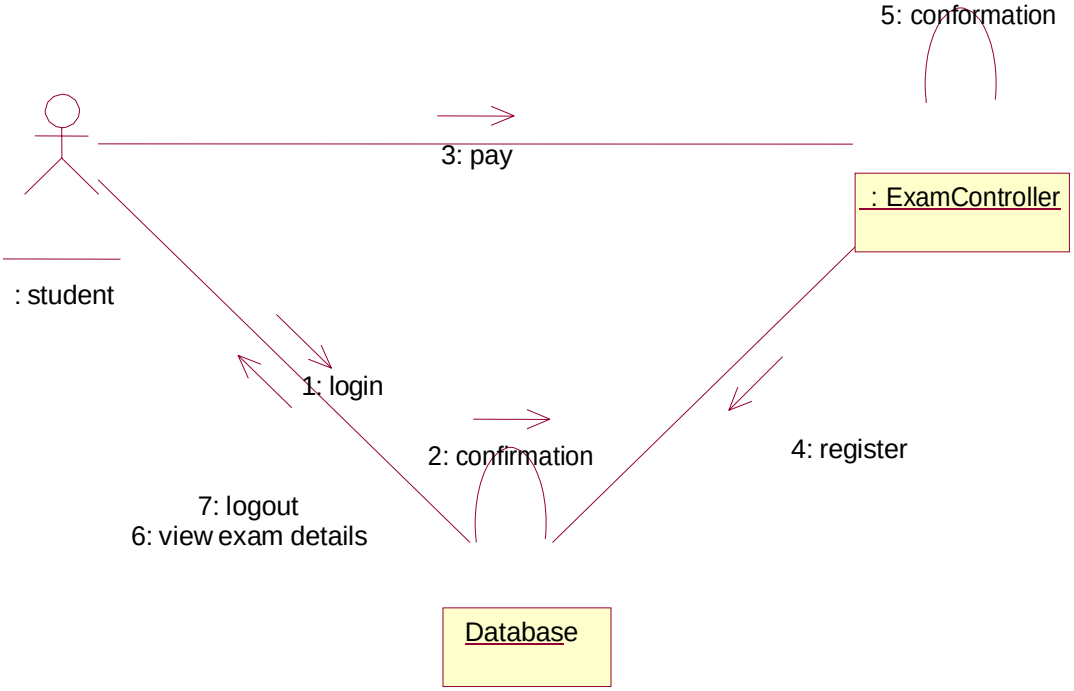
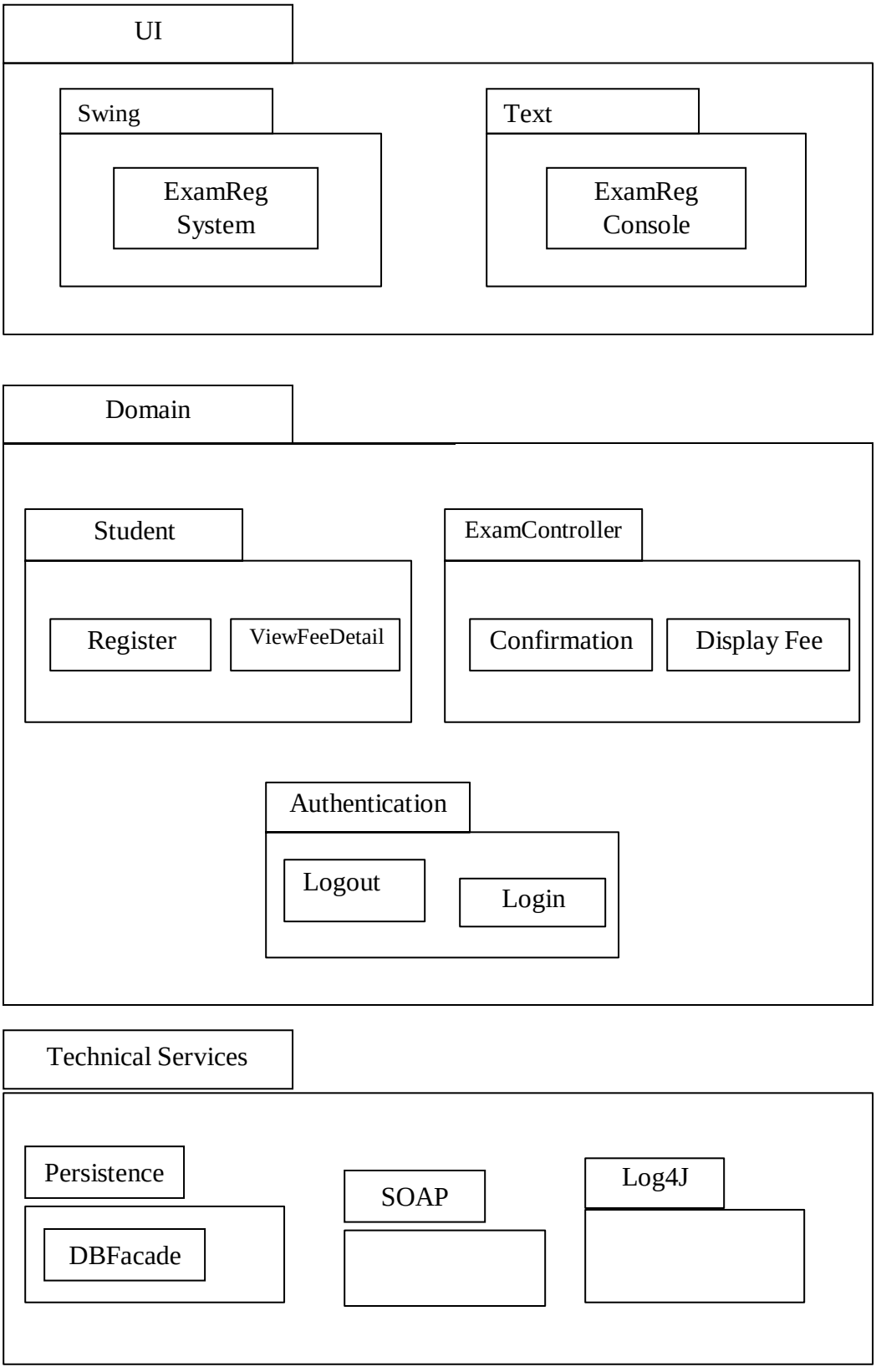


Fig. 6.2. COLLABORATION DIAGRAM FOR REGISTRATION SYSTEM

PARTIAL LAYERD LOGICAL ARCHITECTURE DIAGRAM:



(VII) 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.

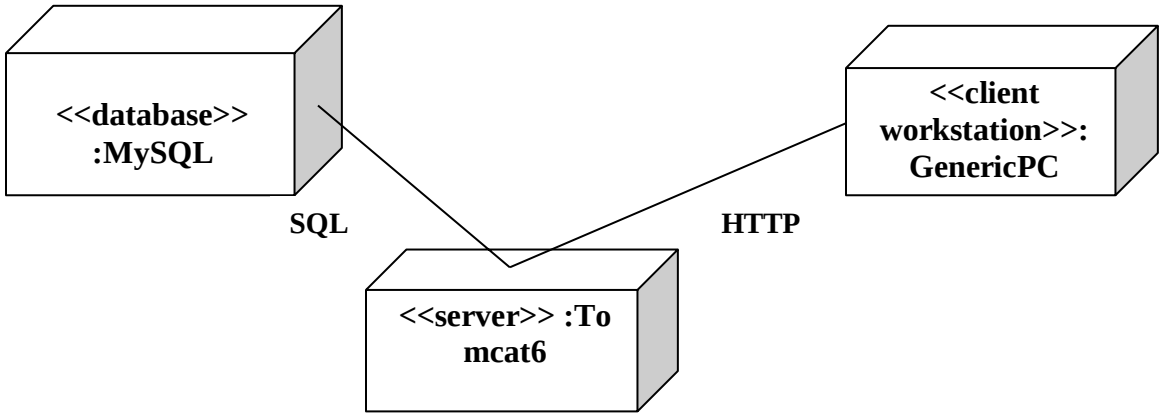


Fig.7.1.DEPLOYMENT DIAGRAM

COMPONENT DIAGRAM

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

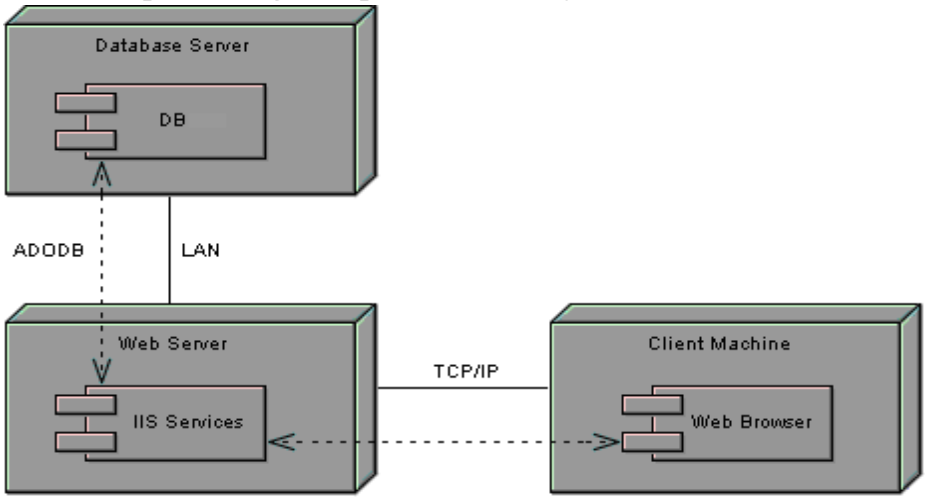


Fig.7.2.COMPONENT DIAGRAM

RESULT:

Thus the mini project for Exam Registration system has been successfully executed and codes are generated.

Ex no: 4

STOCK MAINTENANCE

Date:

AIM:

To create a system to perform the Stock maintenance

(I) PROBLEM STATEMENT

The stock maintenance system must take care of sales information of the company and must analyze the potential of the trade. It maintains the number of items that are added or removed.The sales person initiates this Use case. The sales person is allowed to update information and view the database.

(II) SOFTWARE REQUIREMENT SPECIFICATION

PURPOSE

The entire process of Stock maintenanceis done in a manual manner Considering the fact that the number of customers for purchase is increasing every year, a maintenance system is essential to meet the demand. So this system uses several programming and database techniques to elucidate the work involved in this process.

SCOPE

• The System provides an interface to the customer where they can fill in orders for the item needed.

• The sales person is concerned with the issue of items and can use this system.

• Provide a communication platform between the customer and the sales person.

TOOLS TO BE USED

• Eclipse IDE (Integrated Development Environment)

• Rational Rose tool (for developing UML Patterns)

(III) USE CASE DIAGRAM

The functionality of a system can be described in a number of different use-cases, each of which represents a specific flow of events in a system. It is a graph of actors, a set of use-cases enclosed in a boundary, communication, associations between the actors and the use-cases, and generalizationamong the use-cases

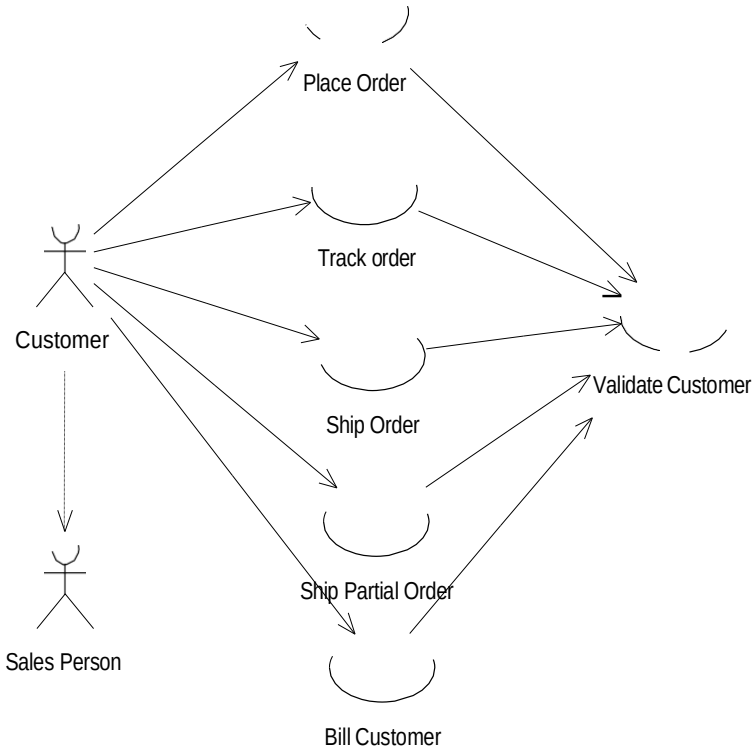


Fig.3. USE CASE DIAGRAM

(IV) ACTIVITY DIAGRAM

It shows organization and their dependence among the set of components. These diagrams are particularly useful in connection with workflow and in describing behavior that has a lot of parallel processing. An activity is a state of doing something: either a real-world process, or the execution of a software routine.

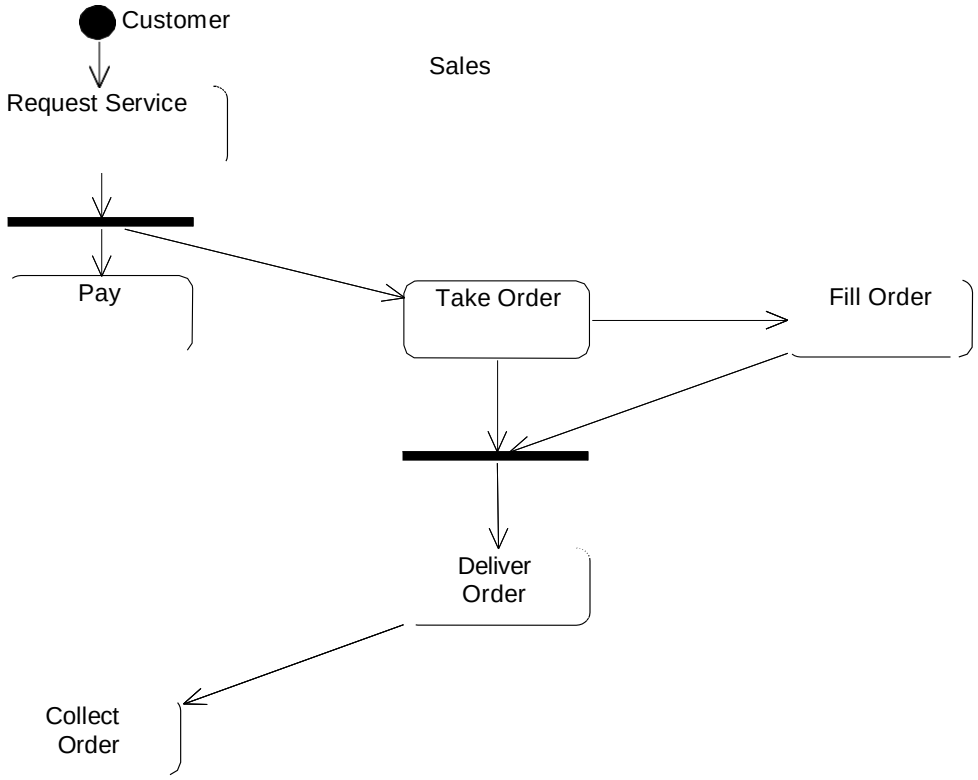


Fig.4. ACTIVITY DIAGRAM

(V) CLASS DIAGRAM

Description:

- A class diagram describes the type of objects in system and various kinds of relationships that exists among them.
- Class diagrams and collaboration diagrams are alternate representations of object models.

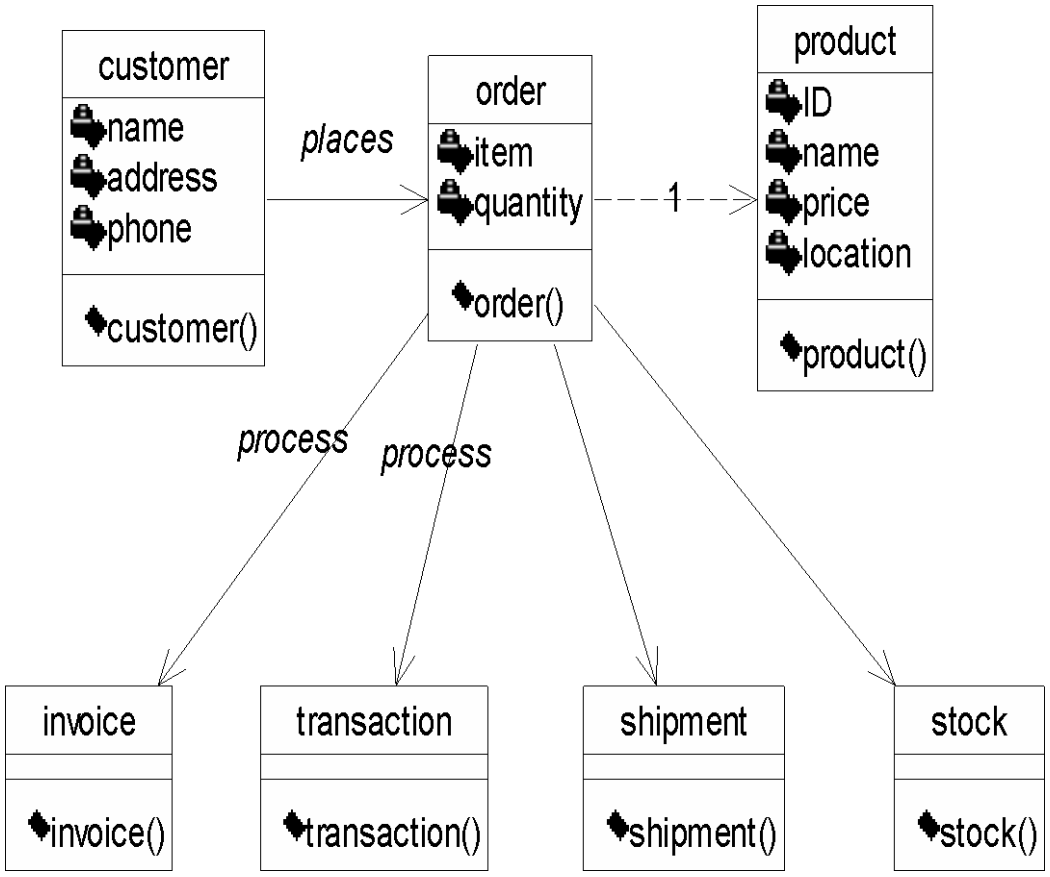


Fig.5. CLASS DIAGRAM

(VI) UML INTERACTION DIAGRAMS

It is the combination of sequence and collaboration diagram. It is used to depict the flow of events in the system over a timeline. The interaction diagram is a dynamic model which shows how the system behaves during dynamic execution.

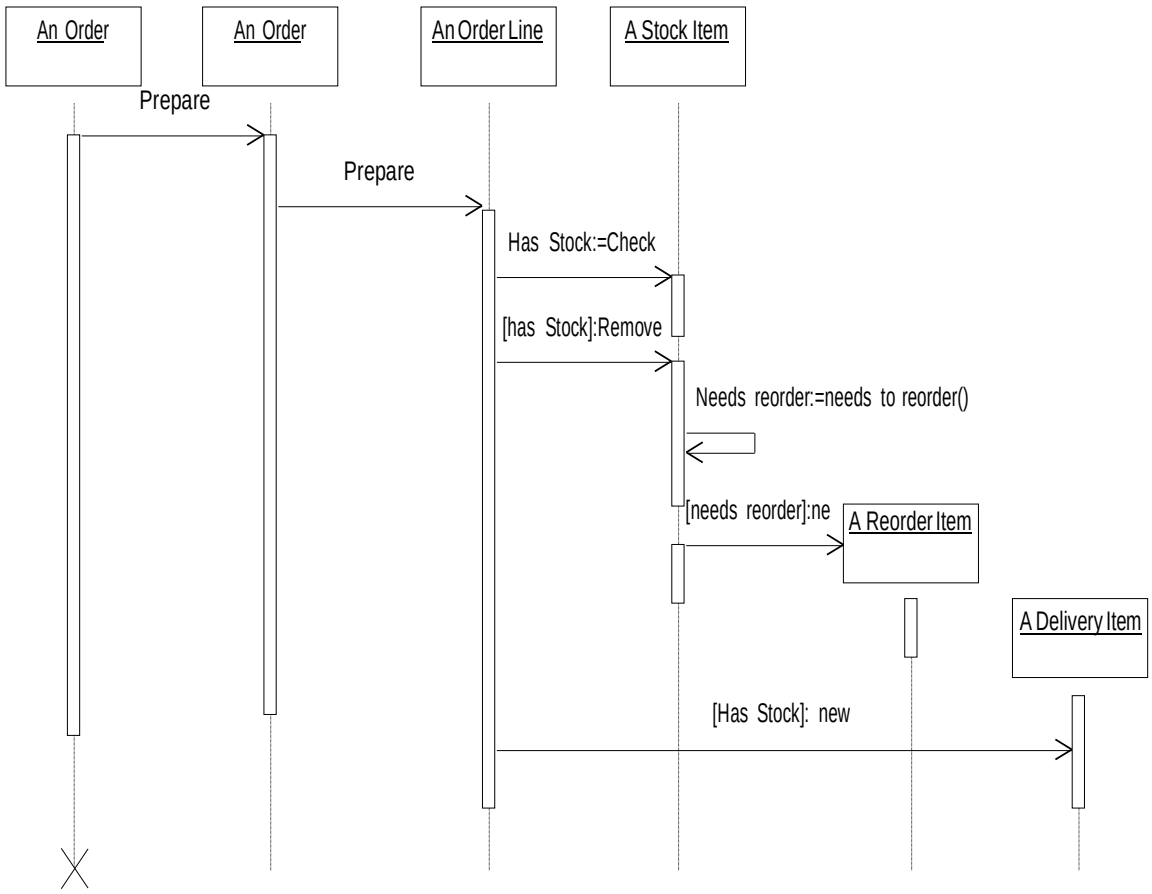


Fig.6.1 SEQUENCE DIAGRAM

COLLABORATION DIAGRAM

Collaboration diagram and sequence diagrams are alternate representations of an interaction. A collaboration diagram is an interaction diagram that shows the order of messages that implement an operation or a transaction.

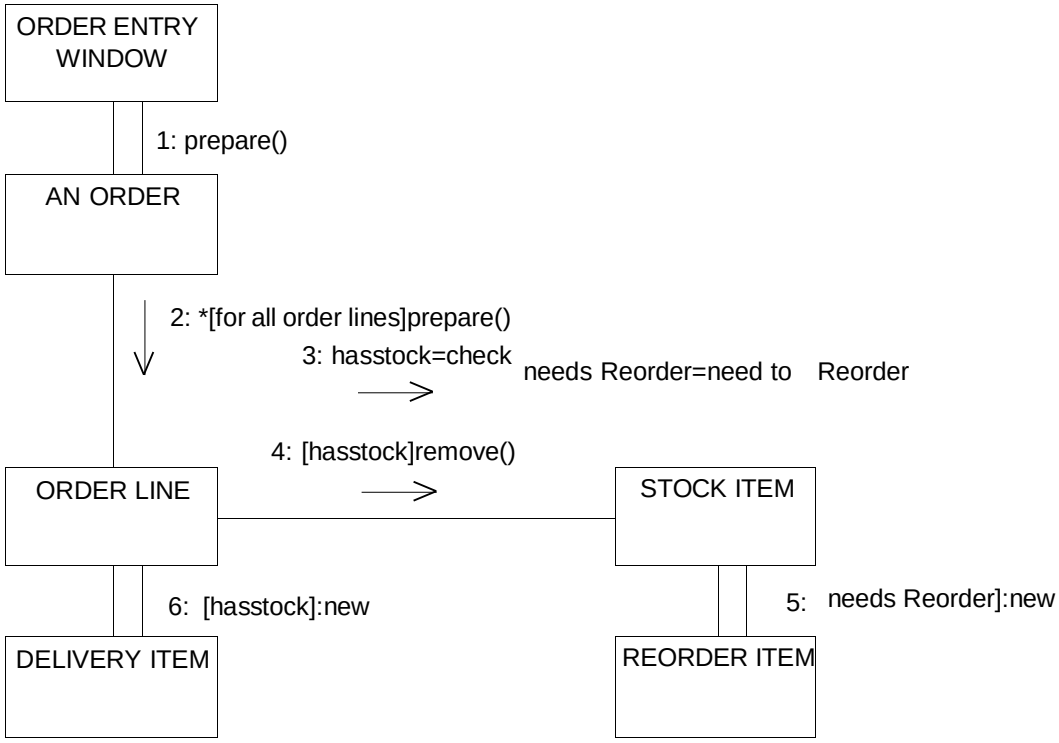
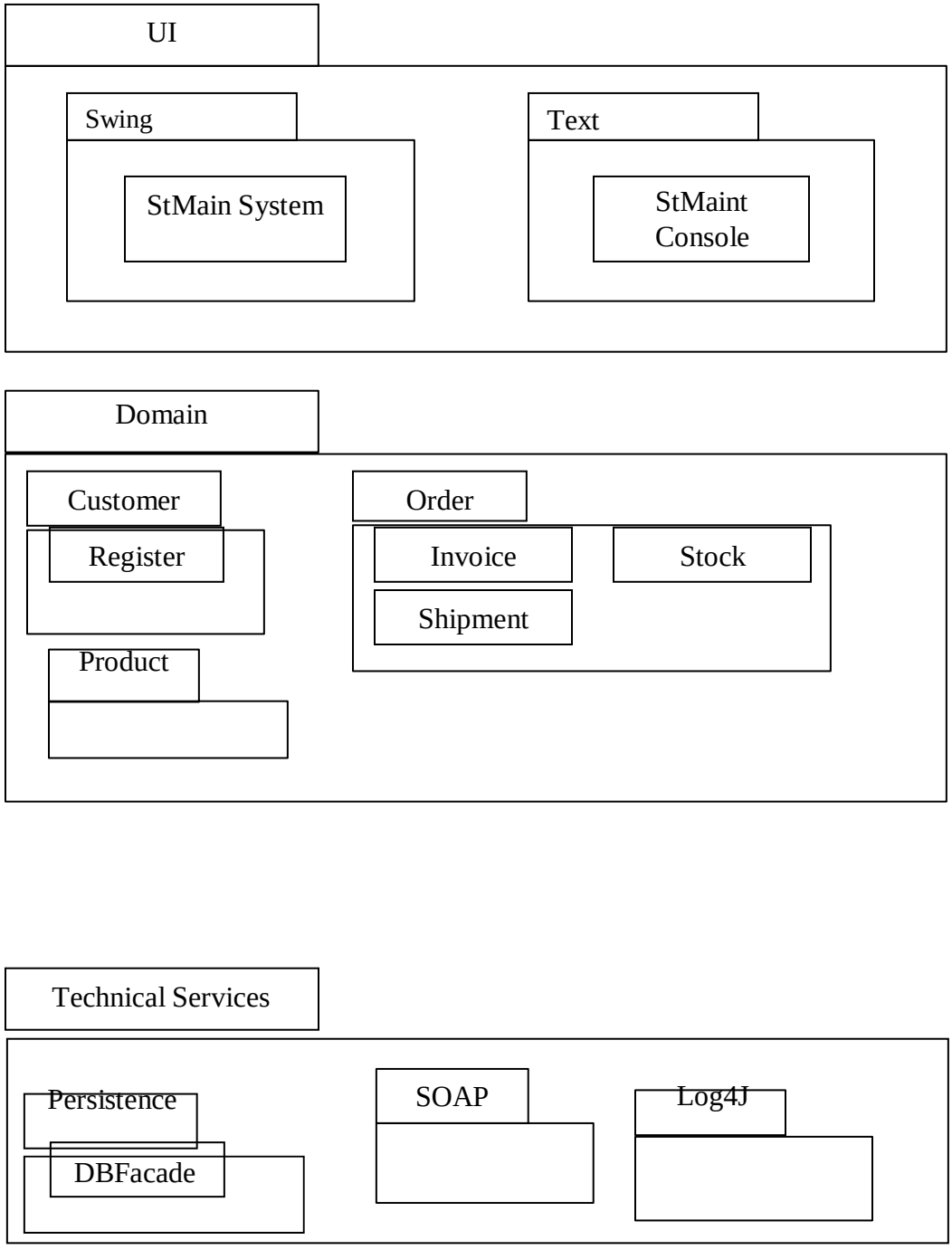


Fig.6.2 COLLABORATION DIAGRAM

(VII) PARTIAL LAYERD LOGICAL ARCHITECTURE 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.

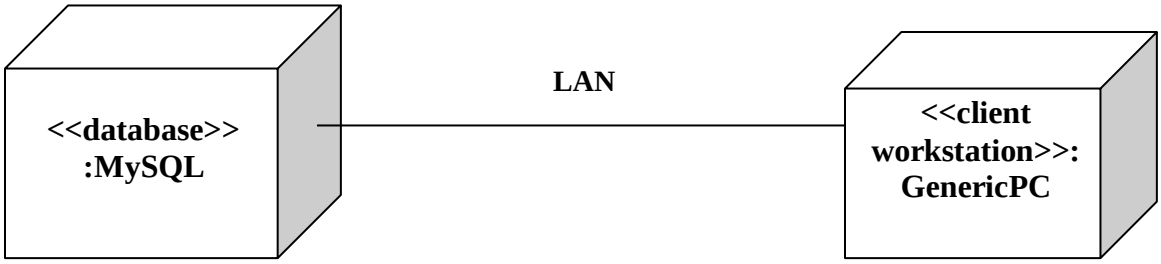


Fig.8.1.DEPLOYMENT DIAGRAM

Component Diagram

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

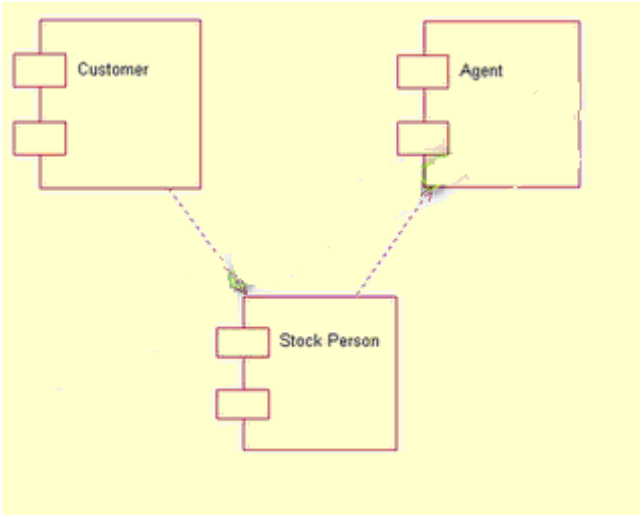


Fig.8.2.COMPONENT DIAGRAM

RESULT:

Thus the mini project for stock maintenance system has been successfully executed and codes are generated.

EX NO: 6 ONLINE COURSE RESERVATION SYSTEM

Date:

AIM

To design an object oriented model for course reservation system.

(I) PROBLEM STATEMENT

a. Whenever the student comes to join the course he/she should be provided with the list of course available in the college.

b. The system should maintain a list of professor who is teaching the course. At the end of the course the student must be provided with the certificate for the completion of the course.

(II) SYSTEM REQUIEMENT SPECIFICATION

OBJECTIVES

a. The main purpose of creating the document about the software is to know about the list of the requirement in the software project part of the project to be developed.

b. It specifies the requirement to develop a processing software part that completes the set of requirement.

SCOPE

a. In this specification, we define about the system requirements that are about from the functionality of the system.

b. It tells the users about the reliability defined in usecase specification

FUNCTIONALITY

Many members of the process line to check for its occurrences and transaction, we are have to carry over at sometimes

USABILITY

The user interface to make the transaction should be effectively

PERFORMANCE

It is the capability about which it can performed function for many user at sometimes efficiently (ie) without any ever occurrences

RELIABILITY

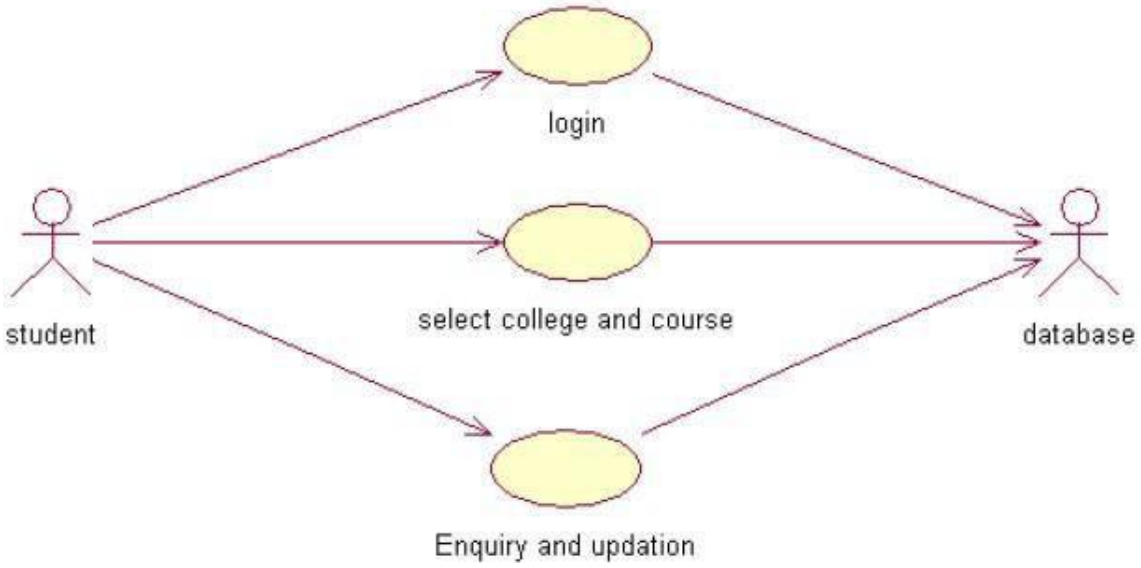
The system should be able to the user through the day to day transaction

(III) USERCASE DIAGRAM

a. Use case is a sequence of transaction in a system whose task is to yield result of measurable value to individual author of the system

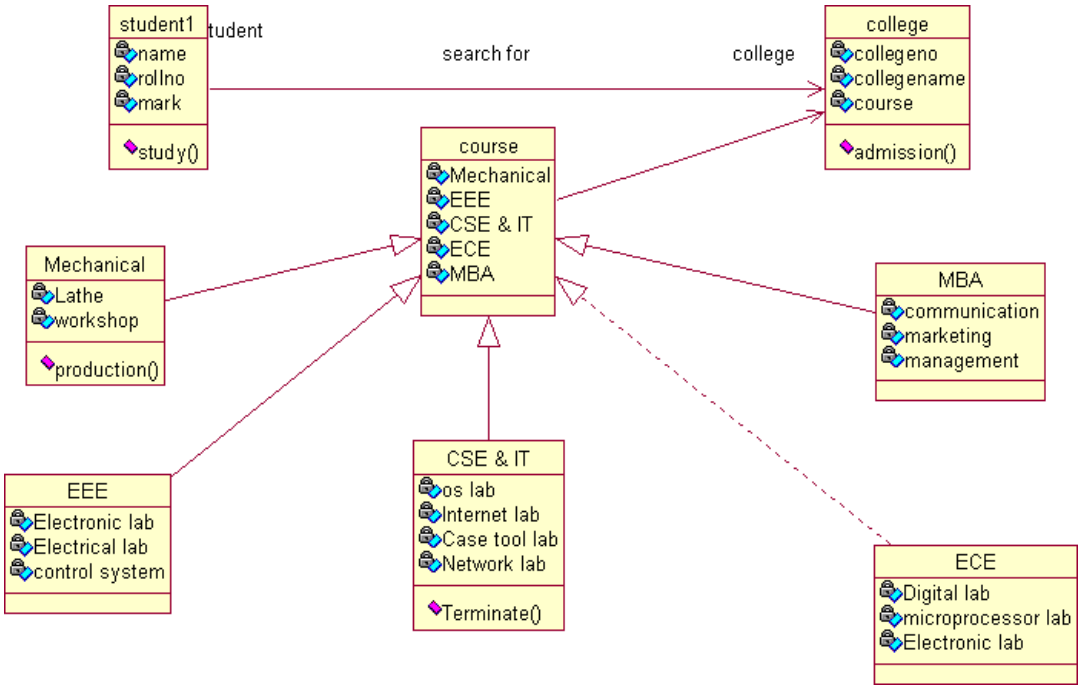
b. Use case is a set of scenarios together by a common user goal

c. A scenario is a sequence of step describing as interaction between a user and a system



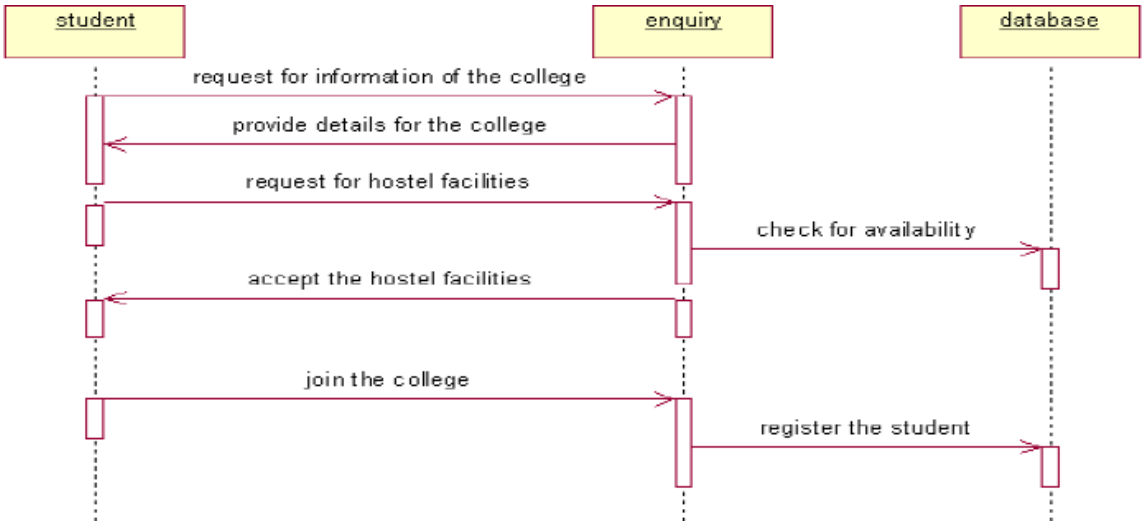
CLASS DIAGRAM:

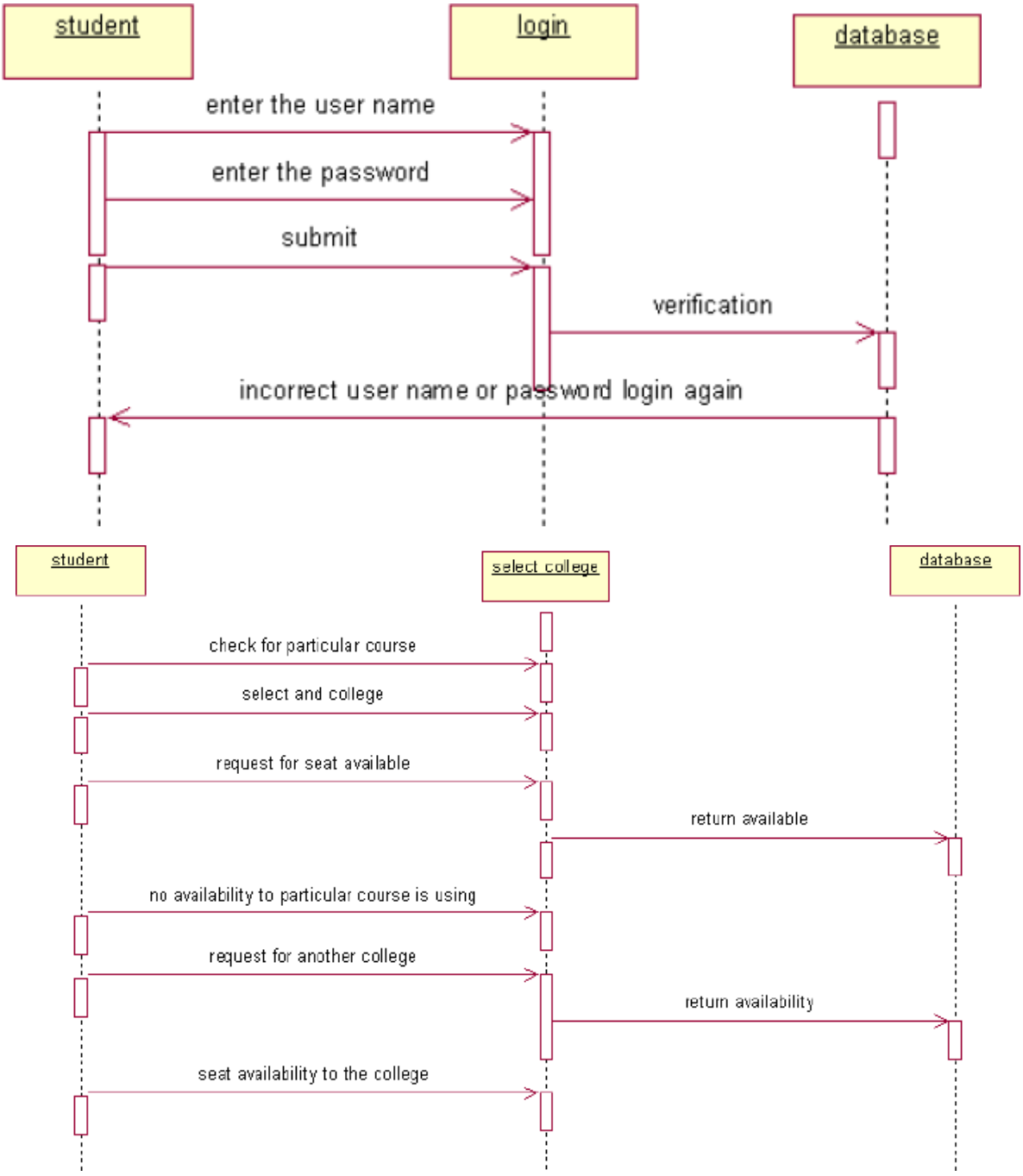
A class diagram describes the type of objectors in the system the various kinds of static relationship that exist among them.



SEQUENCE DIAGRAM

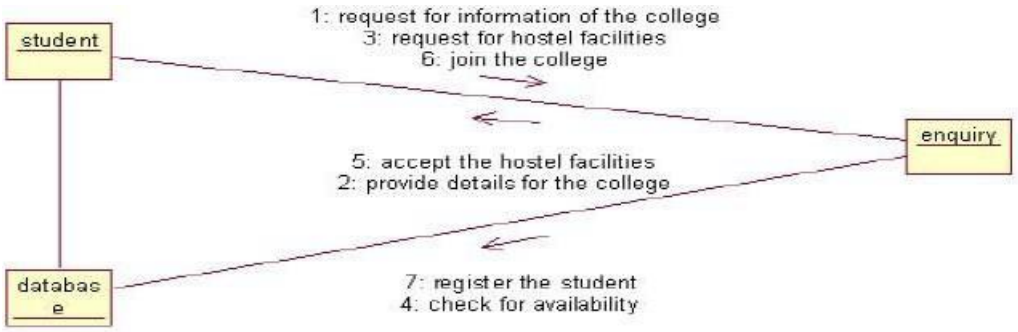
A sequence diagram is one that includes the object of the projects and tells the lifetimes and also various action performed between objects.

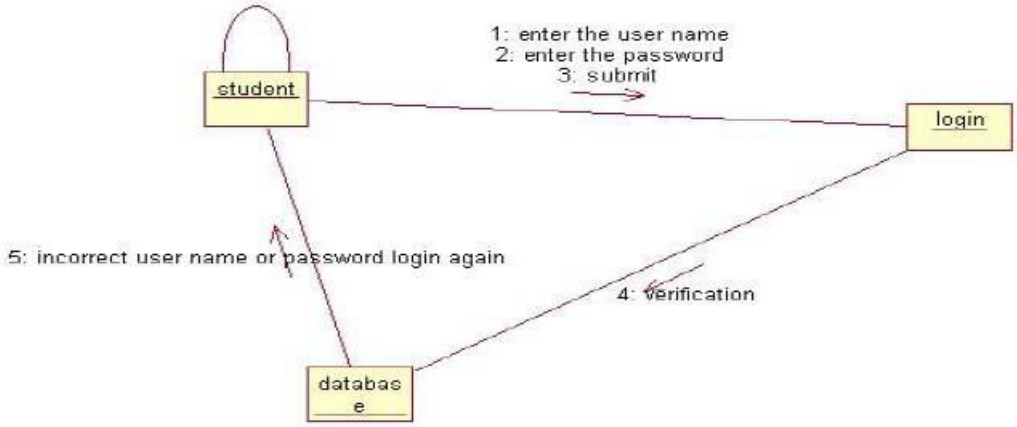


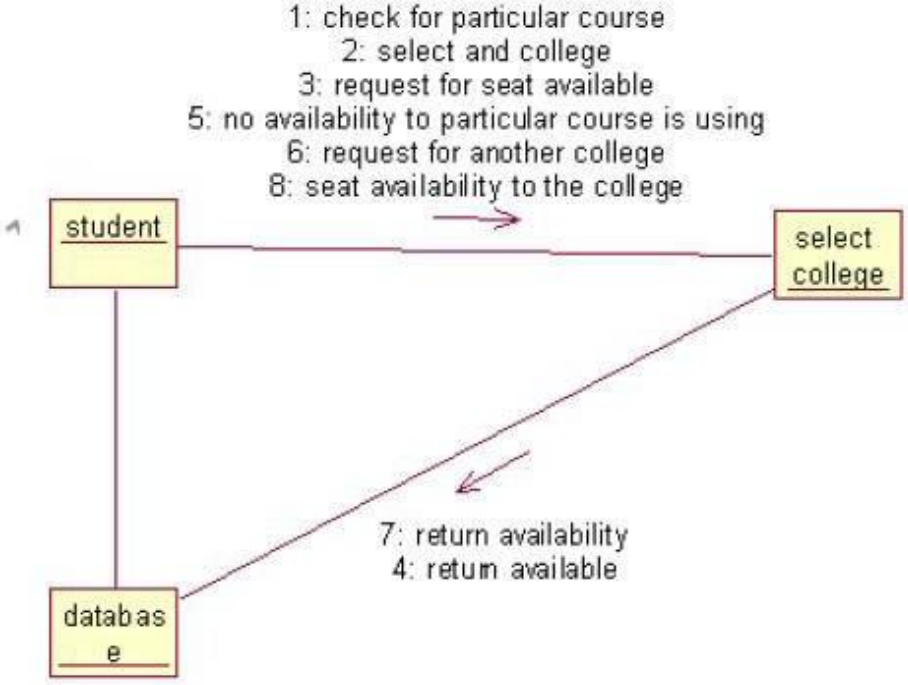


COLLOBORATIION DIAGRAM

It is same as the sequence diagram that involved the project with the only difference that we give the project with the only difference that we give sequence number to each process.

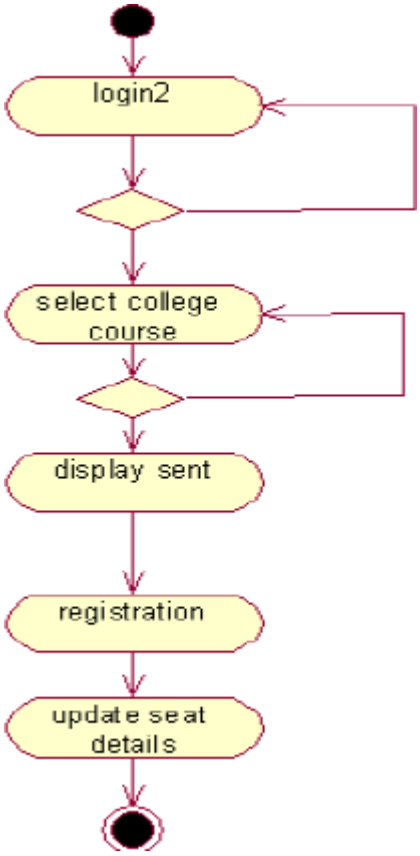

student
1: request for information of the college
3: request for hostel facilities
6: join the college
enquiry
5: accept the hostel facilities
2: provide details for the college
databas
e
7: register the student
4: check for availability


student
1: enter the user name
2: enter the password
3: submit
login
5: incorrect user name or password login again
4: verification
databas
e


1: check for particular course
2: select and college
3: request for seat available
5: no availability to particular course is using
6: request for another college
8: seat availability to the college
student
select
college
7: return availability
4: return available
databas
e

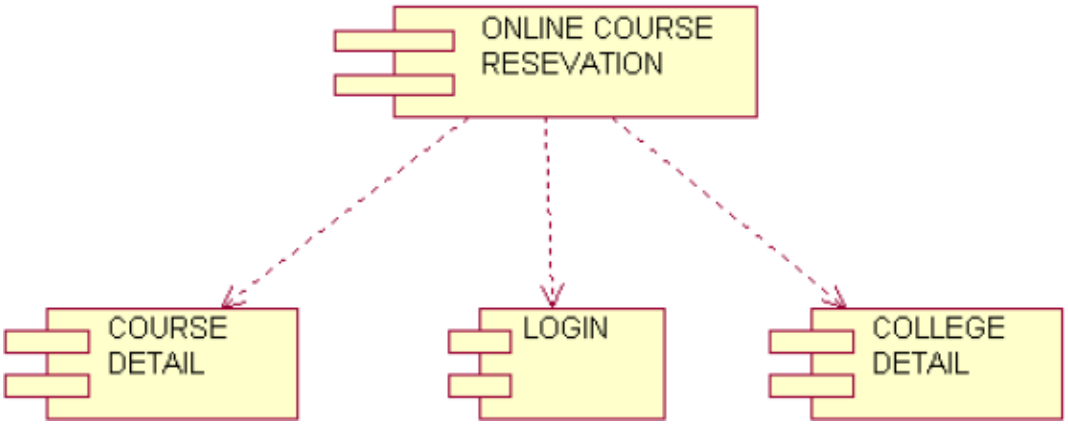
ACTIVIY DIAGRAM

It includes all the activities of particular project and various steps using join and forks



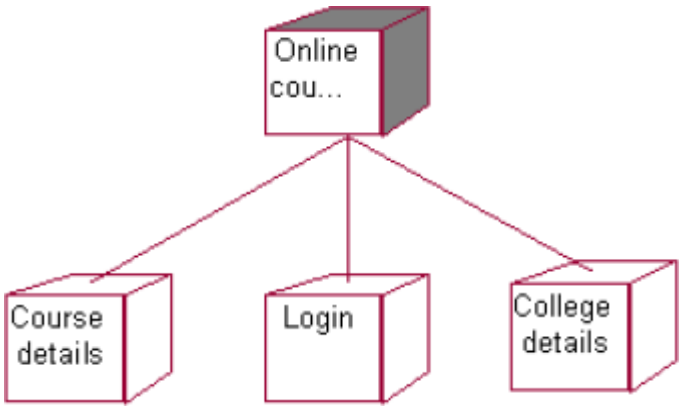
COMPONENT DIAGRAM

The component diagram is represented by figure dependency and it is a graph of design of figure dependency. The component diagram's main purpose is to show the structural relationships between the components of a systems. It is represented by boxed figure. Dependencies are represented by communication association



DEPLOYMENT DIAGRAM

It is a graph of nodes connected by communication association. It is represented by a three dimensional box. A deployment diagram in the unified modeling language serves to model the physical deployment of artifacts on deployment targets. Deployment diagrams show "the allocation of artifacts to nodes according to the Deployments defined between them. It is represented by 3-dimentional box. Dependencies are represented by communication association. The basic element of a deployment diagram is a node of two types



PACKAGE DIAGRAM

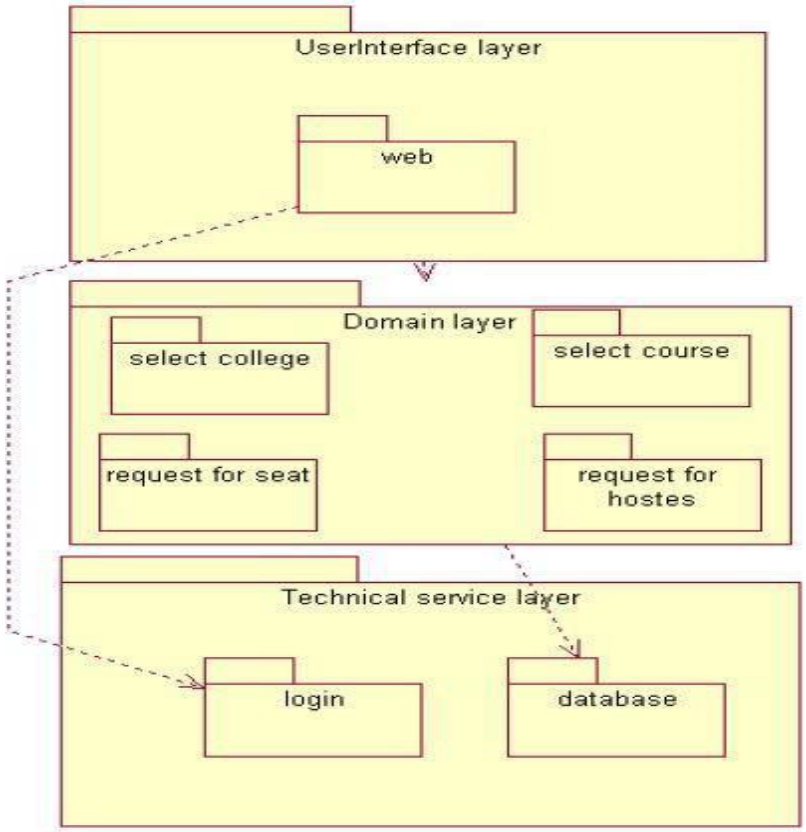
A package diagram is represented as a folder shown as a large rectangle with a top attached to its upper left corner. A package may contain both sub ordinate package and ordinary model elements. All uml models and

diagrams are organized into package. A package diagram in unified modeling language that depicts the dependencies between the packages that make up a model. A Package Diagram (PD) shows a grouping of elements in the OO model, and is a Cradle extension to UML. PDs can be used to show groups of classes in Class Diagrams (CDs), groups of components or processes in Component Diagrams (CPDs), or groups of processors in Deployment Diagrams (DPDs).
There are three types of layer. They are

a. User interface layer

b. Domain layer

c. Technical services layer



RESULT

Thus the mini project for online course reservation system has been successfully executed and codes are generated.

EX NO: 7 AIRLINE/RAILWAY RESERVATION SYSTEM

Date:

AIM

To develop the Airline/Railway reservation System using Rational Rose Software.

(I) PROBLEM ANALYSIS AND PROJECT PLANNING

In the Airline/Railway reservation System the main process is a applicant have to login the database then the database verifies that particular username and password then the user must fill the details about their personal details then selecting the flight and the database books the ticket then send it to the applicant then searching the flight or else cancelling the process.

(II) OVERALL DESCRIPTION

Functionality

The database should be act as an main role of the e-ticketing system it can be booking the ticket in easy way.

Usability

The User interface makes the Credit Card Processing System to be efficient.

Performance

It is of the capacities about which it can perform function for many users at the same times efficiently that are without any error occurrence.

Reliability

The system should be able to process the user for their corresponding request.

(III) USE CASE DIAGRAM

The passenger can view the status of the reserved tickets. So the passenger can confirm his/her travel.

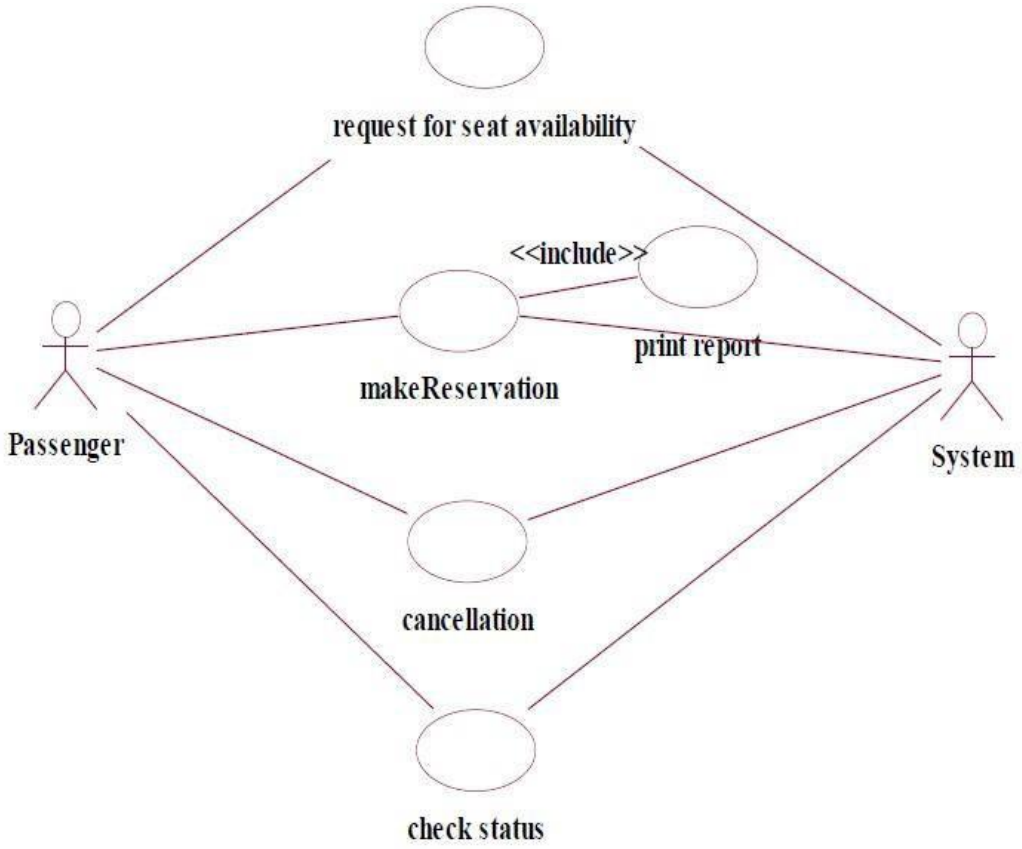
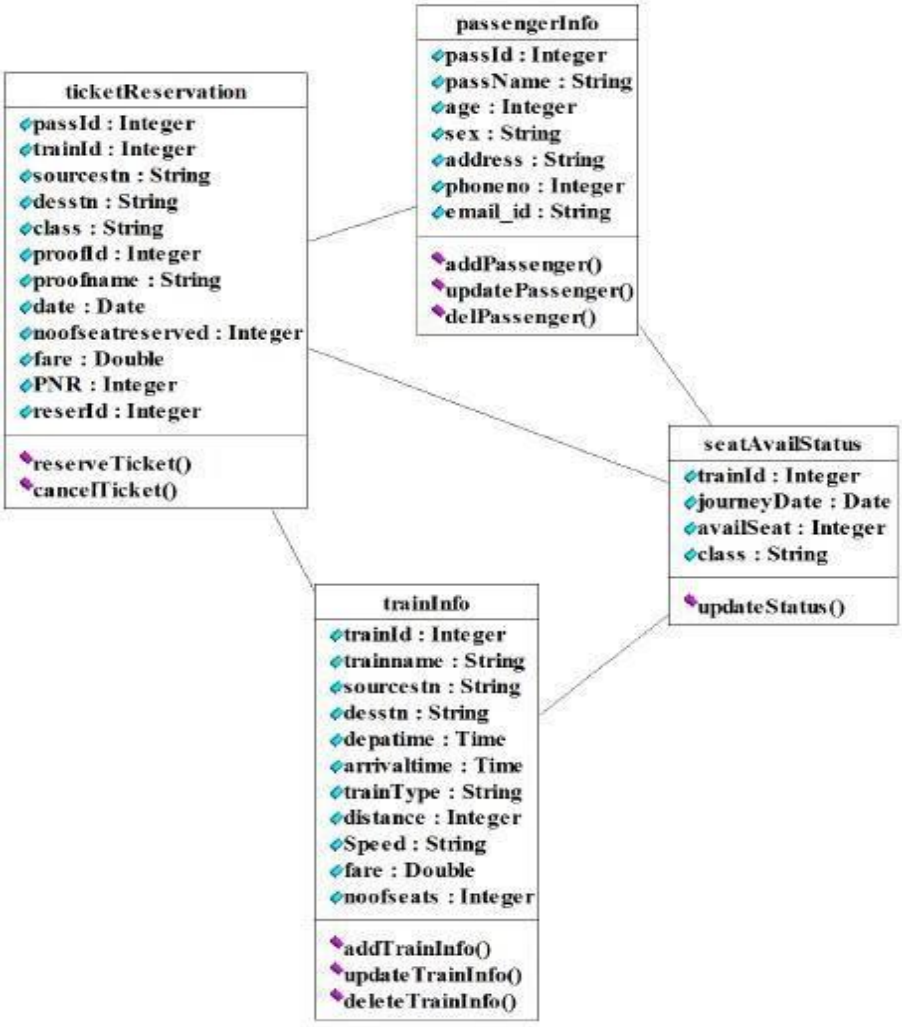


Fig. USE-CASE DIAGRAM FOR AIRLINE RESERVATION

(IV) CLASS DIAGRAM

The online ticket reservation system makes use of the following classes:

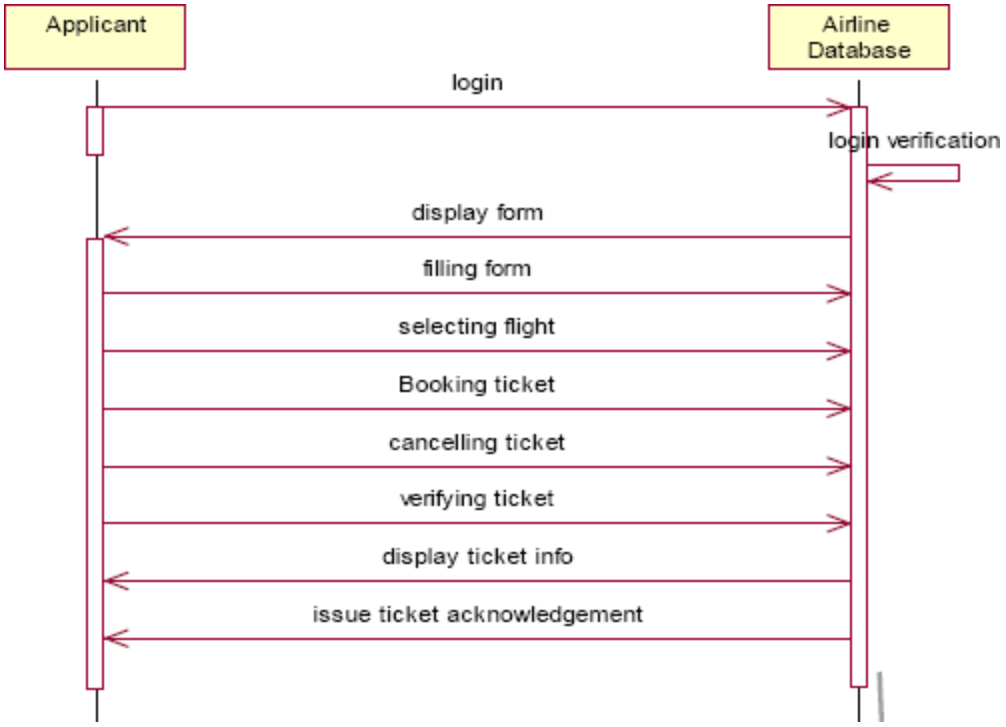
1. ticketReservation
2. trainInfo
3. passengerInfo
4. seatAvailStatus



SEQUENCE DIAGRAM

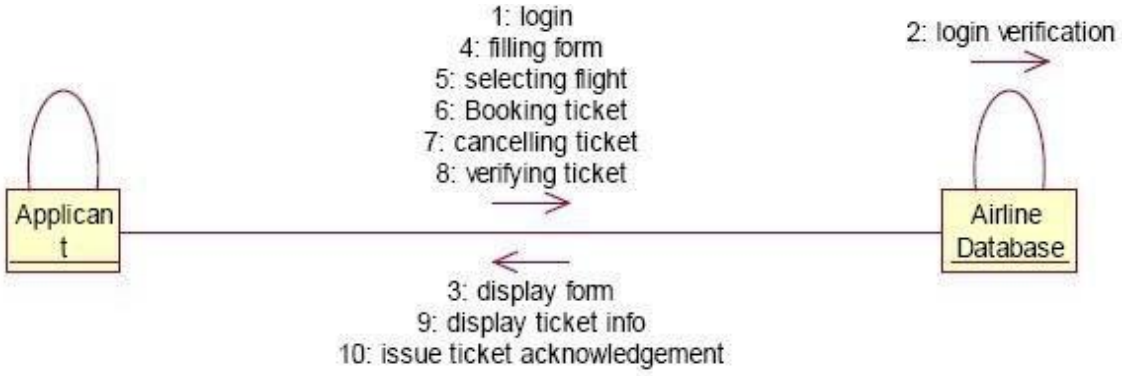
A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. There are two dimensions.

1. Vertical dimension-represent time.
2. Horizontal dimension-represent different objects.



COLLABRATION DIAGRAM

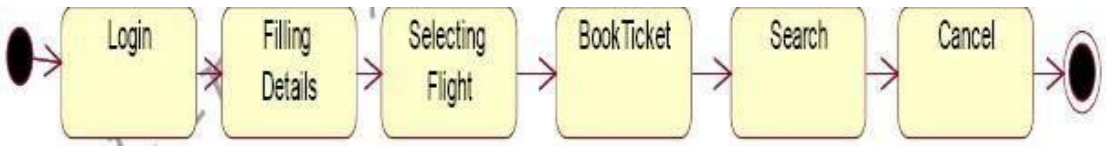
A collaboration diagram, also called a communication diagram or interaction diagram,. A sophisticated modeling tool can easily convert a collaboration diagram into a sequence diagram and the vice versa. A collaboration diagram resembles a flowchart that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in real time.



STATE CHART DIAGRAM

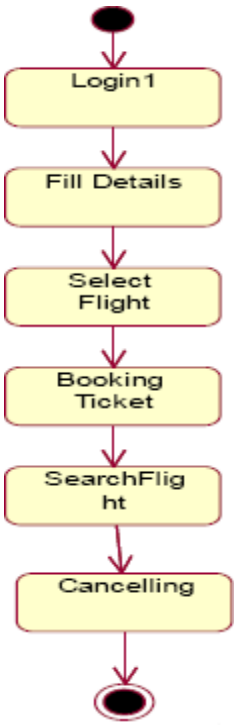
The purpose of state chart diagram is to understand the algorithm involved in performing a method. It is also called as state diagram. A state is

represented as a round box, which may contain one or more compartments. An initial state is represented as small dot. A final state is represented as circle surrounding a small dot.



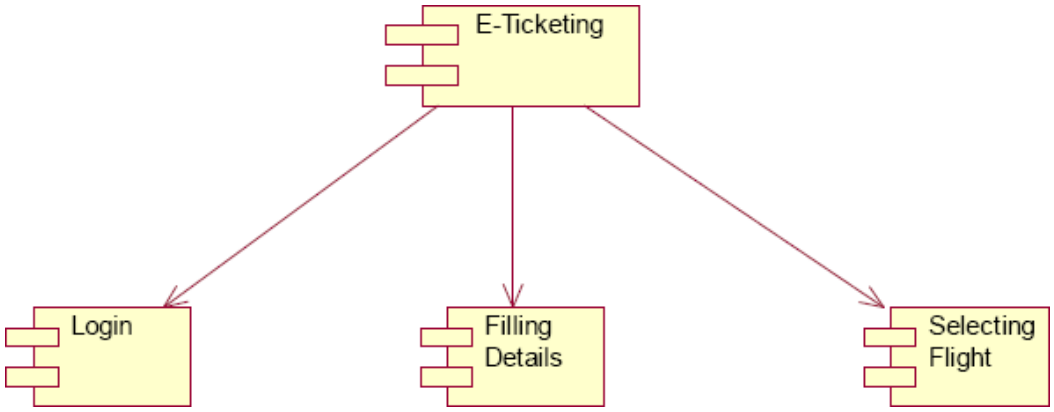
ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control. An activity is shown as an rounded box containing the name of the operation.



COMPONENT DIAGRAM

The component diagram's main purpose is to show the structural relationships between the components of a system. It is represented by boxed figure. Dependencies are represented by communication association.



RESULT

Thus the mini project for Airline/Railway reservation System has been successfully executed and codes are generated.

Ex no: 8 SOFTWARE PERSONNEL MANAGEMENT SYSTEM

Date:

AIM:

To implement a software for software personnel management system.

(I) PROBLEM STATEMENT:

Human Resource management system project involves new and/or system upgrades of software of send to capture information relating to the hiring termination payment and management of employee. He uses system to plan and analyze all components and performance of metrics driven human resource functions, including recruitment, attendance, compensation, benefits and education. Human resources management systems should align for maximum operating efficiency with financial accounting operations customer relationship management,security and business lines as organization.

(II)SOFTWARE REQUIREMENT SPECIFICATION:

2.1SOFTWARE INTERFACE

- **Front End Client** - The applicant and Administrator online interface is built using

  JSP and HTML. The HR's local interface is built using Java.

- **Server** - Glassfish application server(SQL Corporation).
- **Back End** - SQL database.

2.2HARDWARE INTERFACE

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

(III)USECASE DIAGRAM:

The HR of an organization involves recruitment training, monitoring and motivation of an employee. The HR also involves gives salary as observed in the payroll sheet. The employee undergoes training, receives the salary , gives the expected performance and manages time in order to complete a given task within the required period.

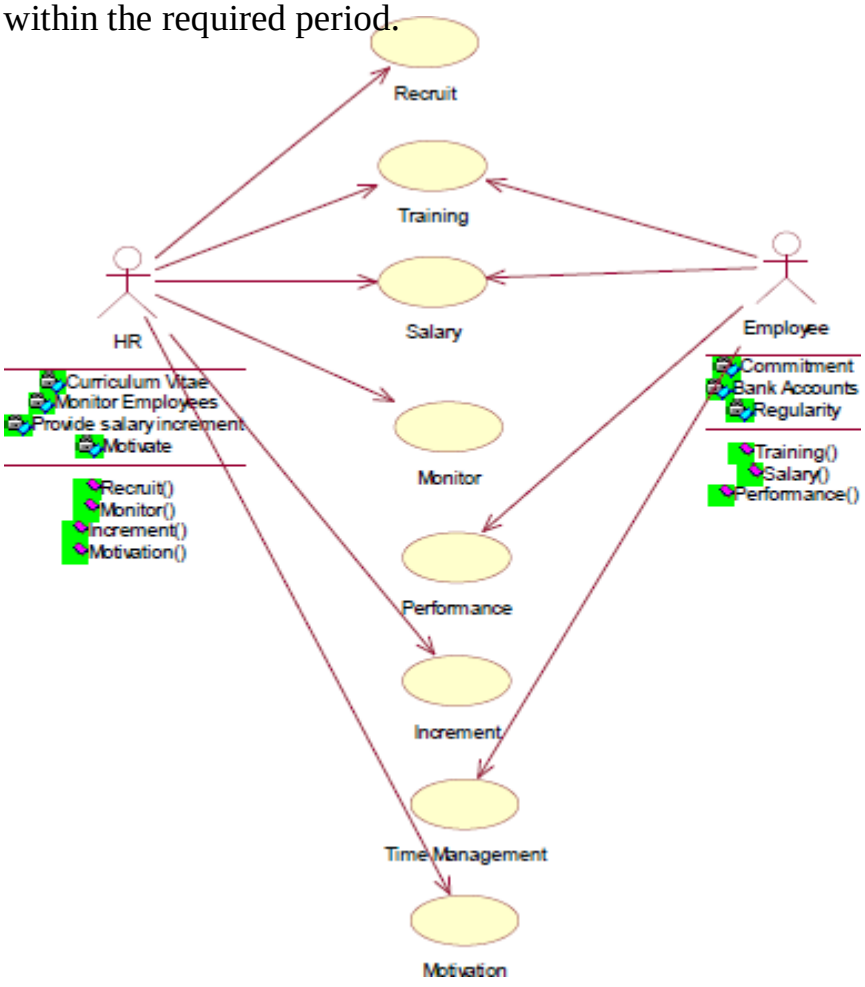


Fig.3. USE CASE DIAGRAM

(IV) ACTIVITY DIAGRAM:

The activity diagram notation is an action, partition, fork join and object node. Most of the notation is self explanatory, two subtle points. Once an action finished, there is an automatic outgoing transaction. The diagram can show both control flow and data flow.

Fig.4. ACTIVITY DIAGRAM

(V) CLASS DIAGRAM:

The class diagram, also referred to as object modeling is the main static analysis diagram. The main task of object modeling is to graphically show what each object will do in the problem domain. The problem domain describes the structure and the relationships among objects.

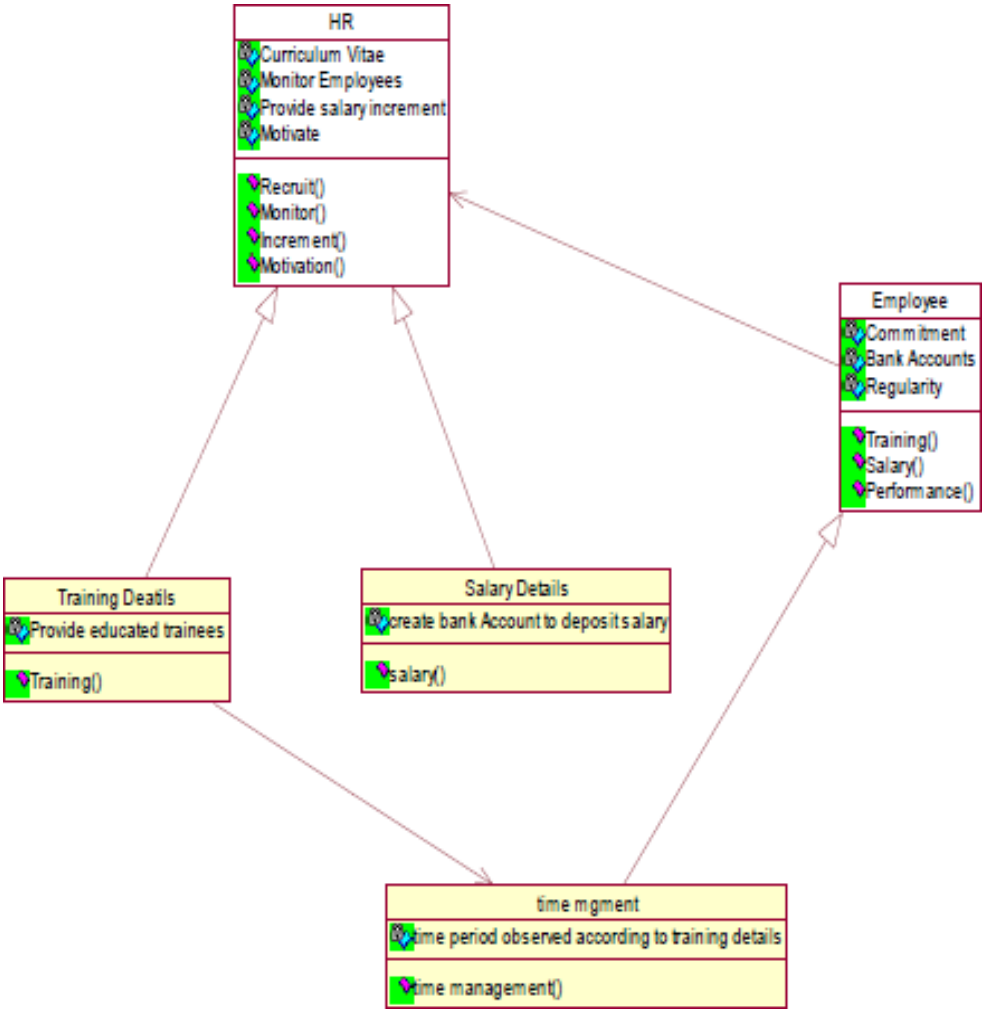


Fig.5.CLASS DIAGRAM

(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.

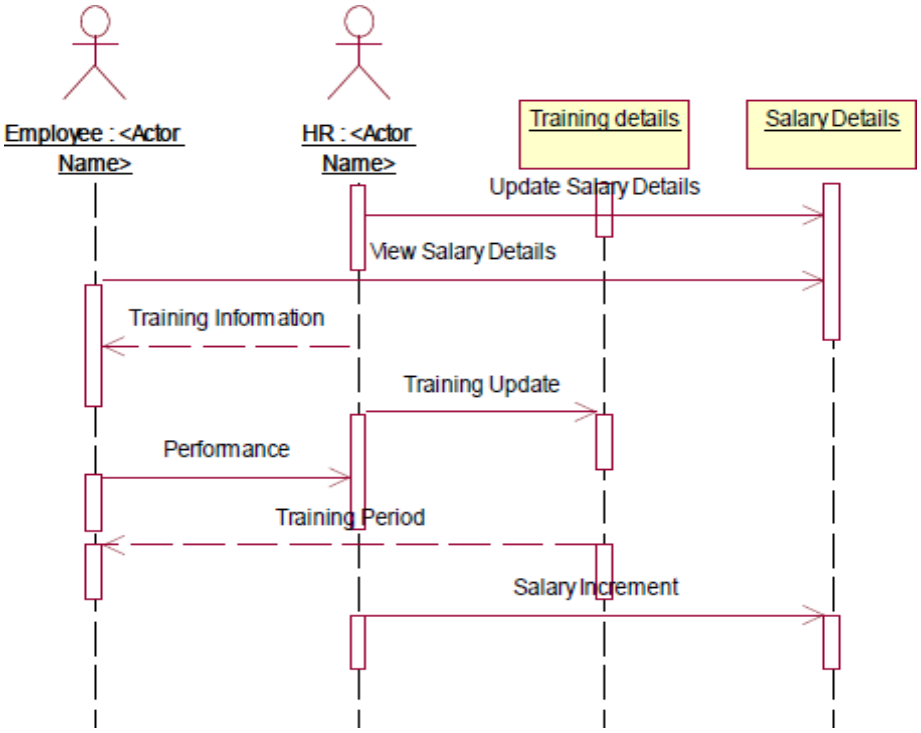


Fig.6.1.SEQUENCE DIAGRAM

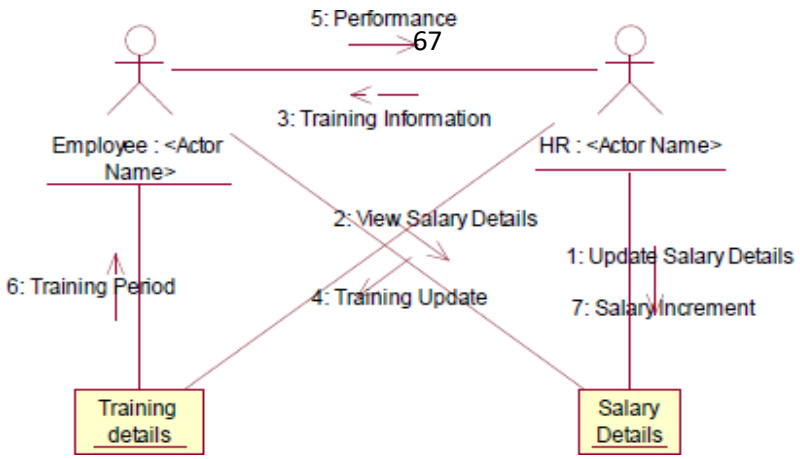


Fig.6.2.COLLABORATION DIAGRAM

(VII)) State Transition Diagram

States of object are represented as rectangle with round corner, the transaction between the different states. A transition is a relationship between two state that indicates that when an event occur the object moves from the prior state to the subsequent.

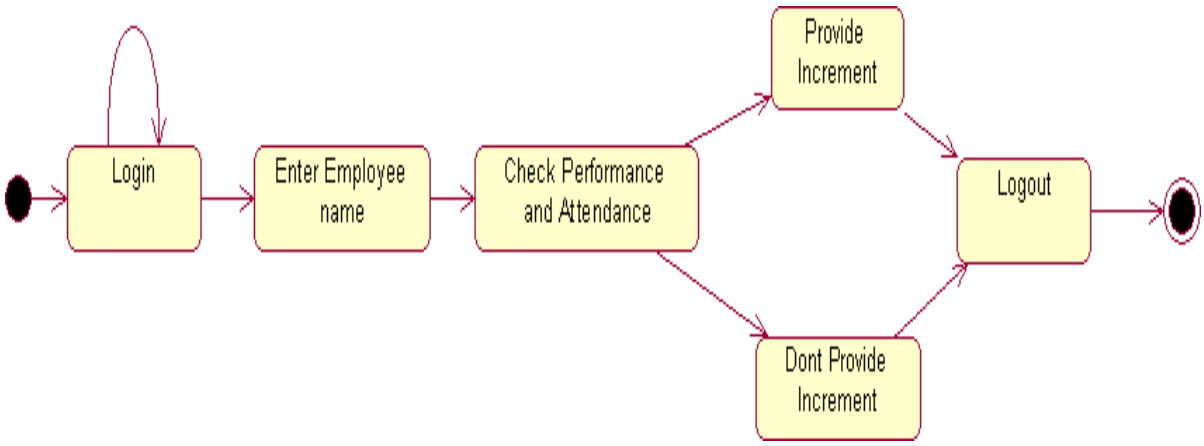


Fig.7.STATE TRANSITION DIAGRAM

(VIII) DEPLOYMENT DIAGRAM AND COMPONENT DIAGRAM

HR recruits employee for a company employee recruited by HR goes under training before actually working. Training period is given to the employee with the training details. The salary details for the employee are provided.

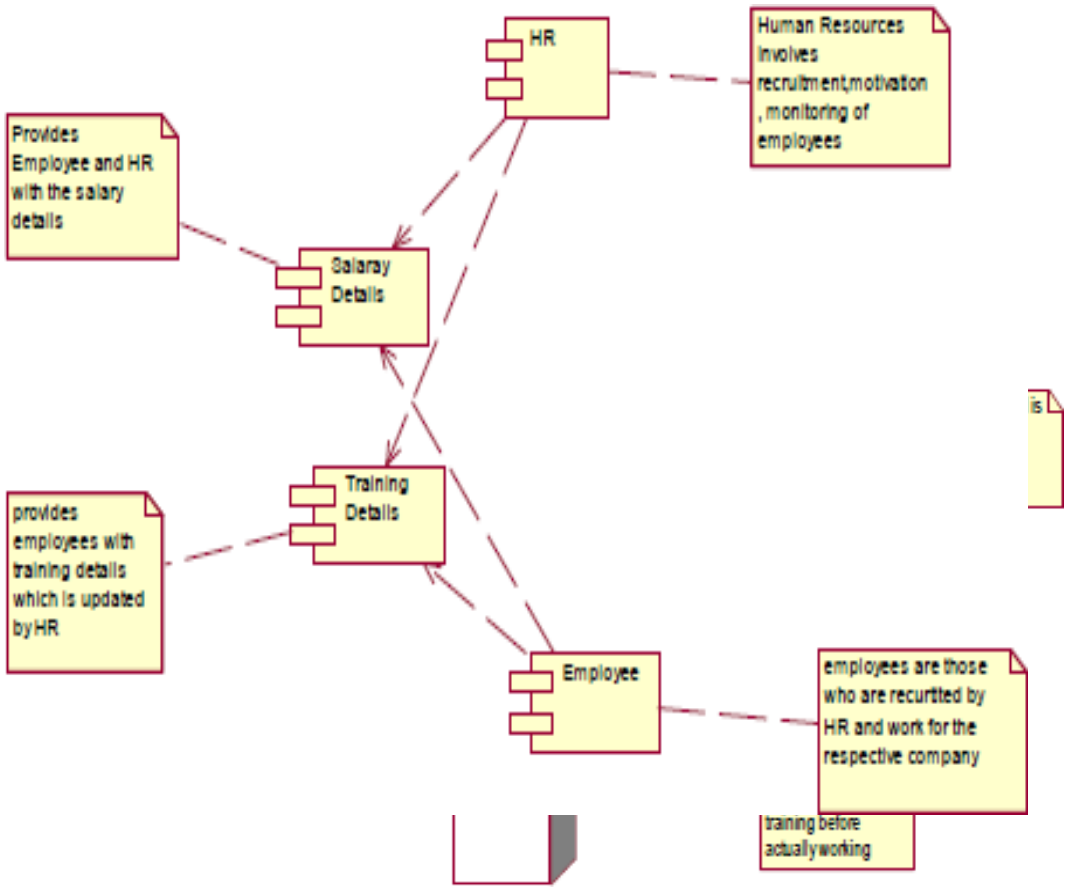


Fig.8.1.DEPLOYMENT DIAGRAM

COMPONENT DIAGRAM

The HR recruits, motivate and monitor the employee, HR also update

the salary details and training details for reference. The employee are those who are recruited by HR and work for the company. The training details provide employees with training details which is updated by HR

Fig.8.2.COMPONENT DIAGRAM

RESULT:

Thus the mini project for software personnel management system has been successfully executed and codes are generated.

Ex. No:9

CREDIT CARD PROCESSING

Date:

AIM:

To create a system to perform the credit card processing

(I) PROBLEM STATEMENT:

Credit card processing through offline involves the merchant collecting order information (including credit card numbers), storing this in a database on your site, and entering it using their on-site merchant credit card processing system. Takes time to manually enter credit card information for each order. This solution creates following cons:

(II)SOFTWARE REQUIREMENT SPECIFICATION:

PRODUCT PERSPECTIVE

This solution involves signing up for a free Business Account. Once this is done and the e-commerce site is properly configured, you can accept payments from Visa, MasterCard, Amex, and Discover cards payments.

SOFTWARE INTERFACE

- **Front End Client** - The applicant and Administrator online interface is built using

  JSP and HTML. The Administrators's local interface is built using Java.

- **Web Server** - Glassfish application server(SQL Corporation).
- **Back End** - SQL database.

HARDWARE INTERFACE

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

(III)USECASE DIAGRAM:

USE-CASE NAME: PAYMENT APPROVAL

The transaction details are recorded by the credit card processor and results are securely relayed to the merchant. Merchant's site receives transaction result and does appropriate actions (e.g. saves the order & shows message).

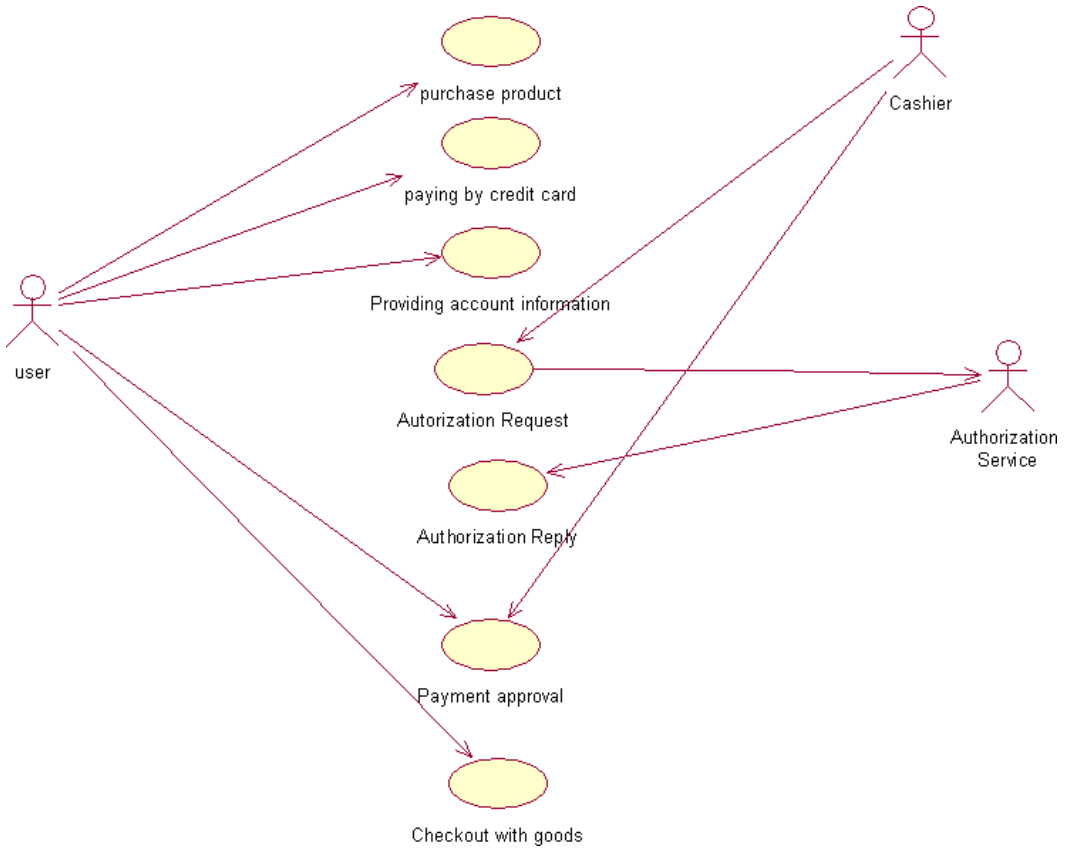


Fig.3. USECASE DIAGRAM FOR PASSPORT AUTOMATION SYSTEM

(IV) CLASS DIAGRAM:

The class diagram, also referred to as object modeling is the main static analysis diagram. The main task of object modeling is to graphically show what each object will do in the problem domain. The problem domain describes the structure and the relationships among objects.

The Credit Card Processing system class diagram consists of three classes.

They are

Cashier

User

Authorization Service

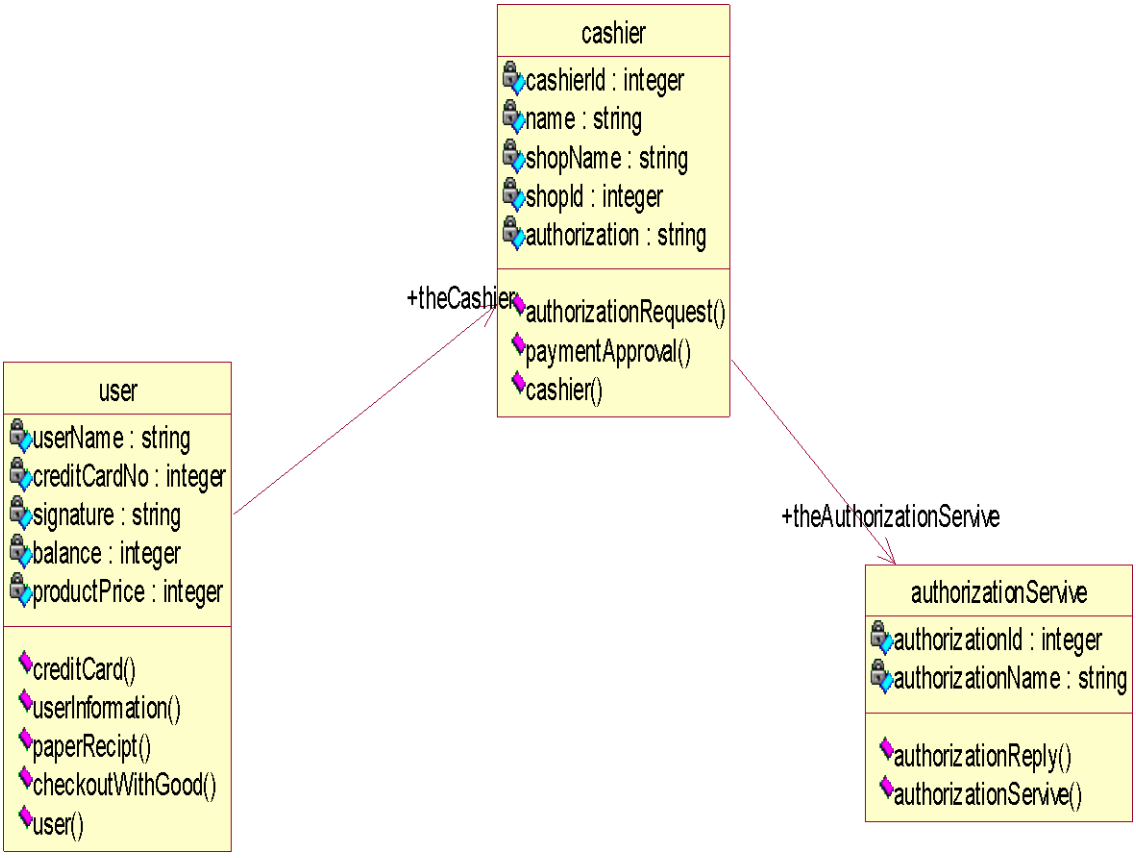


Fig.4.CLASS DIAGRAM

(V)) INTERACTION DIAGRAM:

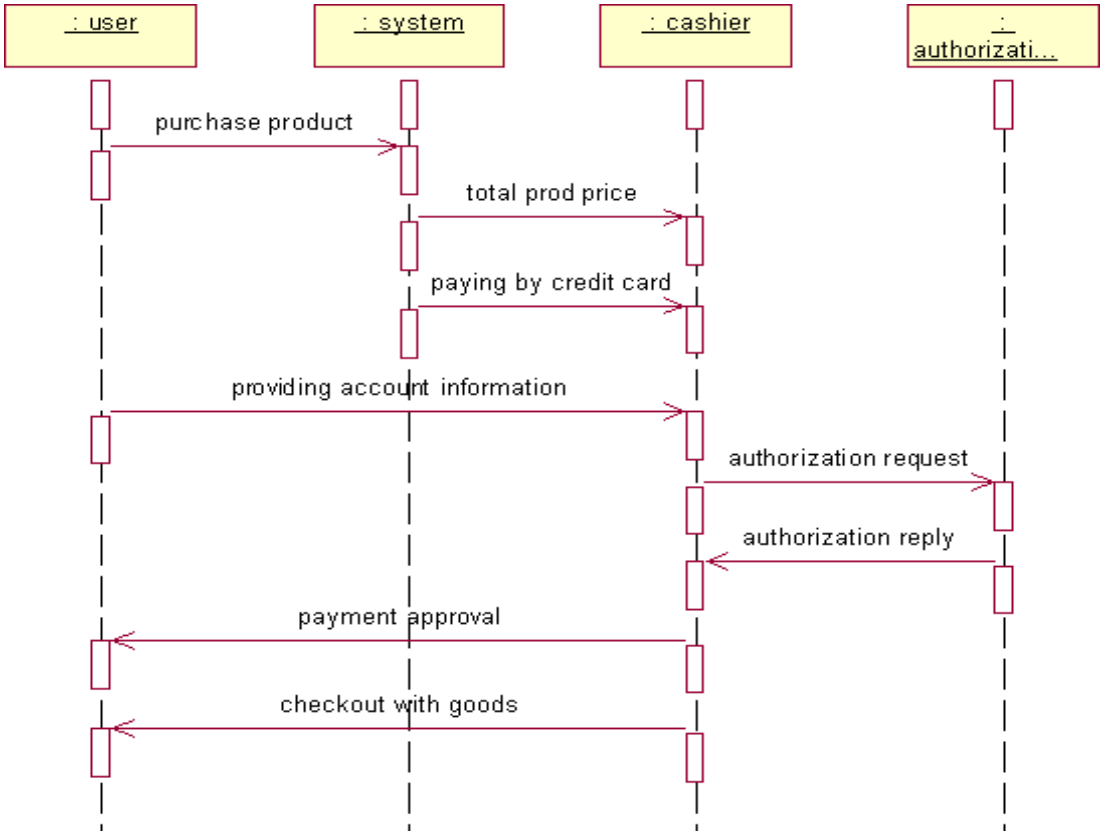


Fig.5.1.SEQUENCE DIAGRAM

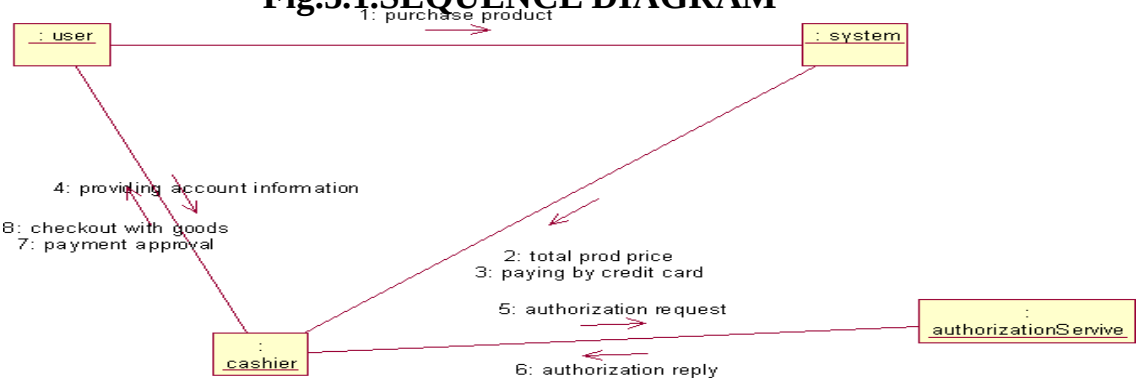
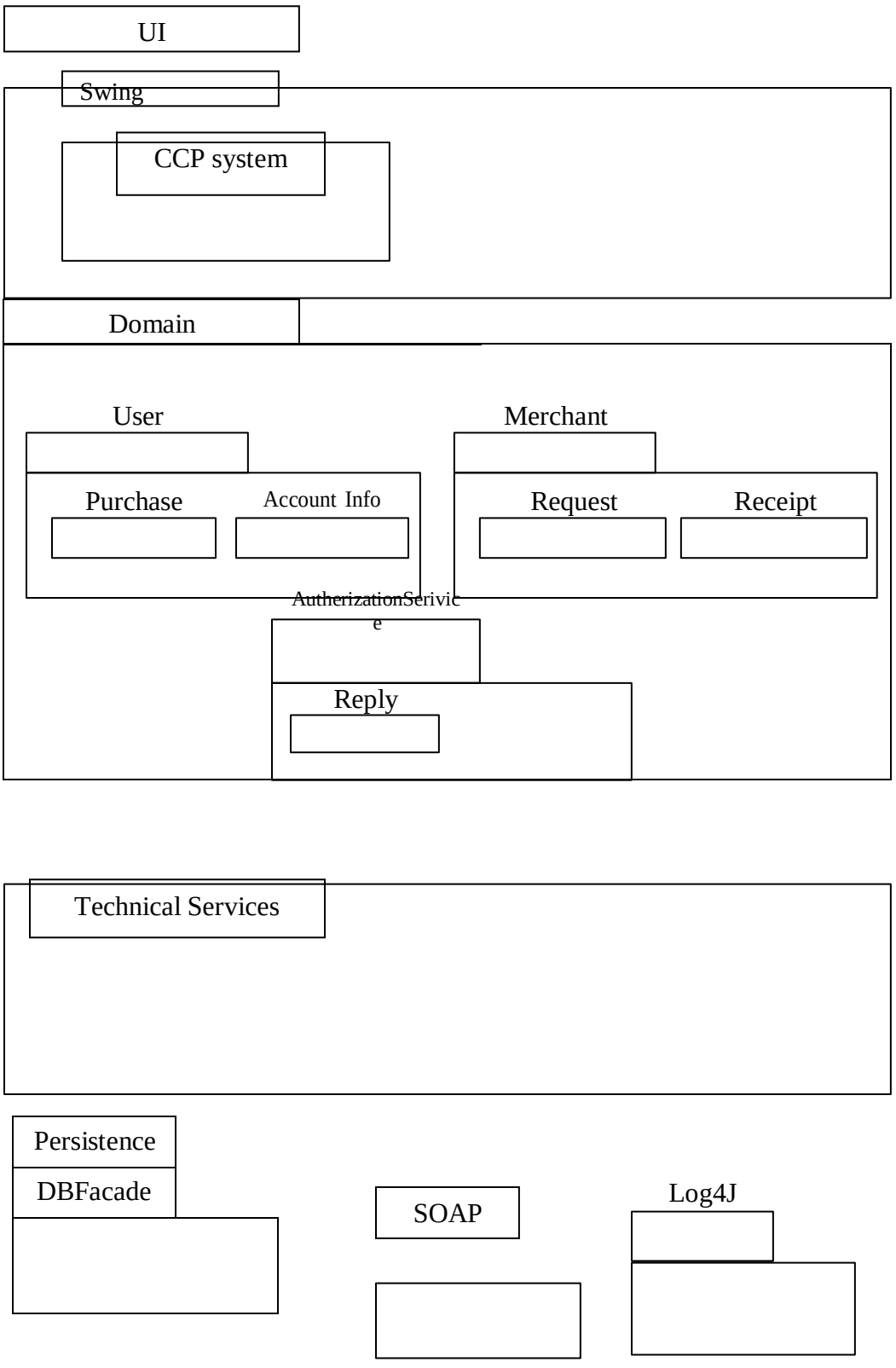


Fig.5.2.COLLABORATION DIAGRAM

(VI) PARTIAL LAYERD LOGICAL ARCHITECTURE DIAGRAM



(VII) 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.

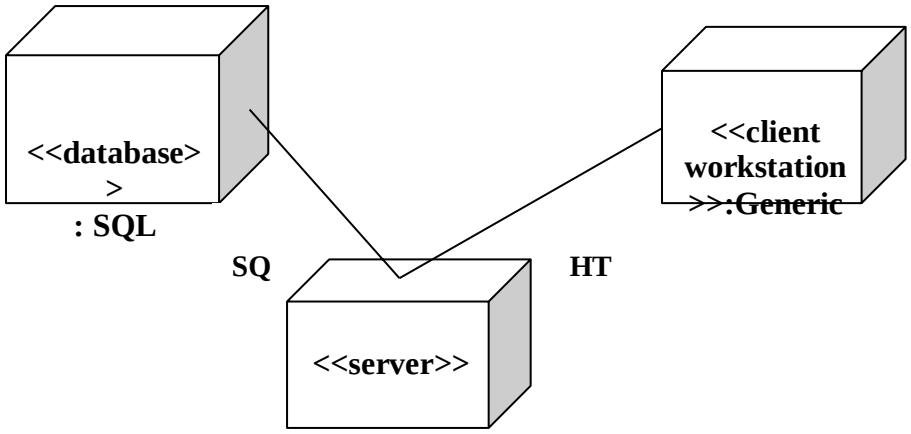


Fig.7.1.DEPLOYMENT DIAGRAM

COMPONENT DIAGRAM

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

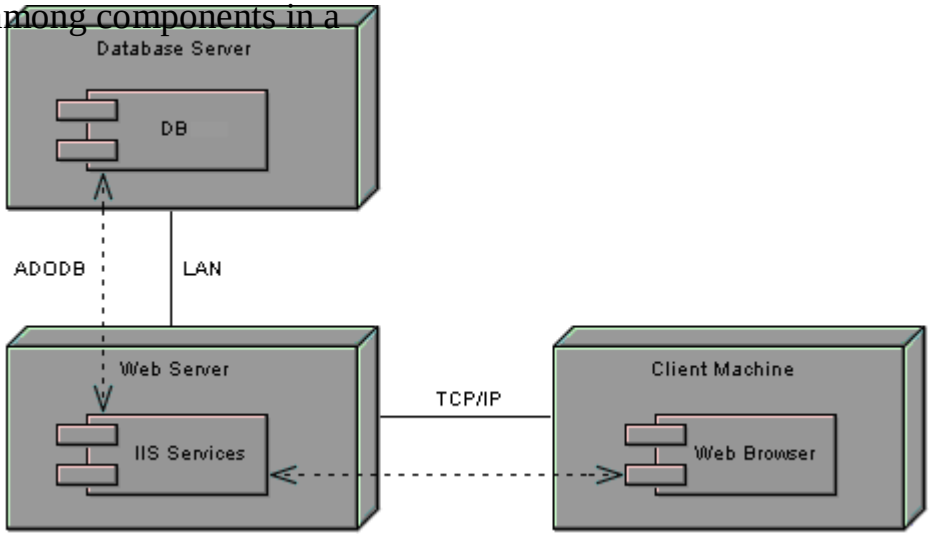


Fig.7.2.COMPONENT DIAGRAM

RESULT:

Thus the mini project for credit card processing system has been successfully executed and codes are generated.

Ex. No:10

OOK MANAGEMENT SYSTEM

Date:

AIM:

To create a system to perform E- book Management System.

(I) PROBLEM STATEMENT:

An E- Book lends books and magazines to member, who is registered in the system. Also it handles the purchase of new titles for the Book Bank. Popular titles are brought into multiple copies. Old books and magazines are removed when they are out or date or poor in condition. A member can reserve a book or magazine that is not currently available in the book bank, so that when it is returned or purchased by the book bank, that person is notified. The book bank can easily create, replace and delete information about the tiles, members, loans and reservations from the system.

(II) SOFTWARE RESOURCE SPECIFICATION:

OVERALL DESCRIPTION

Itwill describe major role of the system components and inter-connections.

PRODUCT PERSPECTIVE

The ORS acts as an interface between the user and the 'e-book manager'. 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 books or magazines.

SOFTWARE INTERFACE

Front End Client - The Student and Librarian online interface is built using JSP and HTML. The Librarians local interface is built using Java.

Web Server - Glassfish application server (Oracle Corporation).

Back End - Oracle database

HARDWARE INTERFACE

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

(III) USE-CASE DIAGRAM:

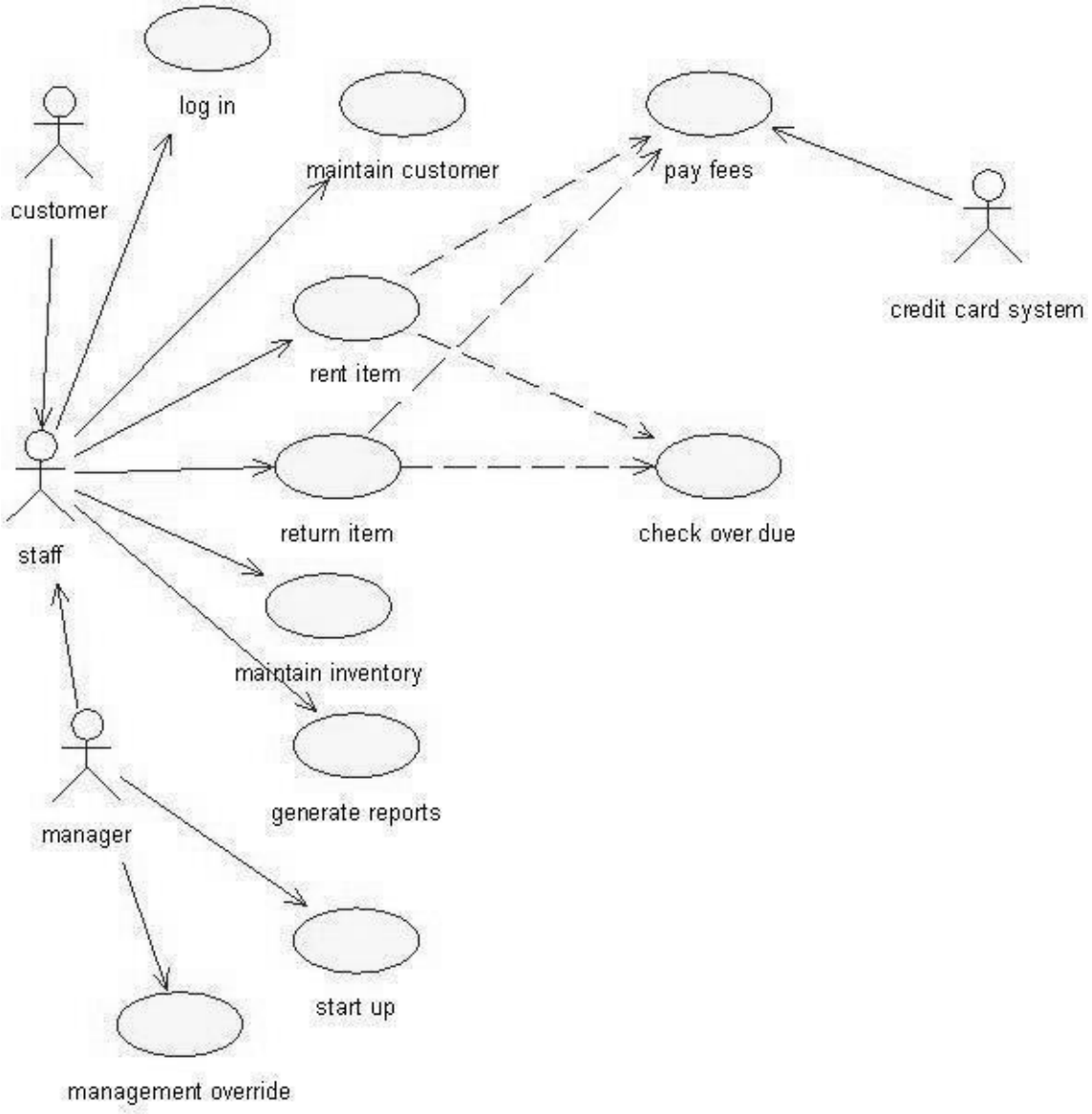
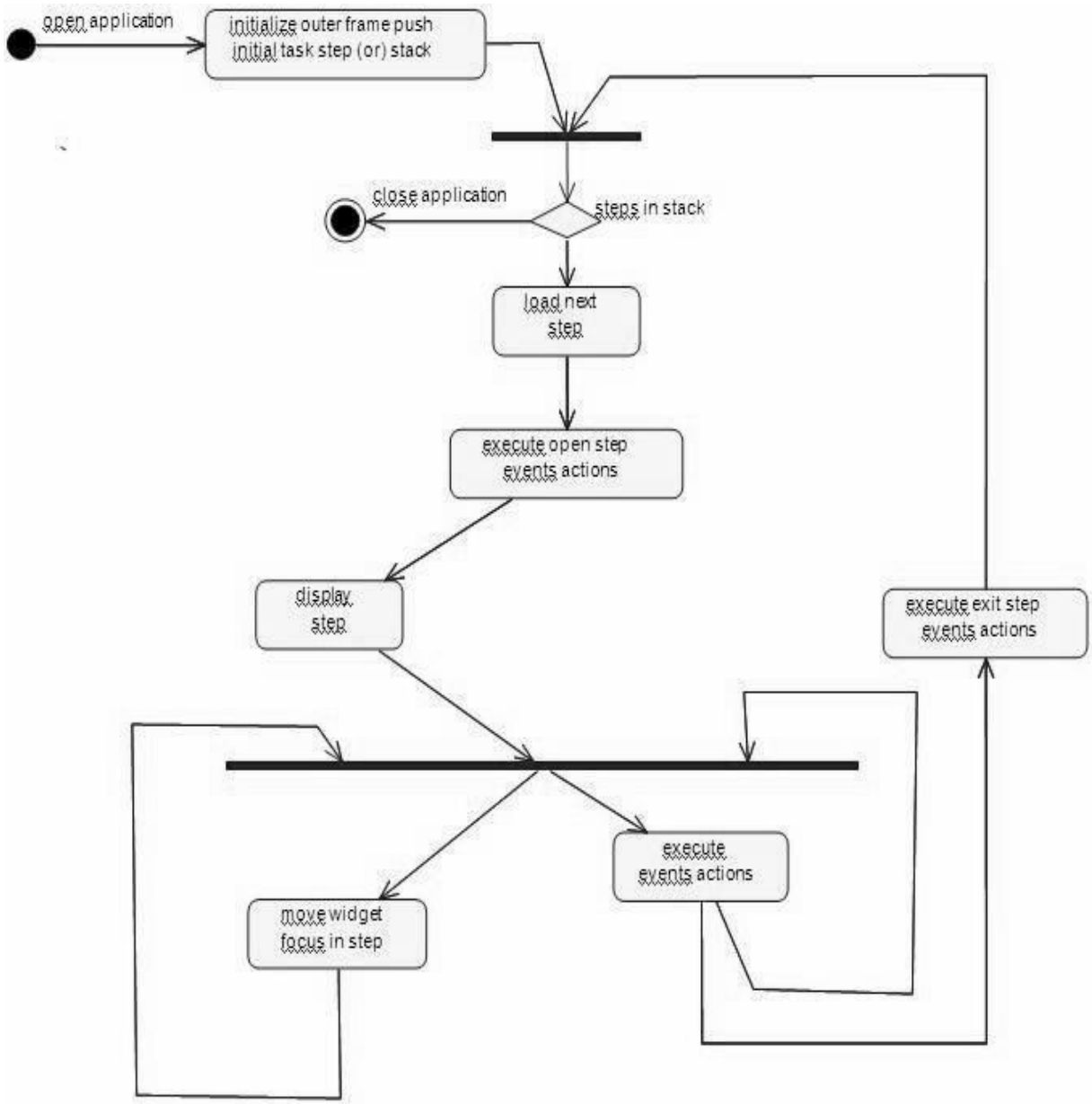


Fig.3.USE-CASE DIAGRAM FOR E-BOOK SYSTEM

(IV) ACTIVITY DIAGRAM:



(V) CLASS DIAGRAM

The class diagram, also referred to as object modeling is the main static analysis diagram. The main task of object modeling is to graphically show what each object will do in the problem domain. The problem domain describes the structure and the relationships among objects.

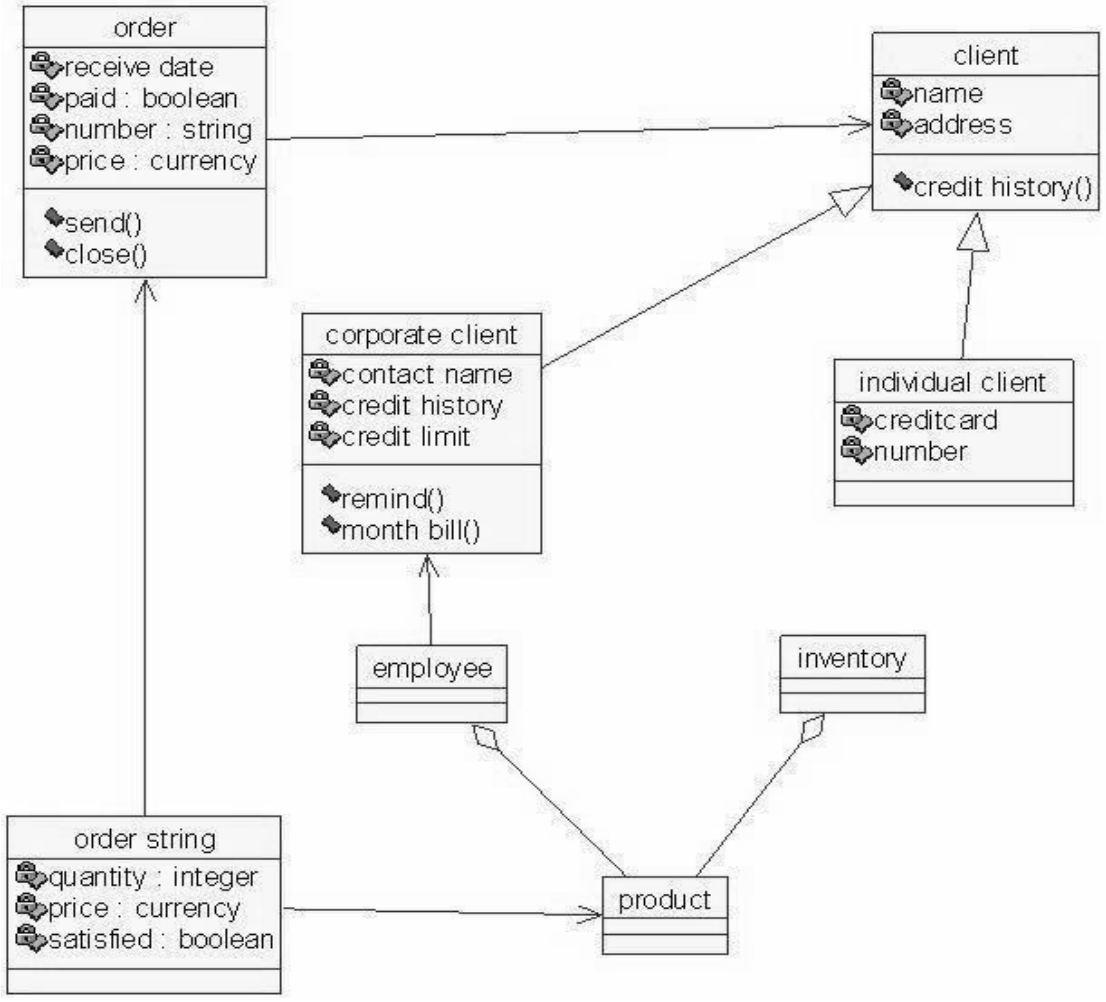


Fig.5.CLASS DIAGRAM FOR E-BOOK SYSTEM

(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 "form" 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 and collaboration diagram are given below.

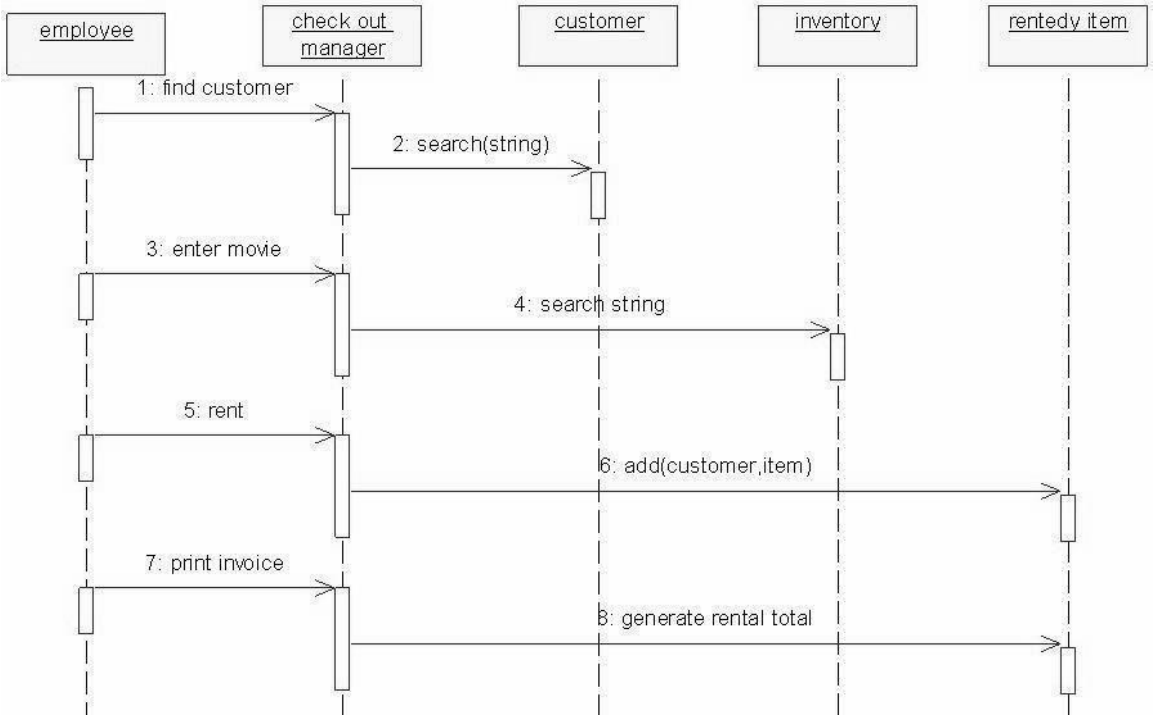


Fig.6.1.SEQUENCE DIAGRAM

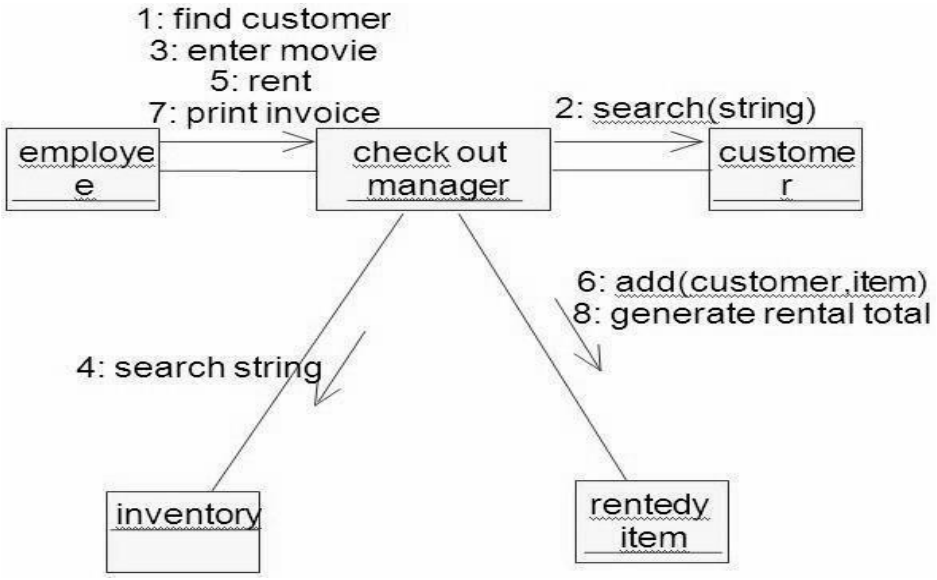
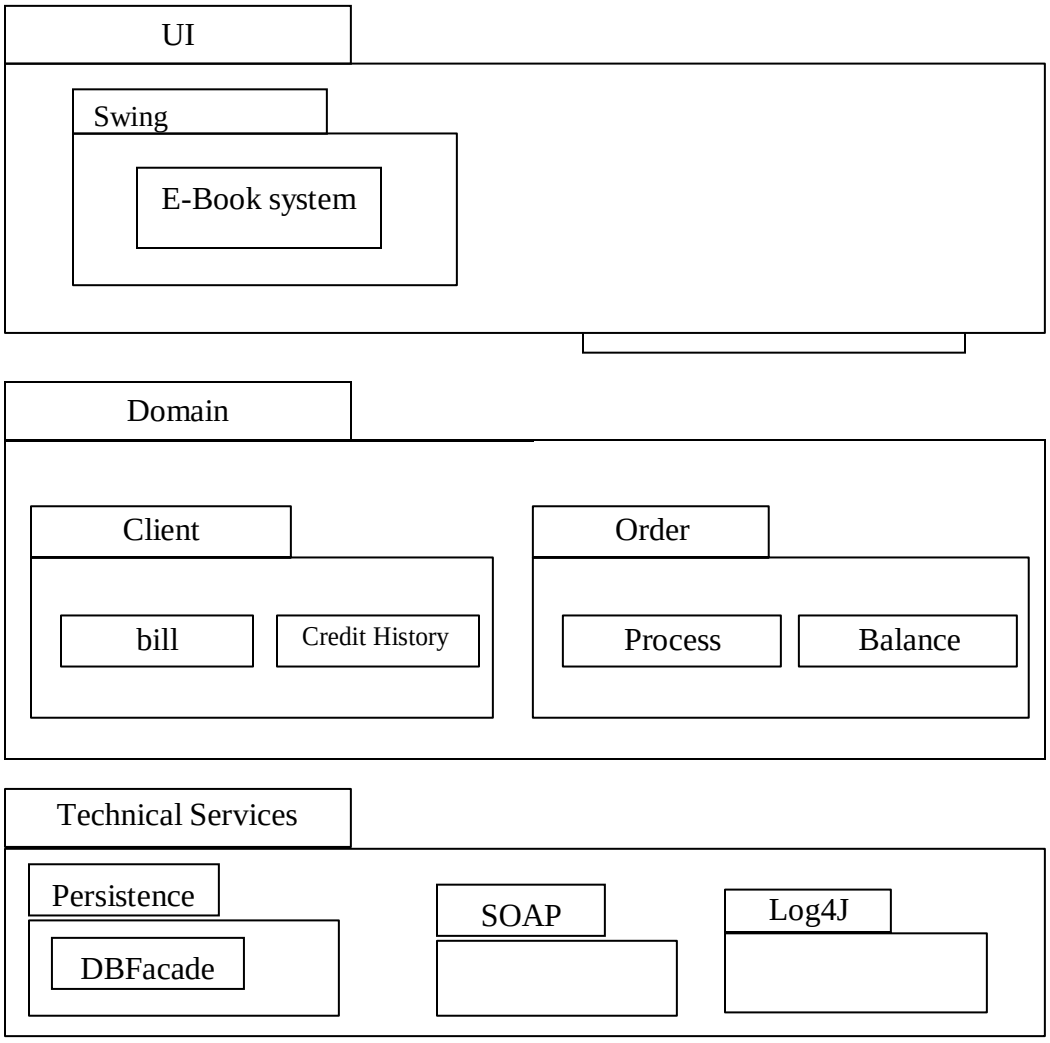


Fig.6.2.COLLABORATION DIAGRAM

(VII) PARTIAL LAYERD LOGICAL ARCHITECTURE 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.

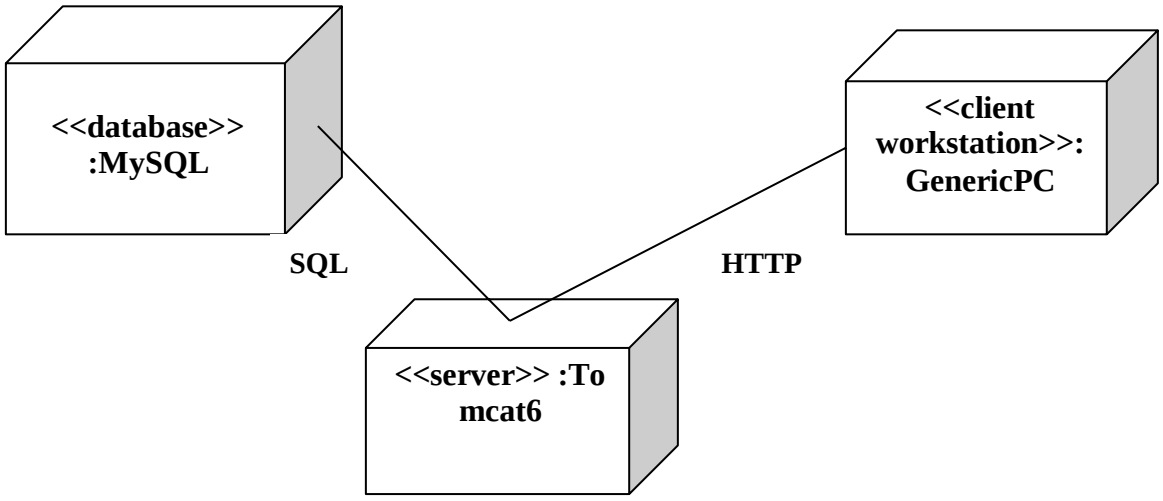


Fig.8.1.DEPLOYMENT DIAGRAM

COMPONENT DIAGRAM

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

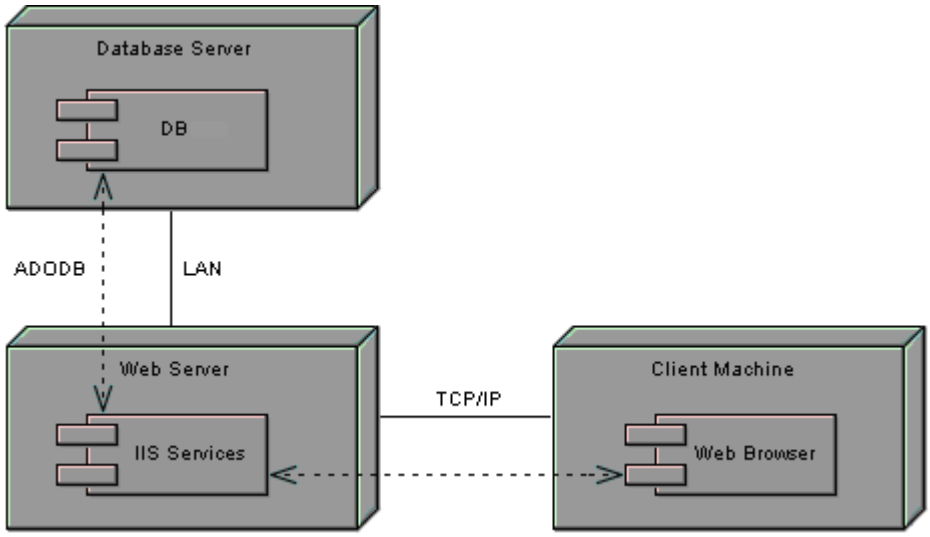


Fig.8.2.COMPONENT DIAGRAM

RESULT:

Thus the mini project for E-Book System has been successfully executed and codes are generated.

Ex.No: 11

RECRUITMENT SYSTEM

Date:

AIM:

To create an automated system to perform the Recruitment System Process.

(I) PROBLEM STATEMENT:

The recruitment system allows the job seekers to enroll their names through the process of registration. The employee also can get the list of available candidates and shortlist for their company requirement. Once the applicant enrolls he receives an id, which helps him in further Correspondence. A fees amount is received from the job seekers for enrollment. This system makes the task of the job seeker easier rather than waiting in queue for enrollment. This also reduces the time consumption for both for the job seeker and employee.

(II) SOFTWARE REQUIREMENT SPECIFICATION:

PRODUCT PERSPECTIVE

The PAS acts as an interface between the 'applicant' and the 'administrator'. 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 recruitment.

SOFTWARE INTERFACE

- **Front End Client** - The applicant and Administrator online interface is built using JSP and HTML. The Administrators's local interface is built using Java.
- **Web Server** - Glassfish application server (SQL Corporation).

• **Back End** - SQL database.

HARDWARE INTERFACE

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

(III)USECASE DIAGRAM:

The Recruitment Automation system use cases are:

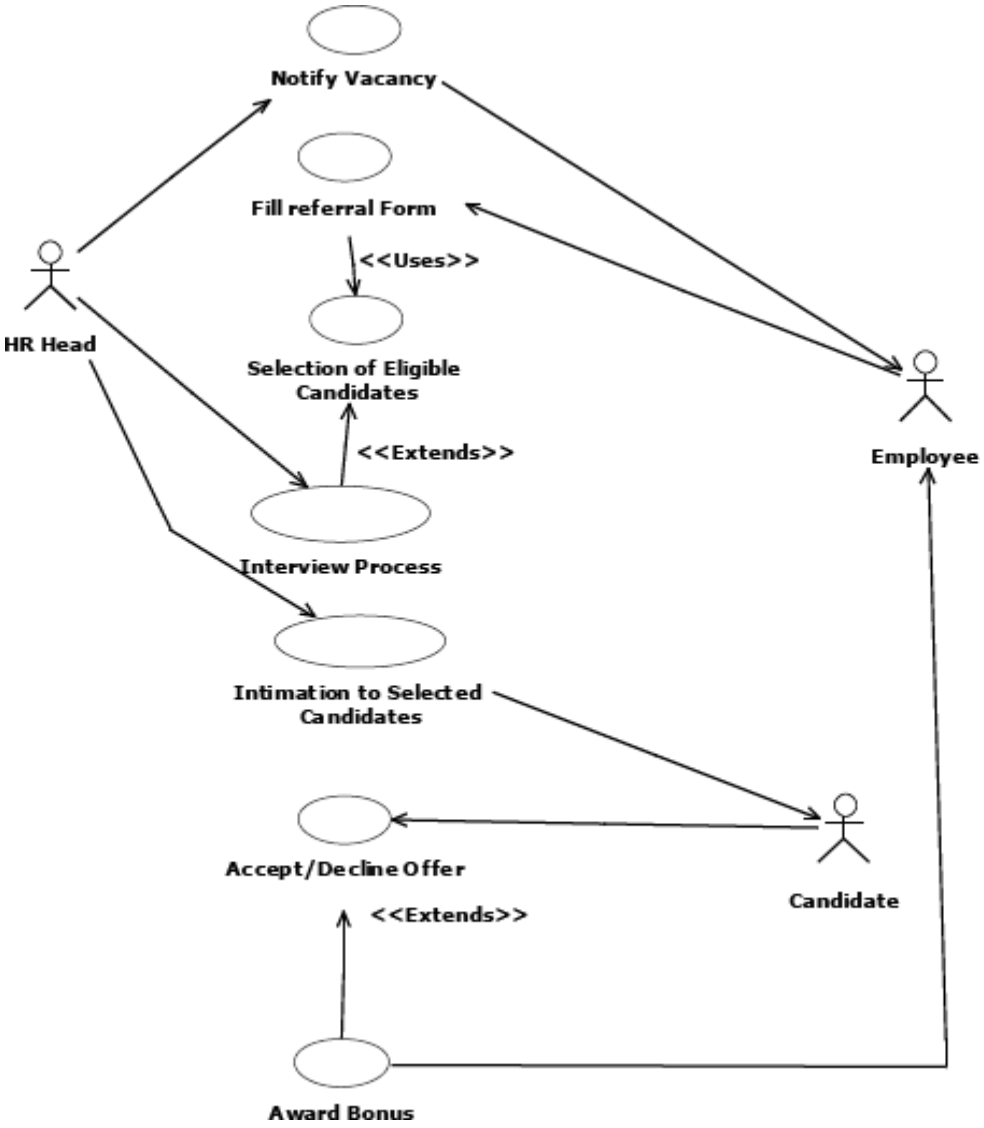


Fig.3. UML USE CASE DIAGRAM

(IV) ACTIVITY DIAGRAM:

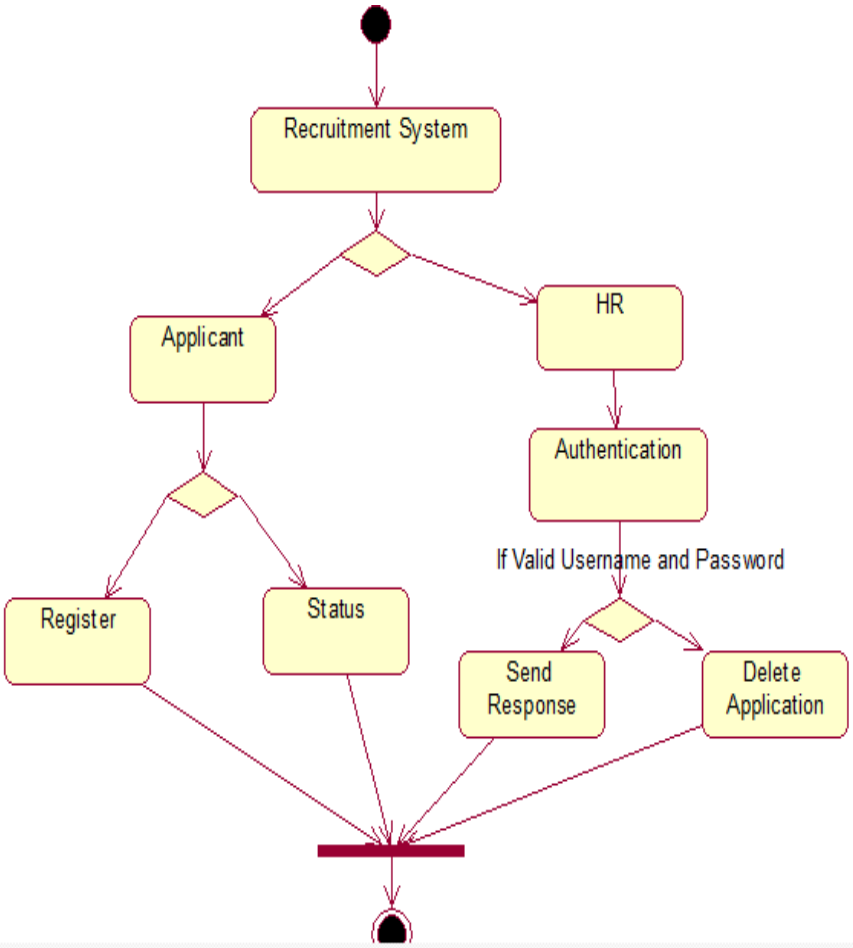


Fig.4. ACTIVITY DIAGRAM

(V) UML CLASS DIAGRAM:

The UML class diagram is to illustrate class interfaces and their actions. They are used for static object modeling, we have already introduced and used their UML diagram while domain modeling.

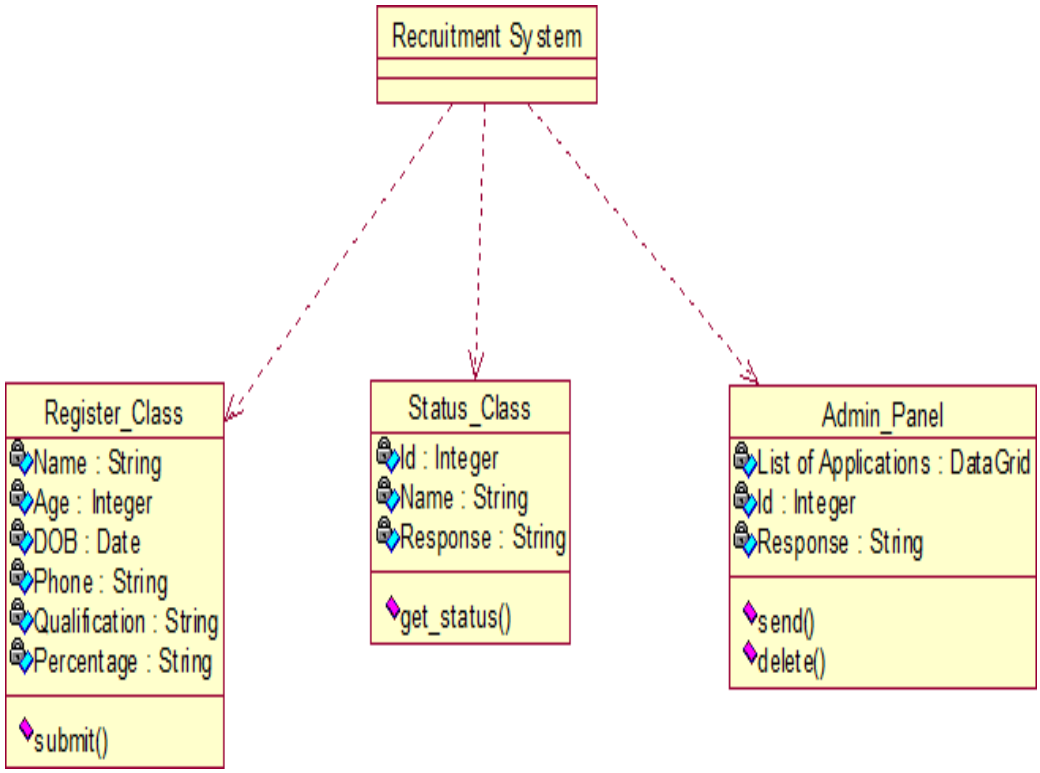


Fig.5. UML CLASS DIAGRAM

(VI) UML SEQUENCE DIAGRAM:

A sequence diagram illustrates a kind of format in which each object interacts via message. It is generalize between two or more specialized diagram.

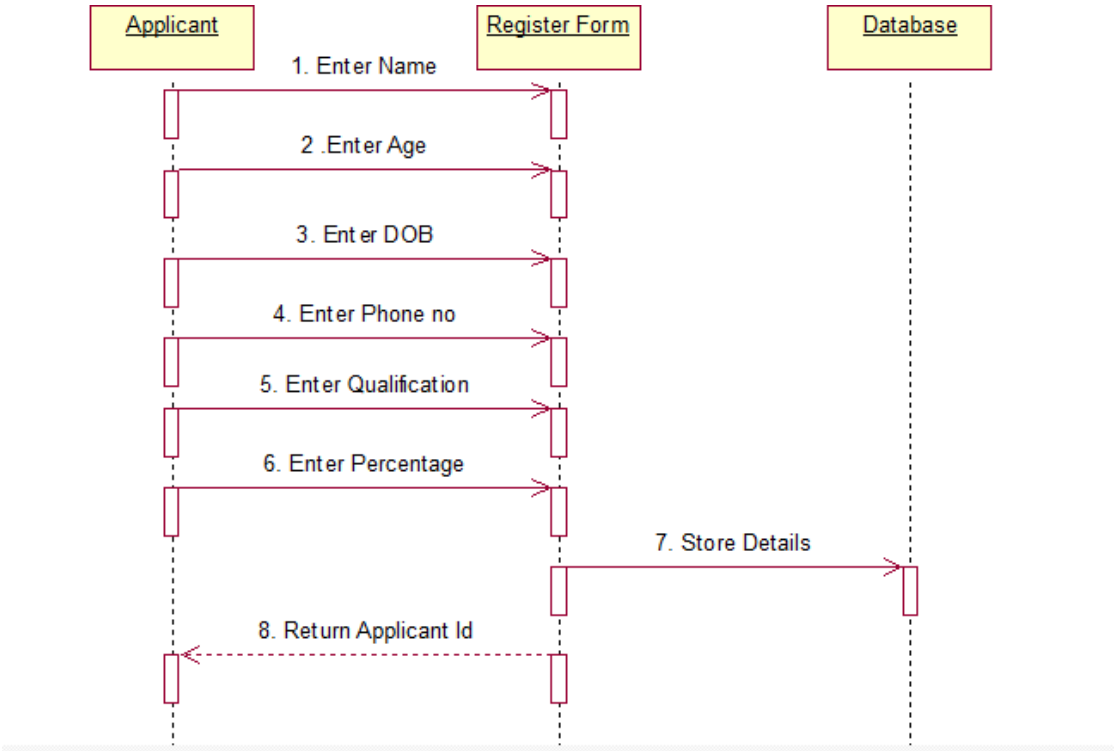


Fig. 6.1SEQEUENCE DIAGRAM FOR Register:

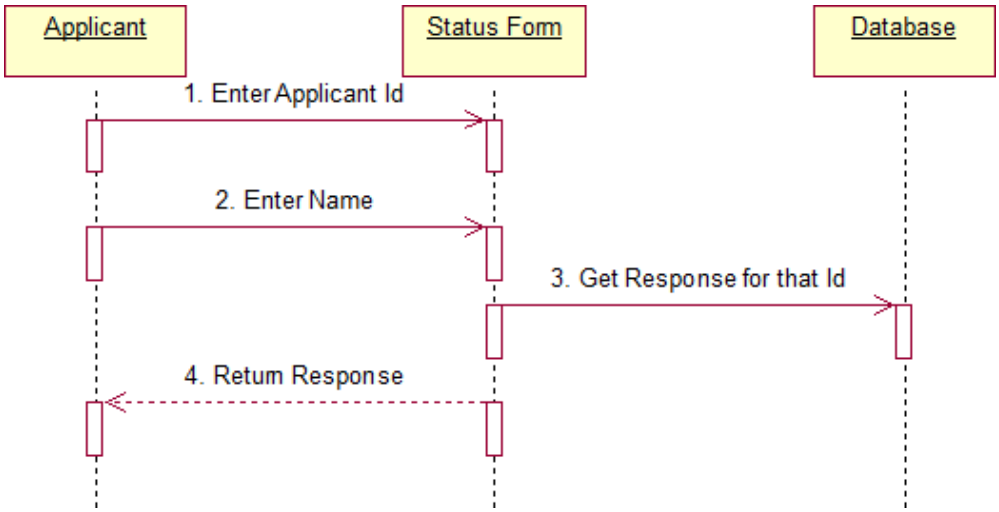


Fig.6.2. SEQUENCE DIAGRAM FOR STATUS

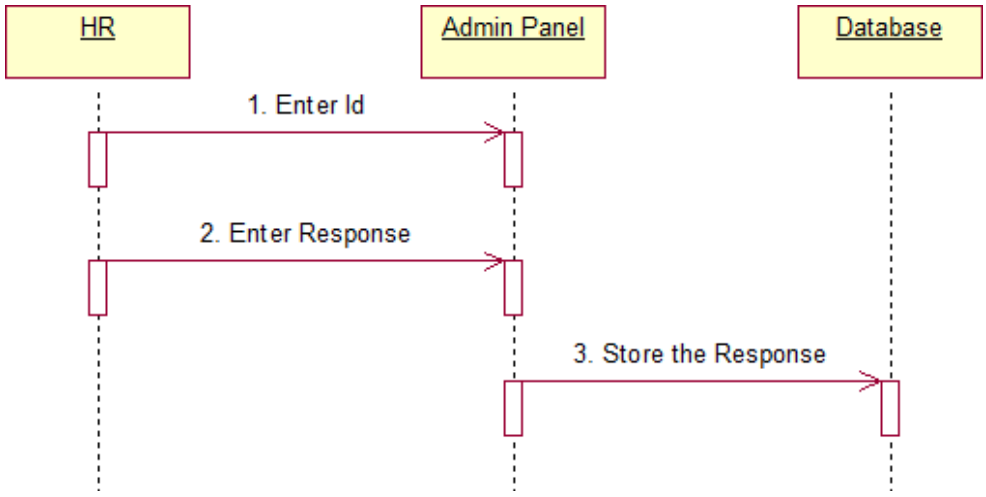
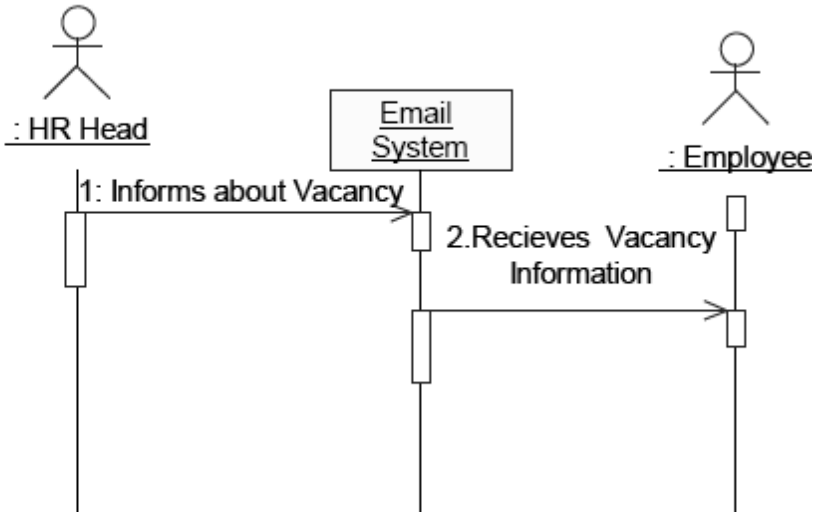


Fig.6.3. SEQUENCE DIAGRAM FOR Admin

Notify Vacancy:



(VII) UML COLLABRATION DIAGRAM:

Communication diagram illustrate that object interact on a graph or network format in which object can be placed where on the diagram. In collaboration diagram the object can be placed in anywhere on the diagram. The collaboration comes from sequence diagram.

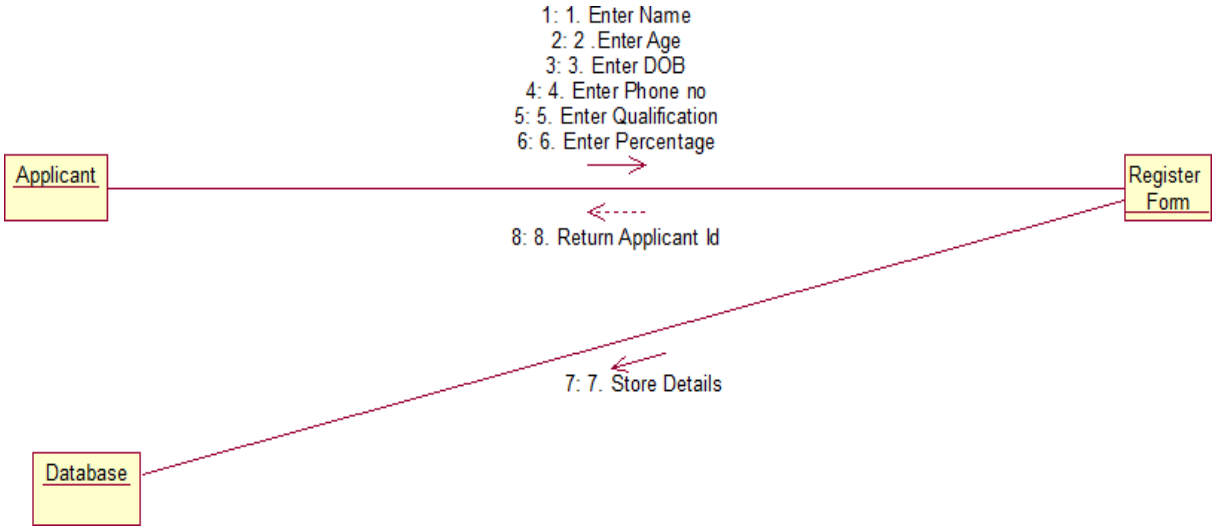


Fig.7.1COLLABRATION DIAGRAM For Register

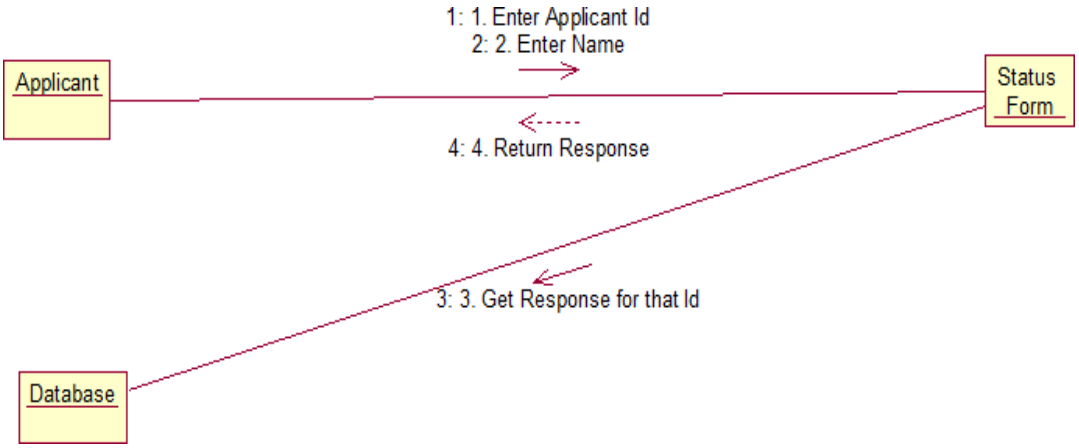


Fig.7.2. COLLABORATION DIAGRAM FOR Status

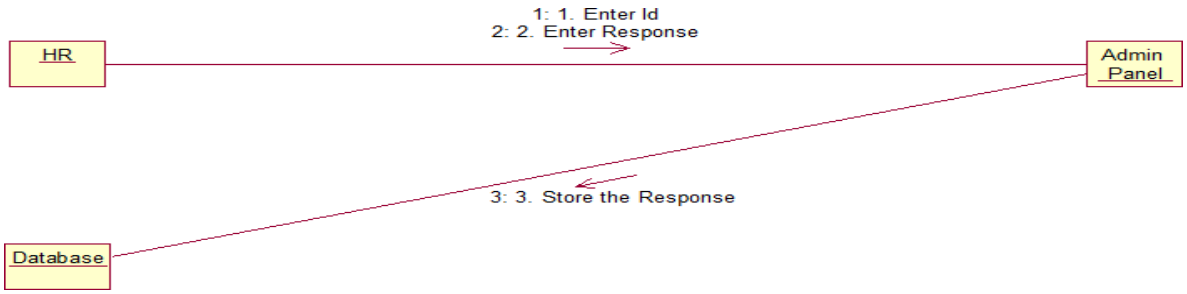
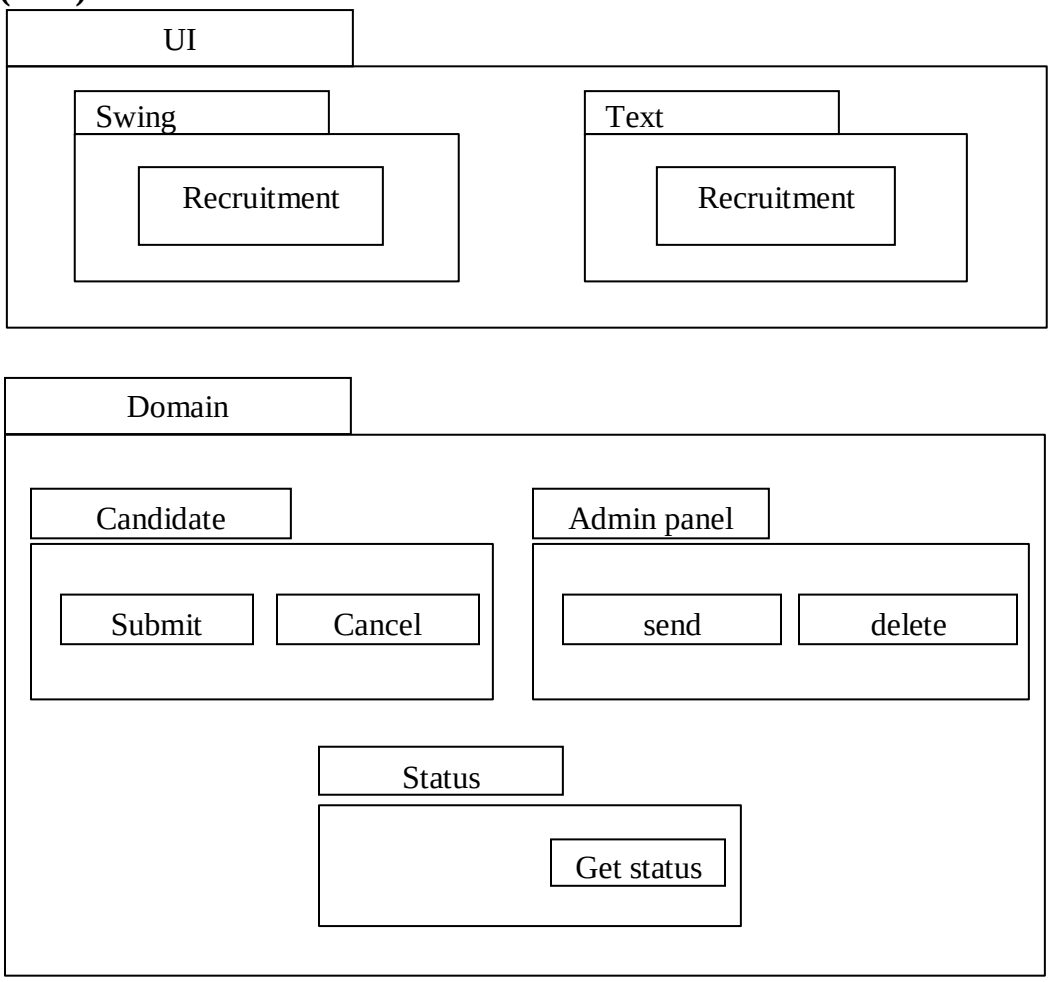
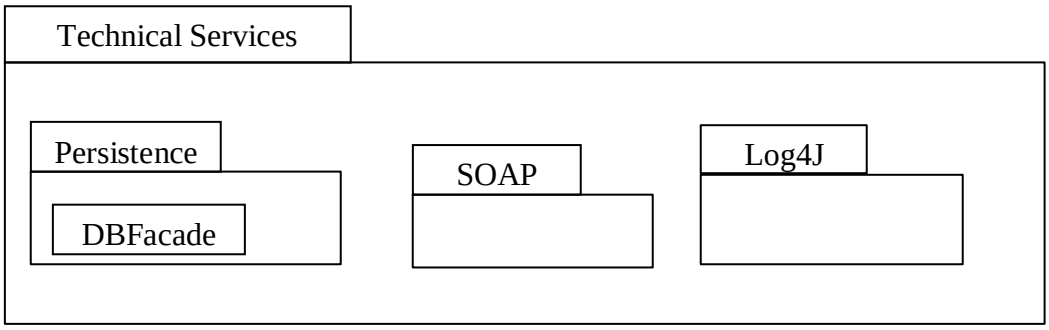


Fig.7.3.COLLABORATION DIAGRAM FOR Admin

(VIII) PARTIAL LAYERD LOGICAL ARCHITECTURE DIAGRAM:





(IX) 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.

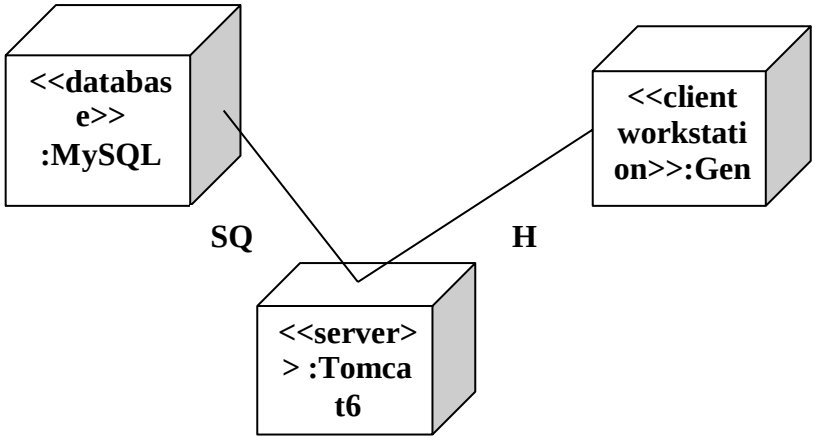


Fig.9.1.DEPLOYMENT DIAGRAM

Component Diagram

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

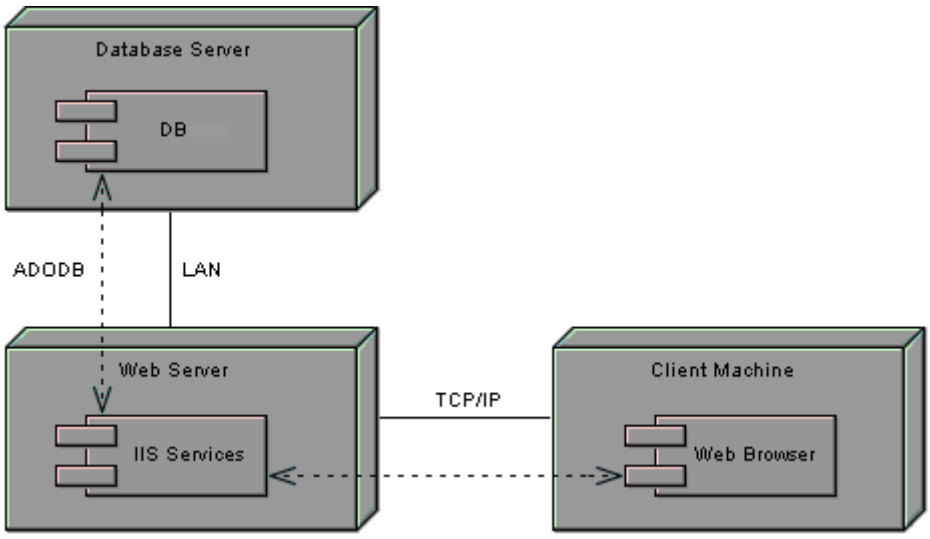


Fig.9.2.COMPONENT DIAGRAM

RESULT:

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