

PONJESLY COLLEGE OF ENGINEERING

NAGERCOIL-629 003

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CCS356- OBJECT ORIENTED SOFTWARE ENGINEERING
LABORATORY RECORD

Name:

Class:

Reg.No:

PONJESLY COLLEGE OF ENGINEERING

NAGERCOIL-629 003

(Affiliated to Anna University Chennai)



CCS356- OBJECT ORIENTED SOFTWARE ENGINEERING

LABORATORY RECORD

Section.....

Regno.....

Certified that this the bonafide record of the work done by
..... in the DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING, PONJESLY COLLEGE OF ENGINEERING,
Nagercoil during the academic year 2024-2025.

HEAD OF THE DEPARTMENT

STAFF INCHARGE

Submitted for the Anna University Chennai Practical Examination, held at PONJESLY
COLLEGE OF ENGINEERING, NAGERCOIL on

Internal Examiner

External Examiner

INDEX

[illegible]

Ex.No :1

Date: PASSPORT AUTOMATION SYSTEM

AIM:

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.

This forms the first and foremost step in the processing of passport application. After the first round of verification done by the system, the information is in turn forwarded to the regional administrator's office. The application is then processed manually based on the report given by the system, and any forfeiting identified can make the applicant liable to penalty as per the law.

The system also provides the applicant the list of available dates for appointment to 'document verification' in the administrator's office, from which they can select one. The system forwards the necessary details to the police for its separate verification whose report is then presented to the administrator. The administrator will be provided with an option to display the current status of application to the applicant, which they can view in their online interface. After all the necessary criteria have been met, the original information is added to the database and the passport is sent to the applicant.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

Use-case Diagrams:

Use-case diagrams graphically depict system behavior. These diagrams provide a high level view of how the system is used as viewed from an outsider's (actor's) perspective. A use-case diagram may depict all or some of the use-cases of a system.

A use-case diagram can contain:

- Actors
- Use-cases
- Interactions or relationships between actors and use-cases in the system including the associations, dependencies and generalizations.

Identify Use-case Diagram

Identifying the use-case diagram involves the following steps.

- i. Choosing the system boundary.
- ii. Identify the primary actor.
- iii. Identify the primary goal for each primary actor.
- iv. Define Use-case.

i. Choosing the System Boundary:

Hardware application : Computers for passport automation system.

Software application : Browsers, Document viewers for receiving the form.

Person : applicant

Organization : regional administrator's (Ministry of External Affairs) office

ii. Identify the Primary Actor:

An Actor represents system users. They help delimit the system and give a clear picture of what the system should do. An actor interacts with, but has no control over the use-case. The primary actor is someone or something that

- Interacts with or uses the system.
- Provides input to and receives information from the system.
- Is external to the system and has no control over the use-cases.

The primary actors involved in “passport automation” are

- Applicant
- regional administrator’s office
- Police station
- system

iii. Identify the goals for each primary actor:

S.No	Actor	Goals
1.	Applicant	Request for passport. Fill form for passport. Submit certificates. Get passport
2.	Regional administrator’s office	Receive details from system. Check the details of applicant. Supply details to Police station. Receives the clearance certificates from Police station. Check the qualifications of candidate. Provide Passport to applicant
3.	Police station	Verifies the details of applicant Provide clearance certificate to regional administrator’s office.
4.	System	Stores the data. Display result by online interface

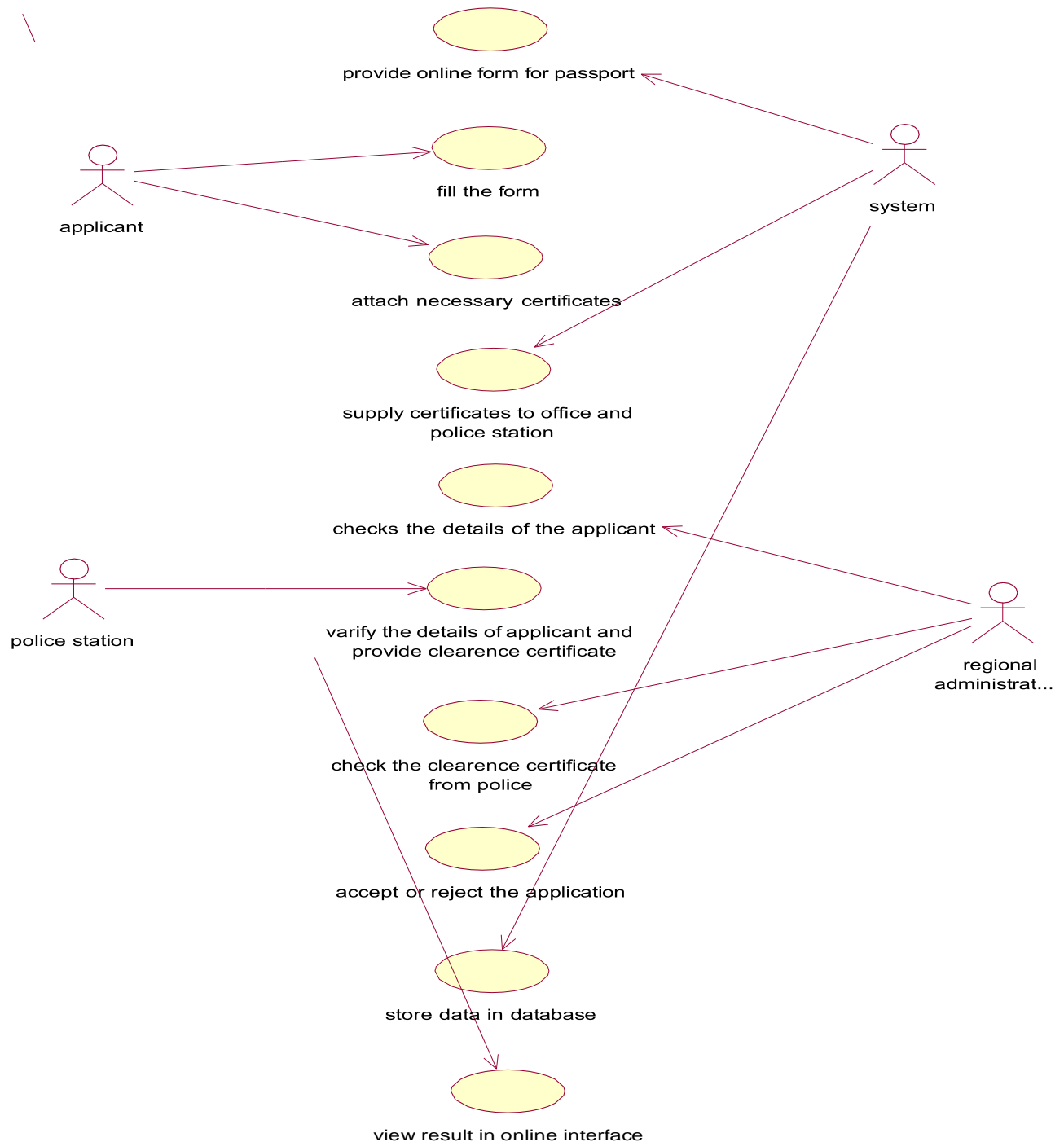
iv. Define Use-Case:

A use-case can be described as a specific way of using the system from a user’s perspective.

A use-case can be characterized as

- A pattern of behavior the system exhibits.
- A sequence of related transactions performed by an actor and the system.
- Delivering something of value to the actor

Use-Case Diagram for passport automation system:



IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

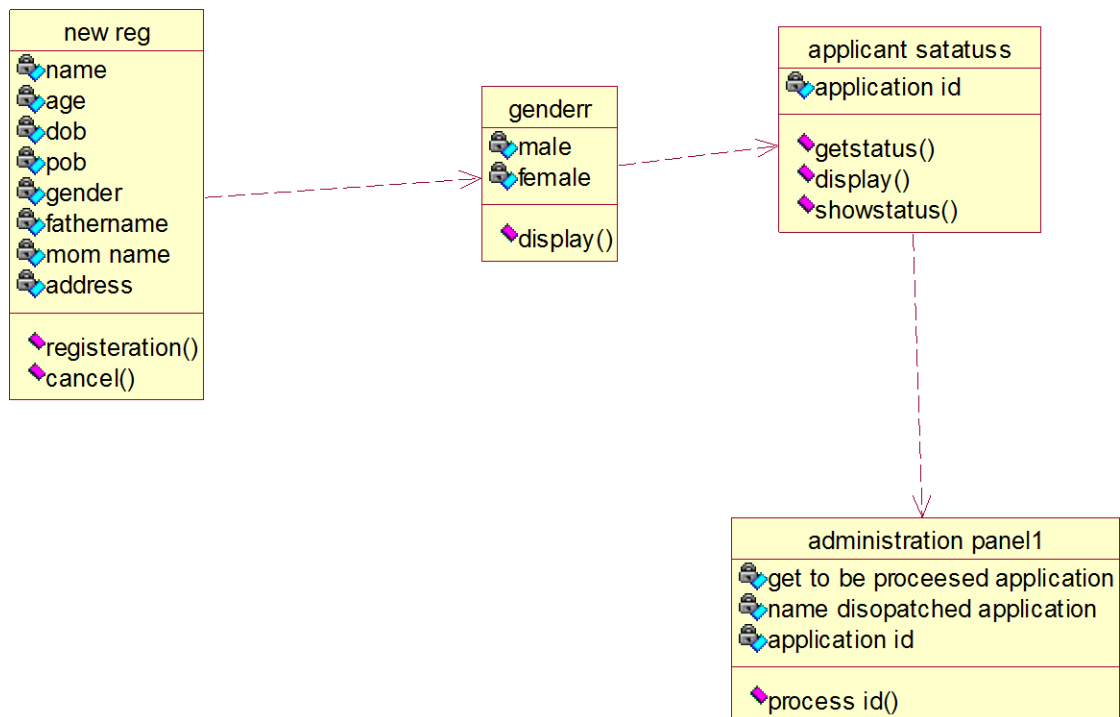
The 3 main strategies to find Conceptual classes are

- Reuse or Modify existing models
- Use a conceptual class category list
- Identify Noun Phases

Identifying the Conceptual Classes using Category List

S.No	Category	Example
1	Business transaction	Registration,payment
2	Transaction line items	Name list
3	Product or service related to a transaction or transaction line item	Passport
4	Where is the transaction recorded?	Database
5	Role of people	Applicant
6	Instruments needed	Cash,certificates,form

UML Class diagram



UML INTERACTION DIAGRAMS

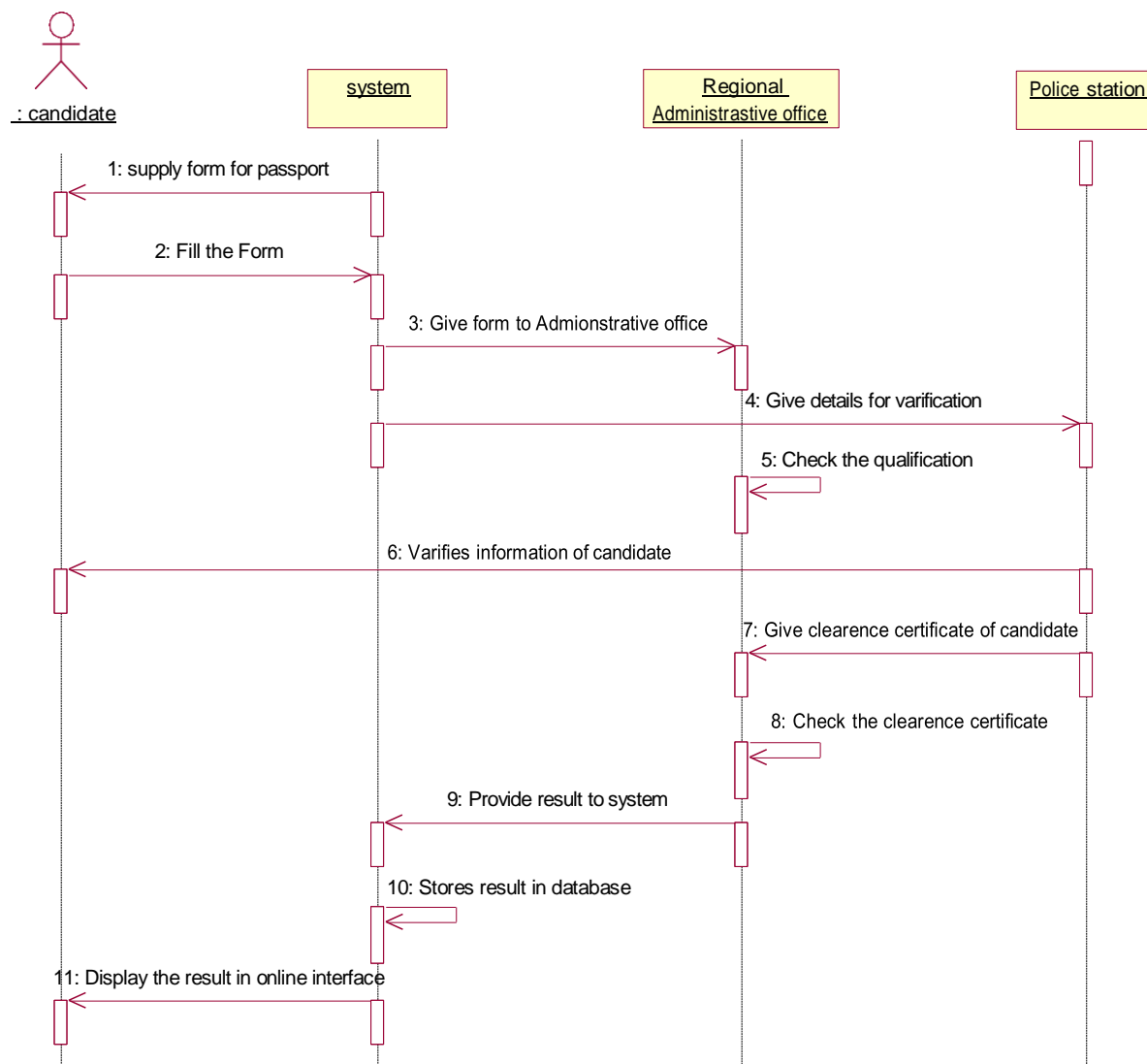
Interaction diagrams are models that describe how a group of objects collaborate in some behavior - typically a single use-case. The diagrams show a number of example objects and the messages that are passed between these objects within the use-case.

I'll illustrate the approach with the following simple use-case. In this behavior the order entry window sends a prepare message to an order. The order then sends prepare to each order line on the order. The order line first checks the stock item, and if the check returns true it removes stock from the stock item. If stock item falls below the reorder level it requests a new delivery.

Interaction diagrams come in two forms, both present in the UML.

- Sequence Diagram
- Collaboration Diagram

UML Sequence Diagram



STATE CHART DIAGRAM

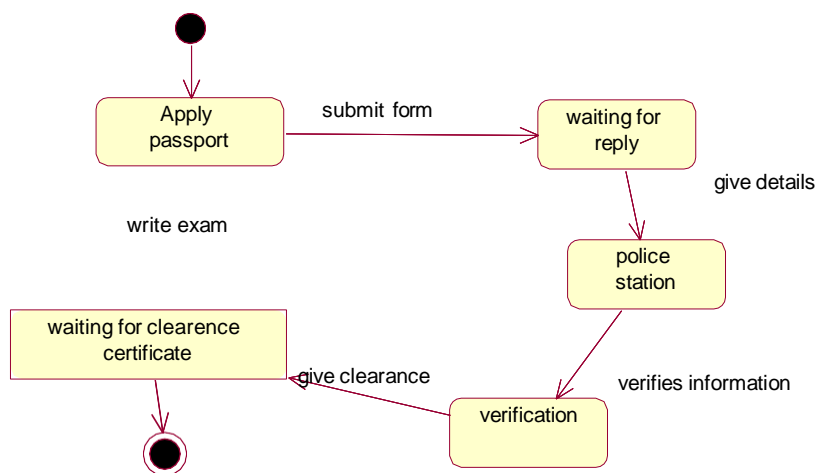
State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. So the most important purpose of State chart diagram is to model life time of an object from creation to termination.

State chart diagram is used to describe the states of different objects in its life cycle. So the emphasis is given on the state changes upon some internal or external events. These states of objects are important to analyze and implement them accurately.

State chart diagrams are very important for describing the states. States can be identified as the condition of objects when a particular event occurs.

Before drawing a State chart diagram we must have clarified the following points:

- Identify important objects to be analyzed.
- Identify the states.
- Identify the events.



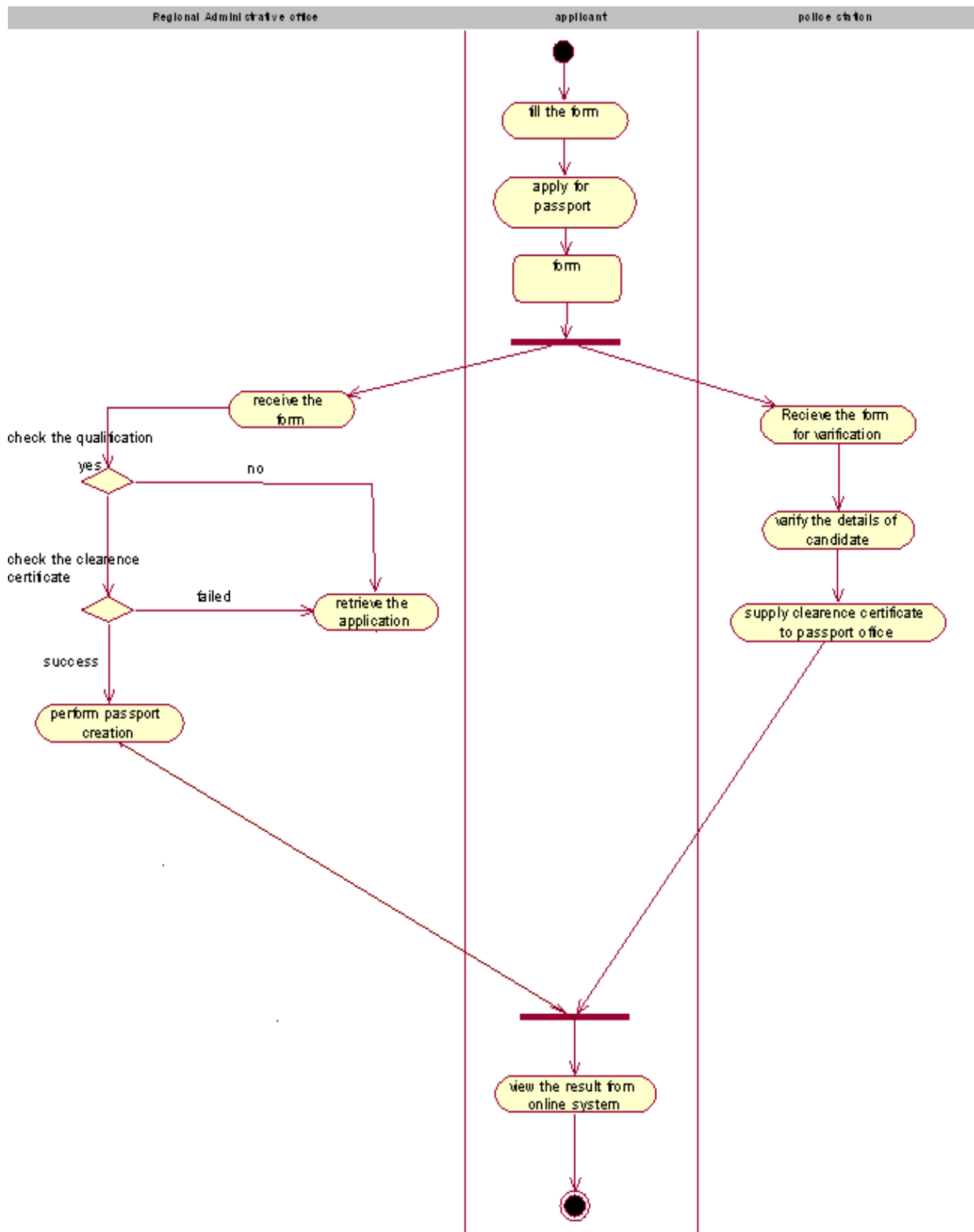
ACTIVITY DIAGRAM

Activity diagrams provide a way to model the workflow of a business process. Activity diagrams are very similar to a flowchart because you can model a workflow from activity to activity. An activity diagram is basically a special case of a state machine in which most of the states are activities and most of the transitions are implicitly triggered by completion of the actions in the source activities. The main difference between activity diagrams and state charts is activity diagrams are activity centric, while state charts are state centric. An activity diagram is typically used for modeling the sequence of activities in a process, whereas a state chart is better suited to model the discrete stages of an object's lifetime.

Understanding Workflows

Each activity represents the performance of a group of actions in a workflow. Once the activity is complete, the flow of control moves to the next activity or state through a transition. If an outgoing transition is not clearly triggered by an event, then it is triggered by the completion of the contained actions inside the activity. A unique activity diagram feature is a swim lane that defines who or what is responsible for carrying out the activity or state. It is also possible to place objects on activity diagrams. The workflow stops when a transition reaches an end state.

Activity Diagram for Passport Automation system



IMPLEMENTATION OF THE CODE

```
/**
 * Class Administration_panel
 */
public class Administration_panel {
// Fields
public void Applicationid;
private void namedispatchedapplication;
// Constructors
//
public Administration_panel () { };
// Methods
// Accessor methods
//
/**
 * Set the value of Applicationid
 * @param newVar the new value of Applicationid
 */
public void setApplicationid ( void newVar ) {
Applicationid = newVar;
}
/**
 * Get the value of Applicationid
 * @return the value of Applicationid
 */
public void getApplicationid () {
return Applicationid;
}
/**
 * Set the value of namedispatchedapplication
 * @param newVar the new value of namedispatchedapplication
 */
private void setNamedispatchedapplication ( void newVar ) {
namedispatchedapplication = newVar;
}
/**
 * Get the value of namedispatchedapplication
 * @return the value of namedispatchedapplication
 */
private void getNamedispatchedapplication () {
return namedispatchedapplication;
}
// Other methods
//
/**
 */
public void processid( )
{
}
```

RESULT:

Ex.No :2

Date: **BOOK BANK**

AIM:

PROBLEM STATEMENT:

Book Bank is a place where we can buy books and magazines where it can be also referred as a book shop where we can find a plenty number of books. Since because of its huge complexity to find a particular book from bank (we refer as bank) we use a computer system embedded with software to find book position in bank. Also reservation for a particular book may be done if one is not found. In a book bank there are cashiers, a clerk (which help to take book) and a system.

- A book bank sells books and magazines.
- A member can reserve book if source is unavailable.
- Reading area must be provided.
- The software embedded on system should provide all the relevant information such as desk number, position, etc.
- From the information collected from the system, clerk will be providing the particular book to customer.
- Clerk should remove or update if any book is being damaged.
- Customer after taking book/magazine should pay the amount to cashier.
- Gift and C.D's must be delivered by cashier if a book belongs.
- The database should track the details about the sale/line item.
- Payment may be taken as either by cheque/cash/ demand draft.
- Barcode reader is embedded with the system for easiness.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

Use-case Diagrams:

Use-case diagrams graphically depict system behavior. These diagrams provide a high level view of how the system is used as viewed from an outsider's (actor's) perspective. A use-case diagram may depict all or some of the use-cases of a system.

A use-case diagram can contain:

- Actors
- Use-cases
- Interactions or relationships between actors and use-cases in the system including the associations, dependencies and generalizations.

Use-case diagrams can be used during analysis to capture the system requirements and to understand how the system should work. During the design phase, you can use use-case diagrams to specify the behavior of the system as implemented.

Identify Use-case Diagram

Identifying the use-case diagram involves the following steps.

i. Choosing the System Boundary:

Hardware application: Barcode reader, printer, scanner, telephone, system.

Software application: Internet, Software for searching books, Database.

Person : Customer, Cashier, Clerk.

Organization : Shops.

ii. Identify the Primary Actor:

An Actor represents system users. They help delimit the system and give a clear picture of what the system should do. An actor interacts with, but has no control over the use-case. The primary actor is someone or something that

- Interacts with or uses the system.
- Provides input to and receives information from the system.
- Is external to the system and has no control over the use-cases.

Actors Involved:

Cashier:

Collects Cash/ Cheque/ Demand Draft and in return gives book.

Clerk:

Updates books and magazines also searches books for customers.

Customer:

Will search books if found means purchases books/ magazines

iii. Identify the goals for each primary actor:

S.No	Actor	Goals
1.	Customer	<ul style="list-style-type: none">• Search book• Buy book• Reading magazines
2.	Cashier	<ul style="list-style-type: none">• Accepting payments• Entering bill details• Printing the bill• Entering in linesaleitem
3.	Clerk	<ul style="list-style-type: none">• Asking feedback• Updating books• Taking book from shelf

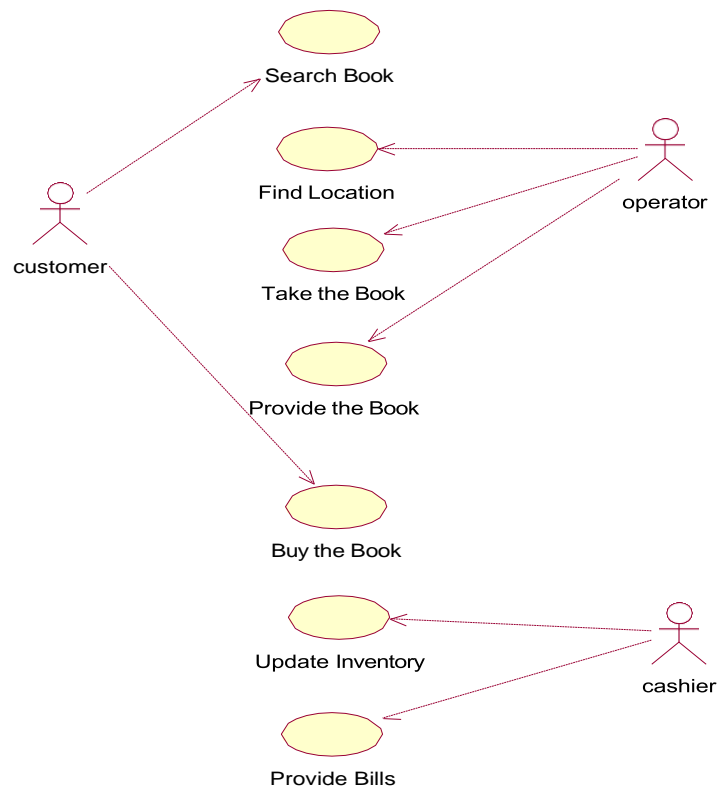
iv. Define Use-Case:

A use-case can be described as a specific way of using the system from a user's perspective.

A use-case can be characterized as

- A pattern of behavior the system exhibits.
- A sequence of related transactions performed by an actor and the system.
- Delivering something of value to the actor

Use-Case Diagram for Book Bank:



IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

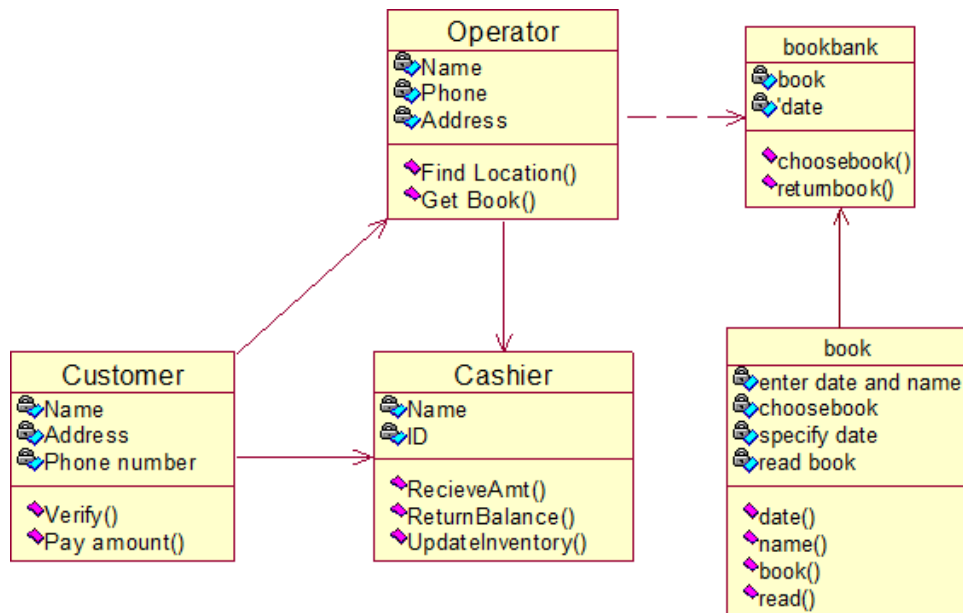
The 3 main strategies to find Conceptual classes are

- Reuse or Modify existing models
- Use a conceptual class category list
- Identify Noun Phases

Identifying the Conceptual Classes using Category List

S.No	Category	Example
1.	Transaction line item	Database of system
2.	Record	System database
3.	Noteworthy events	Suggestions, reservation
4.	Physical objects	Reservation from, registration form, book
5.	Container of things	Book bank
6.	Catalogs	System, menu cards
7.	Schedule manuals	Shop schedule

UML Class diagram



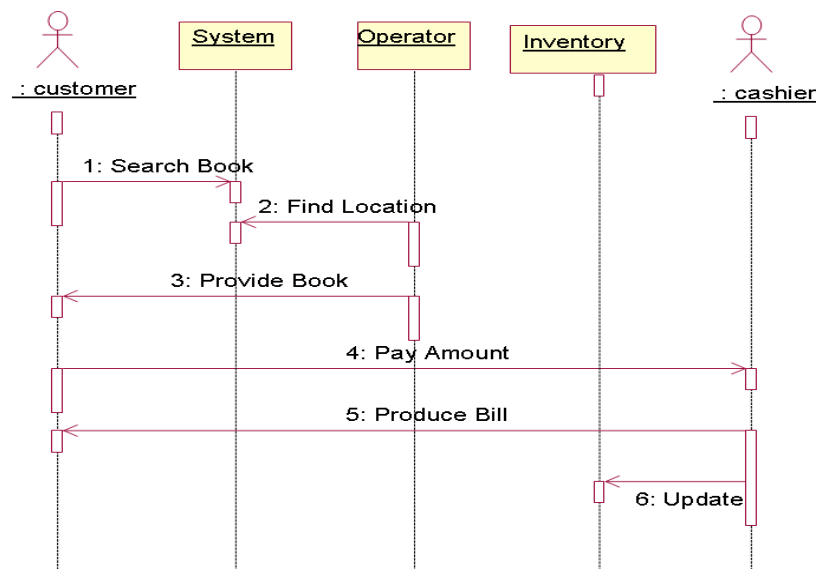
UML INTERACTION DIAGRAMS

Interaction diagrams are models that describe how a group of objects collaborate in some behavior - typically a single use-case. The diagrams show a number of example objects and the messages that are passed between these objects within the use-case.

Interaction diagrams come in two forms, both present in the UML.

- Sequence Diagram
- Collaboration Diagram

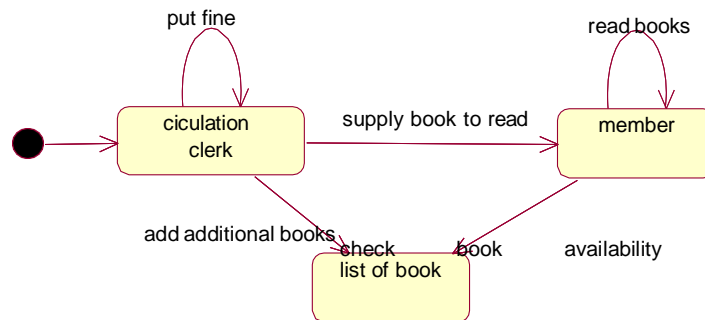
UML Sequence Diagram



STATE CHART DIAGRAM

State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. So the most important purpose of State chart diagram is to model life time of an object from creation to termination.

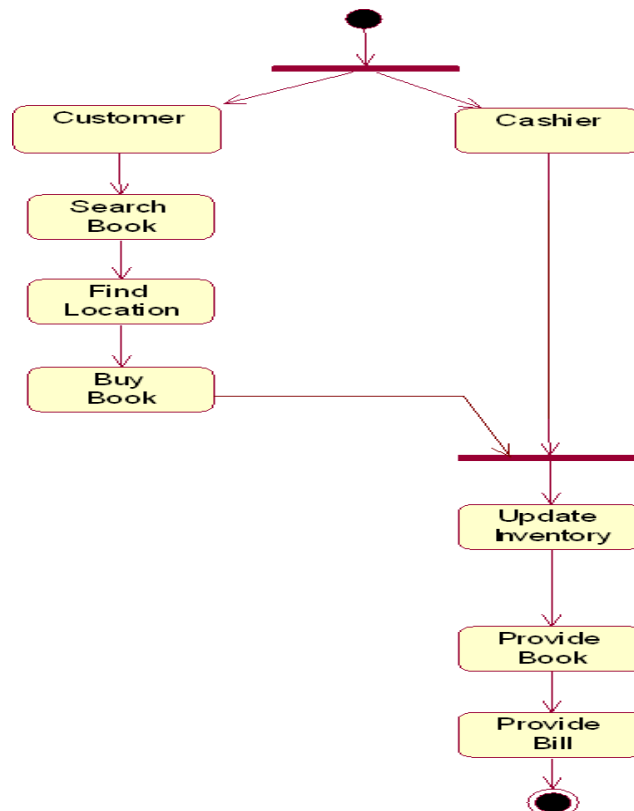
State chart diagram is used to describe the states of different objects in its life cycle. So the emphasis is given on the state changes upon some internal or external events. These states of objects are important to analyze and implement them accurately.



ACTIVITY DIAGRAM

Activity diagrams provide a way to model the workflow of a business process. Activity diagrams are very similar to a flowchart because you can model a workflow from activity to activity. An activity diagram is basically a special case of a state machine in which most of the states are activities and most of the transitions are implicitly triggered by completion of the actions in the source activities.

Activity Diagram for BookBank



IMPLEMENTATION OF THE CODE

```
/**
 * Class operators
 */
public class operators {
    public char name;
    public long phone_no_;
    public char address;
    public operators () { };
    public void setName ( char newVar ) {
        name = newVar;
    }
    public char getName ( ) {
        return name;
    }
}
```

RESULT:

Ex.No :3

Date: EXAM REGISTRATION

AIM:

PROBLEM STATEMENT

The Exam Registration system is one of the common ways of registering participants in an exam in today's world. The Exam Registration system starts with filling up the pre-exam participation form, sending it to the corresponding exam authorities via post or by mail. The authorities involved verify the contents and send the exam id or exam number. The student or the concerned person will now be termed as eligible or liable to write the test. The Exam Registration System has advanced to such an extent that manual work has been greatly reduced.

- The Candidate applies for the Registration form
- Once the form is received, the Candidate can fill up the form and send it either through post or by mail.
- The form is verified by the concerned authorities(EC- COUNCIL)
- Once the verification is done and the candidate is approved as eligible to write the test, he/she will be sent an id containing the exam number , instructions and schedule.
- The Candidate then sits the exam.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

Use-case Diagrams:

Use-case diagrams graphically depict system behavior. These diagrams provide a high level view of how the system is used as viewed from an outsider's (actor's) perspective. A use-case diagram may depict all or some of the use-cases of a system.

A use-case diagram can contain:

- Actors
- Use-cases
- Interactions or relationships between actors and use-cases in the system including the associations, dependencies and generalizations.

Identify Use-case Diagram

Identifying the use-case diagram involves the following steps.

i. Choosing the System Boundary:

Hardware application: Computers for online Registration.

Software application : Browsers, Document viewers for receiving the form.

Person : Candidates

Organization : Companies

ii. **Identify the Primary Actor:**

Actors Involved:

Candidate.

EC- Councilor

Exam Database

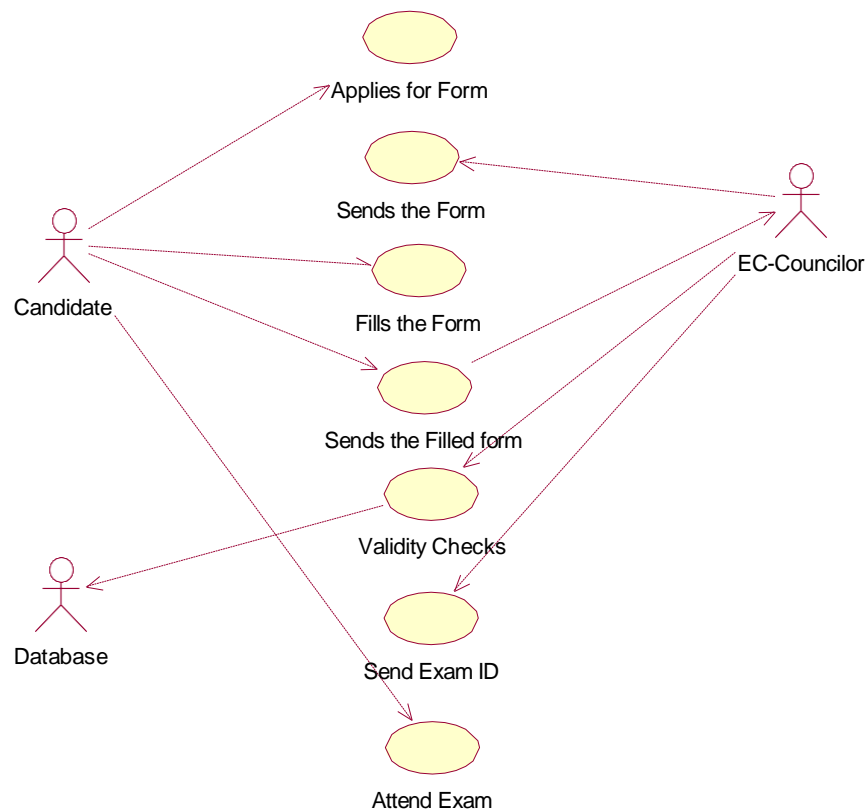
iii. **Identify the goals for each primary actor:**

Sl.No	Actor	Goals
1.	Candidate	Apply for Registration Form, Fill the form, Send it to the concerned Department.
2.	EC-Councilor	Receives the filled forms Verifies the contents, Sends the ID, registration Number& Schedule
3.	System Database	Stores the Registration Form.

iv. **Define Use-Case:**

A use-case can be described as a specific way of using the system from a user's perspective.

Use-Case Diagram for Exam Registration:

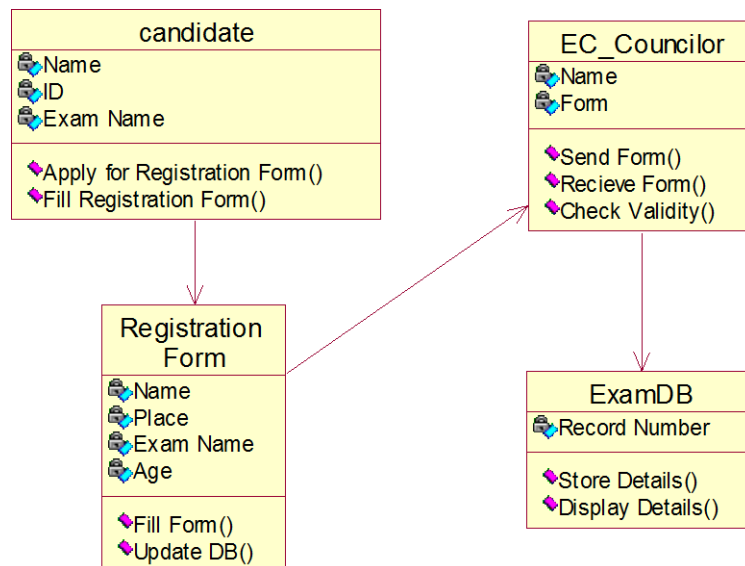


IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

Identifying the Conceptual Classes using Category List

SI.NO	Conceptual Class Category	Example
1.	Record	Exam Database
2.	Roles of People	Candidate,EC-Councilor
3.	Place of Event	Exam Registration Center
4.	Noteworthy Events	Exam Registration
5.	Physical Objects	Registration Form, Registration ID
6.	Description of Things	Candidate Info

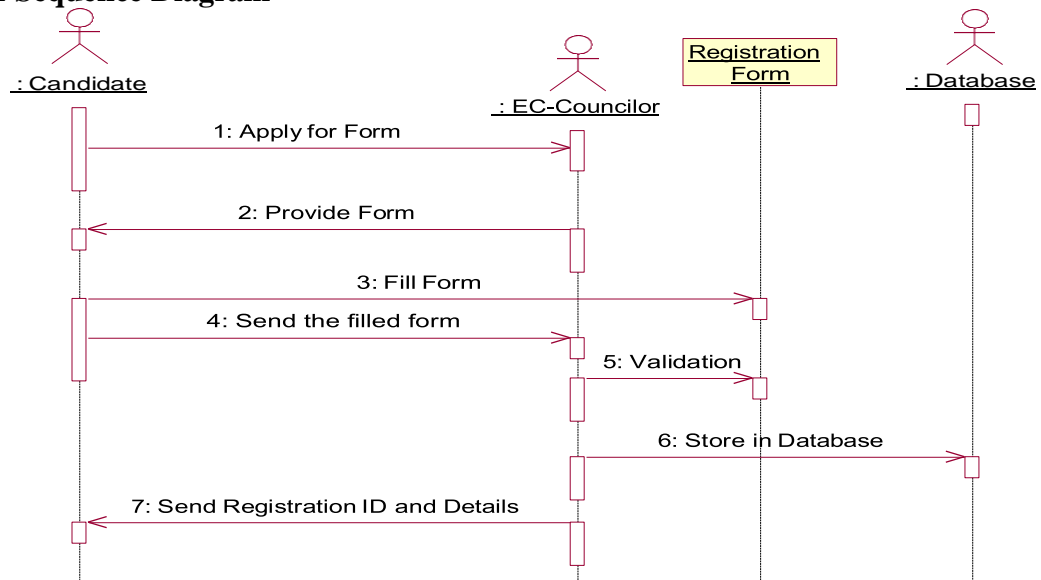
UML Class diagram



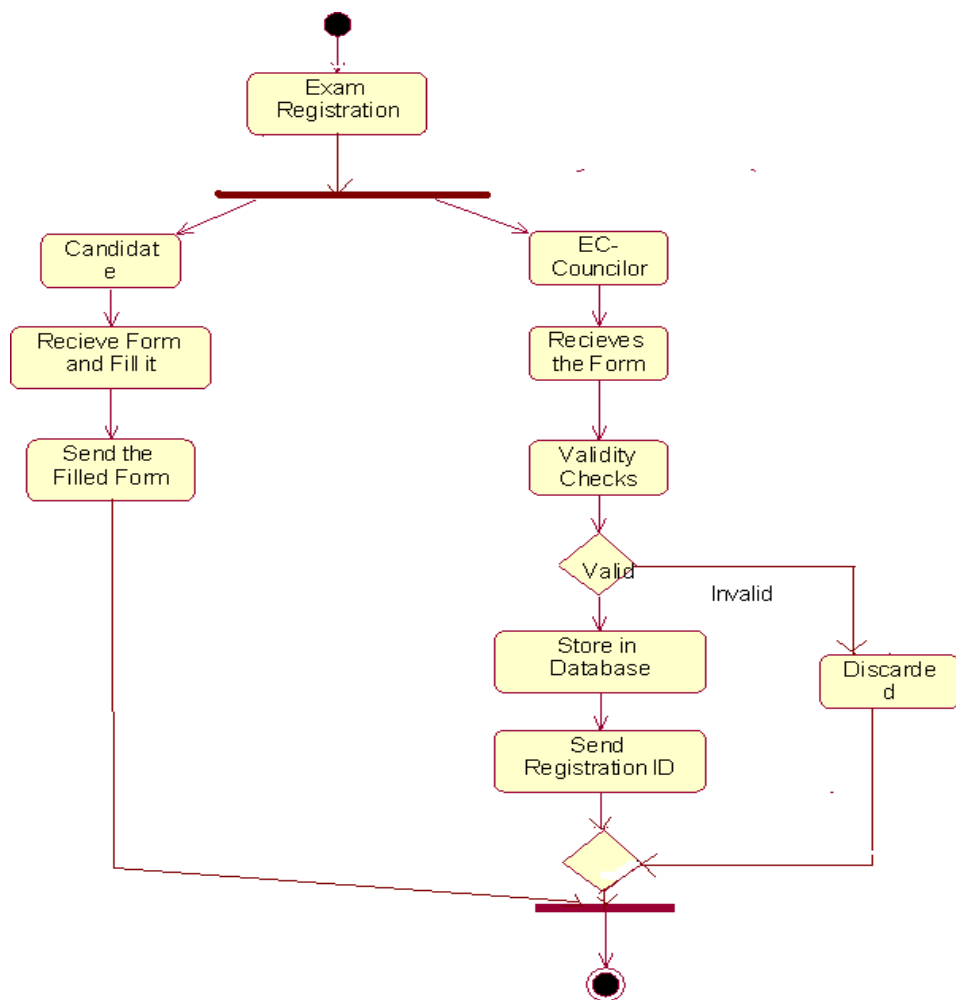
UML INTERACTION DIAGRAMS

Interaction diagrams are models that describe how a group of objects collaborate in some behavior - typically a single use-case. The diagrams show a number of example objects and the messages that are passed between these objects within the use-case.

UML Sequence Diagram



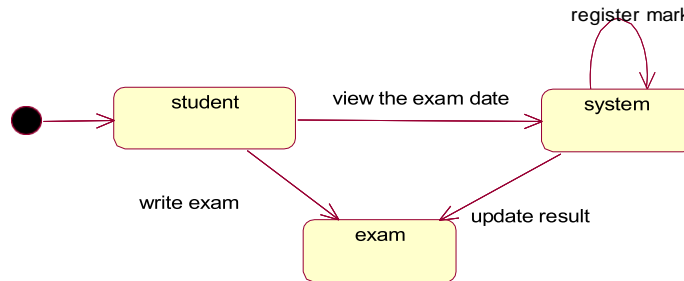
ACTIVITY DIAGRAM



STATE CHART DIAGRAM

State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. So the most important purpose of State chart diagram is to model life time of an object from creation to termination.

States can be identified as the condition of objects when a particular event occurs.



IMPLEMENTATION OF THE CODE

```
#include "students.h"

///ModelId=512746F901E4
students::getmark()
{
}

#ifndef STUDENTS_H_HEADER_INCLUDED_AE96D2A7
#define STUDENTS_H_HEADER_INCLUDED_AE96D2A7
///ModelId=512746E80148
class students
{
public:
    ///ModelId=512746F901E4
    getmark();
private:
    ///ModelId=512746F201C5
    name;
    ///ModelId=512746F40203
    dept;
    ///ModelId=512746F6001F
    age;
};

#endif /* STUDENTS_H_HEADER_INCLUDED_AE96D2A7 */
#include "examm.h"
///ModelId=5127471D00CB
examm::list()
{
}
///ModelId=5127471F029F
examm::show()
{
}
```

RESULT:

Ex.No :4

Date:

STOCK MAINTENANCE SYSTEM

AIM:

PROBLEM STATEMENT

- The stock has to be maintained properly since the whole marketing process can be improved. Stock maintenance in this project gives the idea about how products are maintained in a particular concern.
- The stock details which includes the amount of stock available, the stock is to be purchased, the date or the time it is being bought or delivered, the amount that is already available are maintained in this project.
- The stock maintenance in this project is understood by going through the modules that is being involved. The whole economic status is being improved properly if stock is maintained.
- Storing of information about the stock values and updating the stock values for each organization which is using this system, keeps track of all the information about the stock exchange that are made by the customers, having registration feature of adding up new customers to the organization are provided in this system.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

A use-case diagram can contain:

- Actors
- Use-cases
- Interactions or relationships between actors and use-cases in the system including the associations, dependencies and generalizations.

Identify Use-case Diagram

Identifying the use-case diagram involves the following steps.

i. **Choosing the System Boundary:**

Hardware application: Computers for inventory control.

Software application : object oriented language for run the system.

Person : Administrator, Customer,Supplier.

Organization : shop.

ii. **Identify the Primary Actor:**

An Actor represents system users. They help delimit the system and give a clear picture of what the system should do. An actor interacts with, but has no control over the use-case. The primary actor is someone or something that

- Interacts with or uses the system.
- Provides input to and receives information from the system.

- Is external to the system and has no control over the use-cases.

Actors Involved:

Administrator

Customer

Supplier

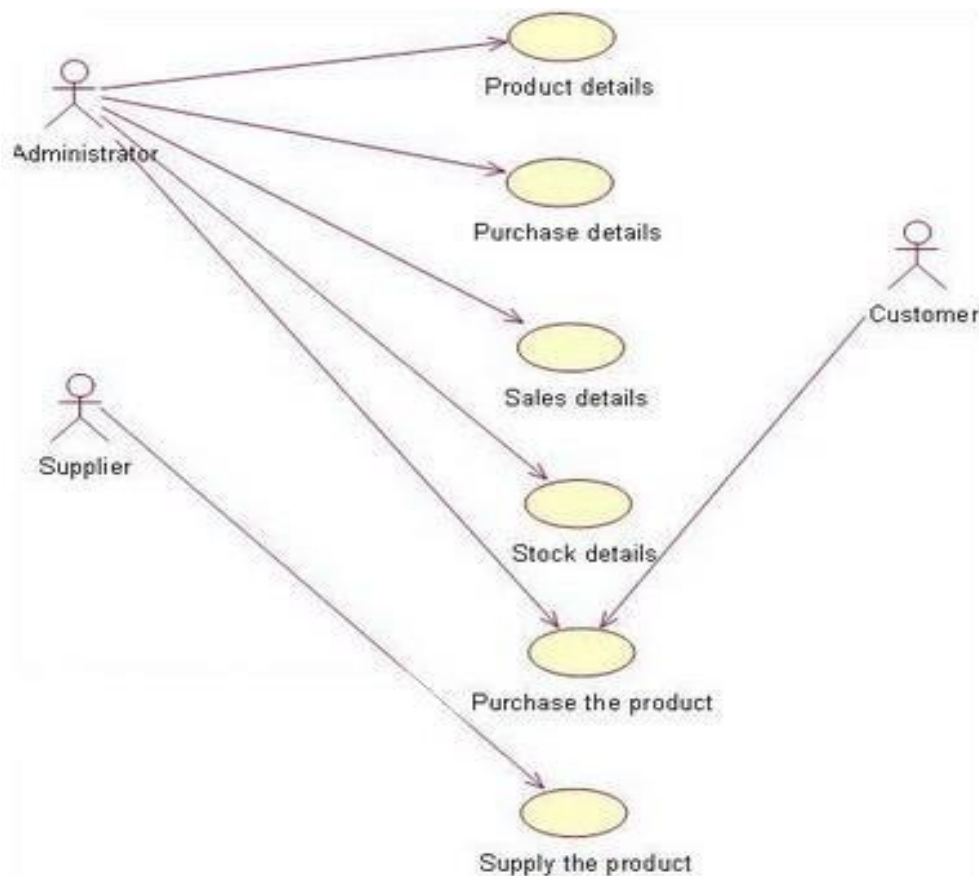
iii. Identify the goals for each primary actor:

Sl.No	Actor	Goals
1.	Administrator	Enter product details, Enter purchase details, Enter sale details, Enter stock details
2.	Customer	Purchase the product
3.	Supplier	Supply the product

iv. Define Use-Case:

A use-case can be described as a specific way of using the system from a user's perspective.

Use-Case Diagram for Stock maintenance system:

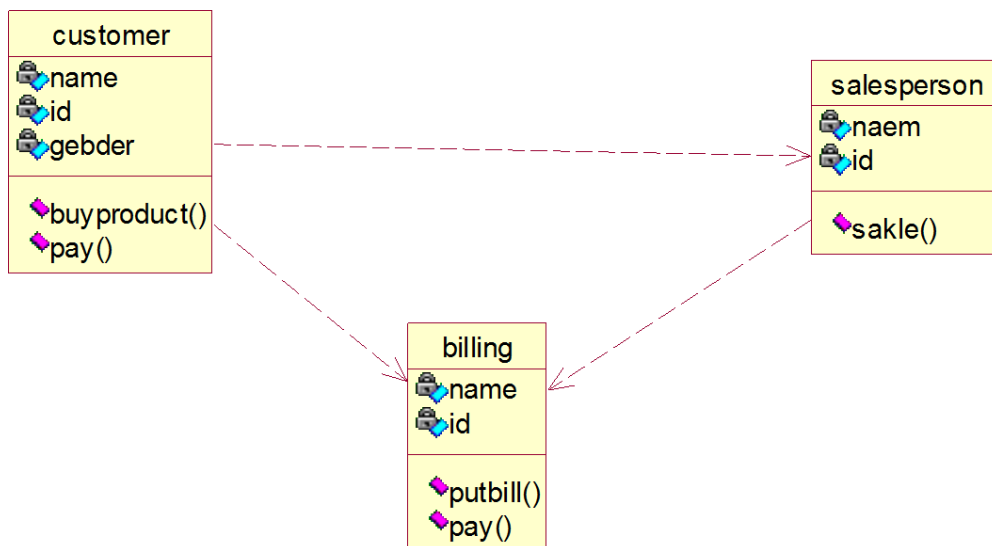


IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

Identifying the Conceptual Classes using Category List

SL.NO	Conceptual Class Category	Example
1.	Record	Register
2.	Roles of People	Customer, salesperson
3.	Place of Event	Store
4.	Noteworthy Events	Stock maintenance
5.	Physical Objects	Register
6.	Description of Things	List of item

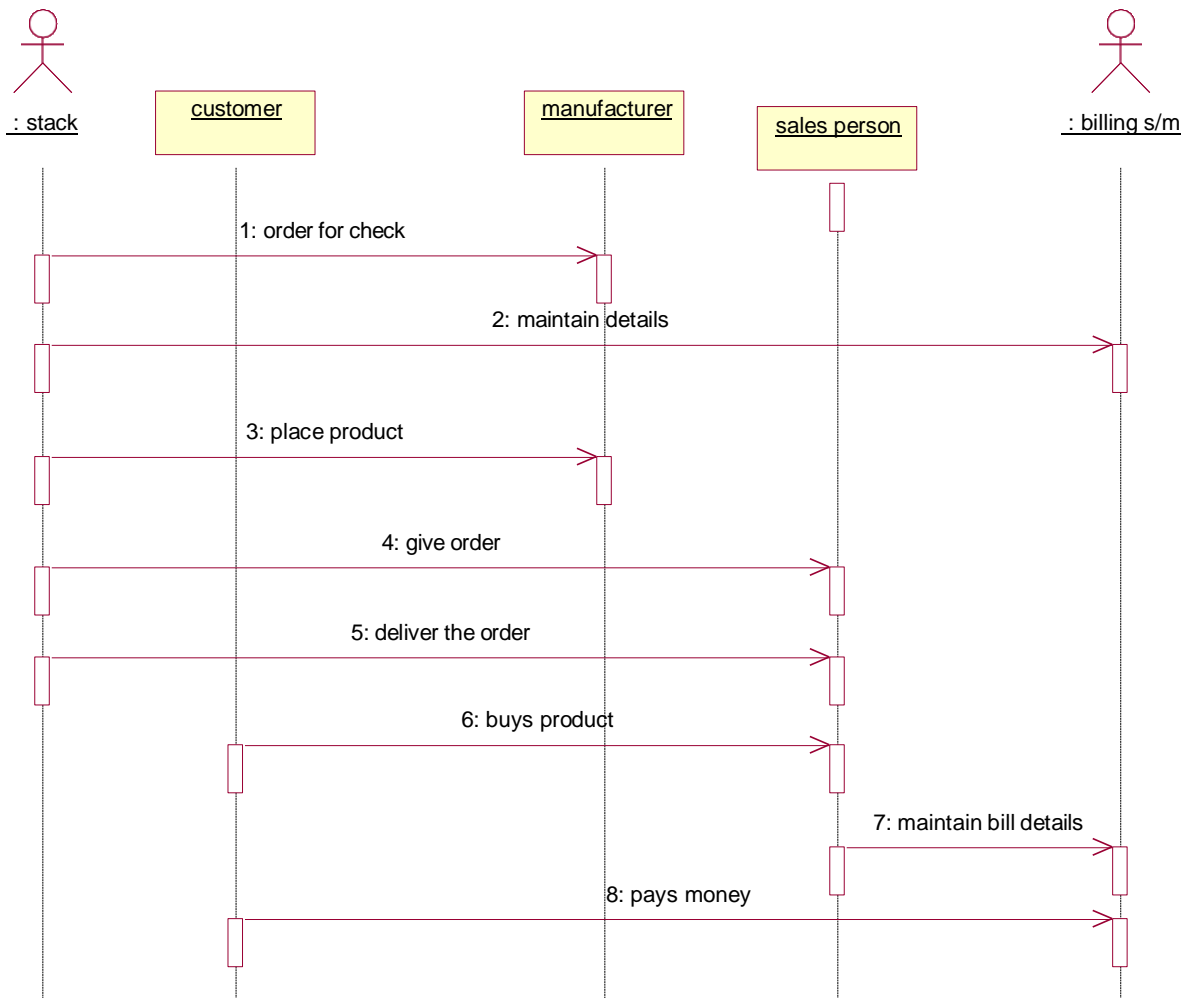
UML Class diagram



UML INTERACTION DIAGRAMS

Interaction diagrams are models that describe how a group of objects collaborate in some behavior - typically a single use-case.

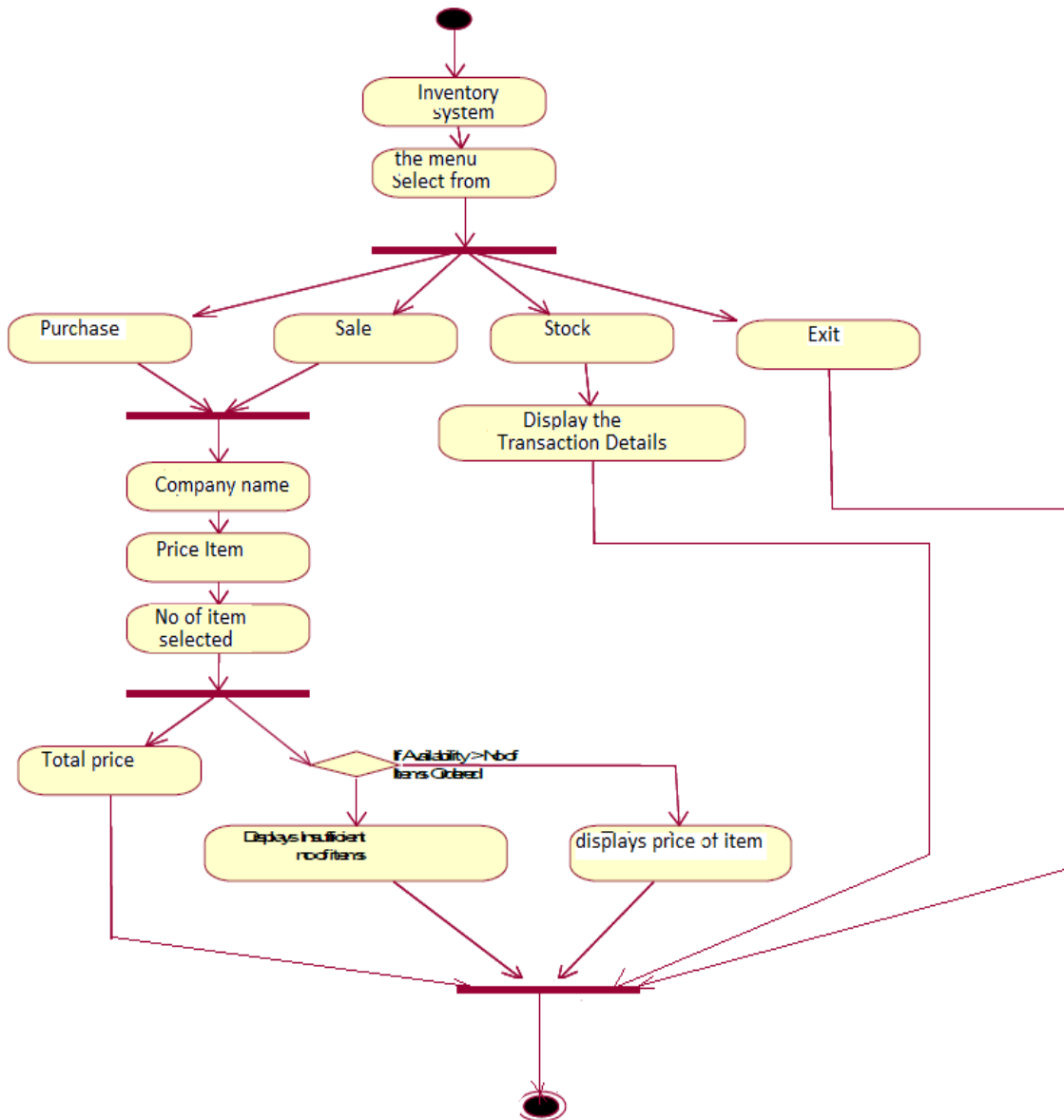
UML Sequence Diagram



ACTIVITY DIAGRAM

Activity diagrams provide a way to model the workflow of a business process. Activity diagrams are very similar to a flowchart because you can model a workflow from activity to activity. An activity diagram is basically a special case of a state machine in which most of the states are activities and most of the transitions are implicitly triggered by completion of the actions in the source activities.

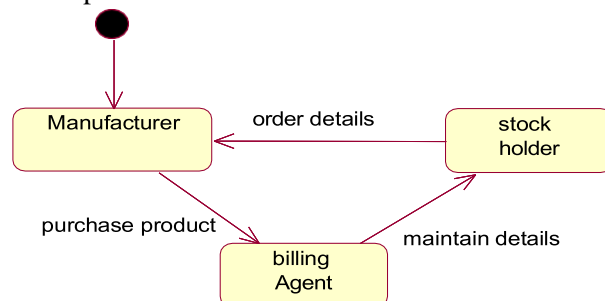
Activity Diagram for Stock maintenance system



STATE CHART DIAGRAM

State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. So the most important purpose of State chart diagram is to model life time of an object from creation to termination.

State chart diagrams are very important for describing the states. States can be identified as the condition of objects when a particular event occurs.



IMPLEMENTATION OF THE CODE

```
#include "billing.h"

///
```

```
{  
}
```

```
#ifndef CUSTOMER1_H_HEADER_INCLUDED_AE96BAC3  
#define CUSTOMER1_H_HEADER_INCLUDED_AE96BAC3
```

```
///  
//##ModelId=5140432202BF  
class customer  
{  
public:  
    ///  
    //##ModelId=5140434502CE  
    buyproduct();  
  
    ///  
    //##ModelId=5140434D00DA  
    pay();  
  
private:  
    ///  
    //##ModelId=5140433E0261  
    name;  
    ///  
    //##ModelId=514043400148  
    id;  
    ///  
    //##ModelId=5140434100FA  
    gebder;  
};
```

RESULT:

Ex.No :5

Date: ONLINE COURSE RESERVATION SYSTEM

AIM:

PROBLEM STATEMENT

- Search the online course and select any one of the course and download it
- Download the form fill the form and send to the correct destination
- The administrator receive the form ,and send acknowledgement if the course is available
- The student attend the interview and get the application slip
- If the student pay the fees and receipt and then join the course

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

Identify Use-case Diagram

Identifying the use-case diagram involves the following steps.

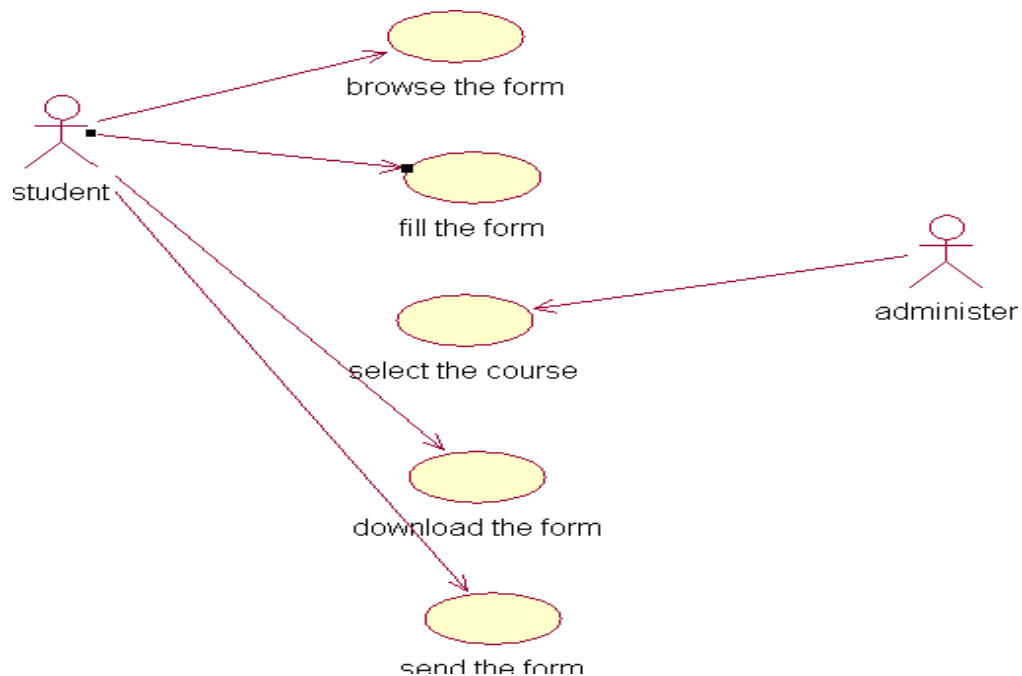
- i. Choosing the system boundary.
 - ii. Identify the primary actor.
 - iii. Identify the primary goal for each primary actor.
 - iv. Define Use-case.
- i. **Choosing the System Boundary:**
- Hardware application : Computers for online course reservation system.
- Software application : Browsers, Document viewers for receiving the form.
- Person : applicant
- Organization : College.
- ii. **Identify the Primary Actor:**
- An Actor represents system users. They help delimit the system and give a clear picture of what the system should do. An actor interacts with, but has no control over the use-case. The primary actor is someone or something that
- Interacts with or uses the system.
 - Provides input to and receives information from the system.
 - Is external to the system and has no control over the use-cases.
- Actors Involved:**
- Student

- Administrator
- system

iii. Identify the goals for each primary actor:

ACTOR	GOALS
Student	1 Select the course 2 Download the form 3 Fill the form 4 Send the form
system	1. Searching the online 2. Send the form 3 Enter the details
administer	1 Receiving the form 2 Sending the Acknowledgement 3 Conducting the interview 4 Select the student

Use-Case Diagram for Online course reservation system:

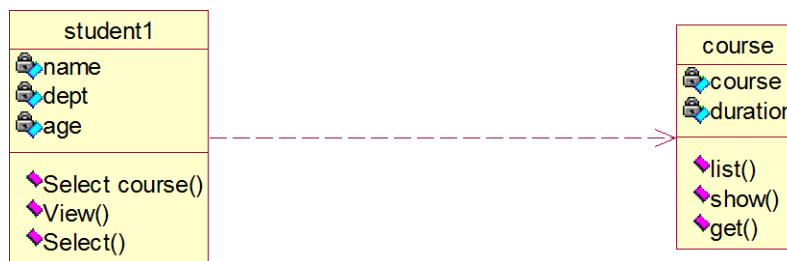


IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

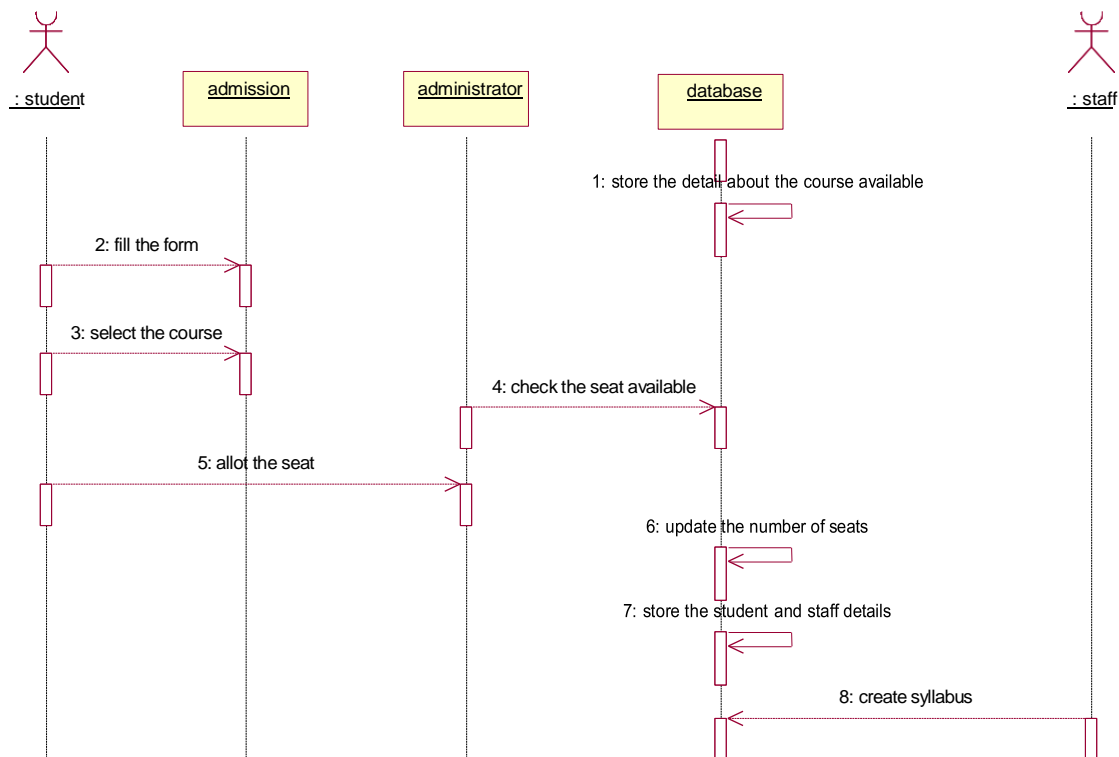
Identifying the Conceptual Classes using Category List

Conceptual class category List	object
Role of people	Student,administer
Place of transation	College
Container of things	College
Things in a container	Course

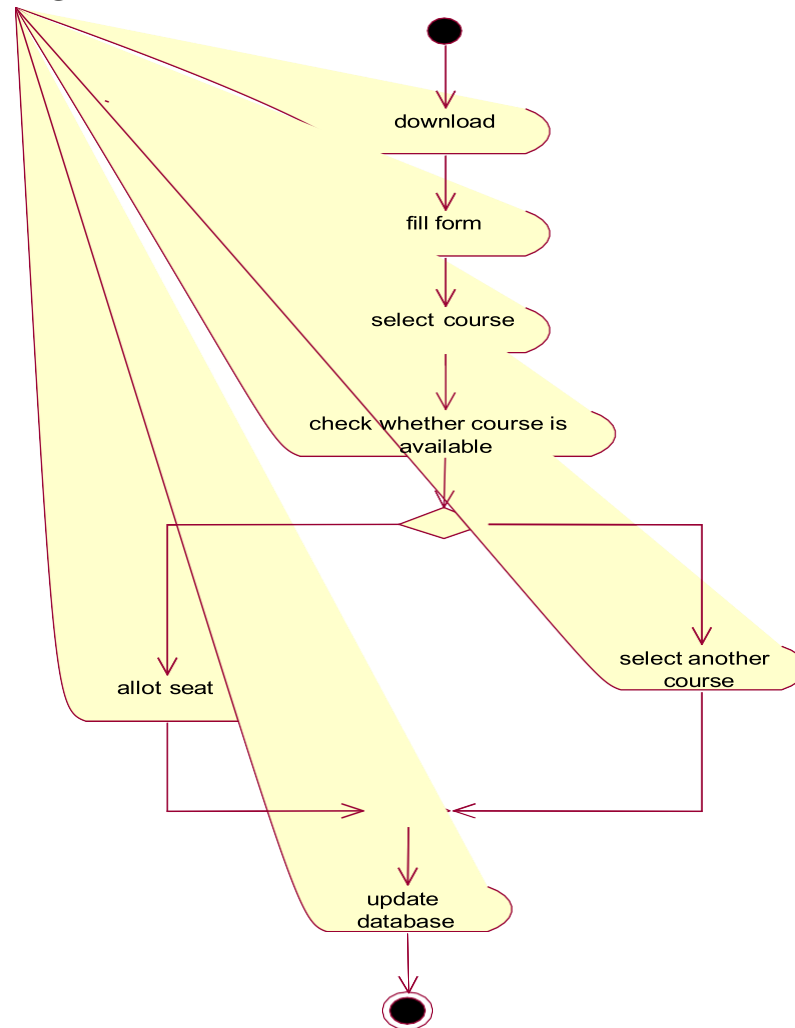
UML Class diagram



UML INTERACTION DIAGRAMS

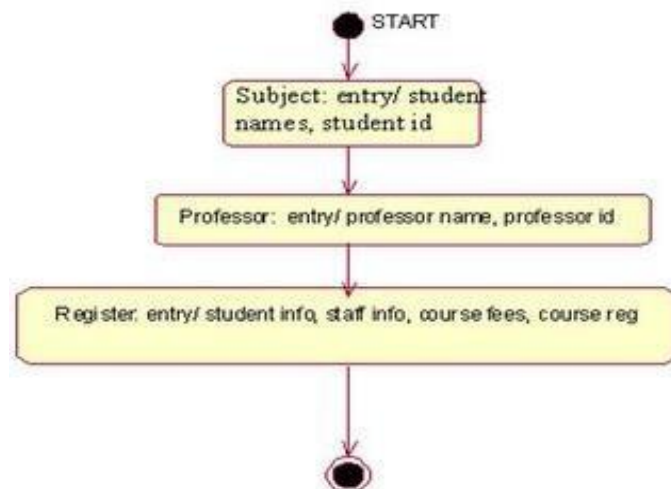


ACTIVITY DIAGRAM



Activity Diagram for Online course reservation system

STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

Course.cpp

```
#include "course1.h"

///
```

RESULT:

Ex.No :6

Date: **AIRLINE/RAILWAY RESERVATION SYSTEM**

AIM:

PROBLEM STATEMENT

- First passenger's browse the website
- Passengers check whether seats are available
- Then passengers send the credit card numbers
- Travel agent checks whether the card is valid
- Then the passengers accept the credit card
- Finally receive the ticket.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

I. Identify Use-case Diagram

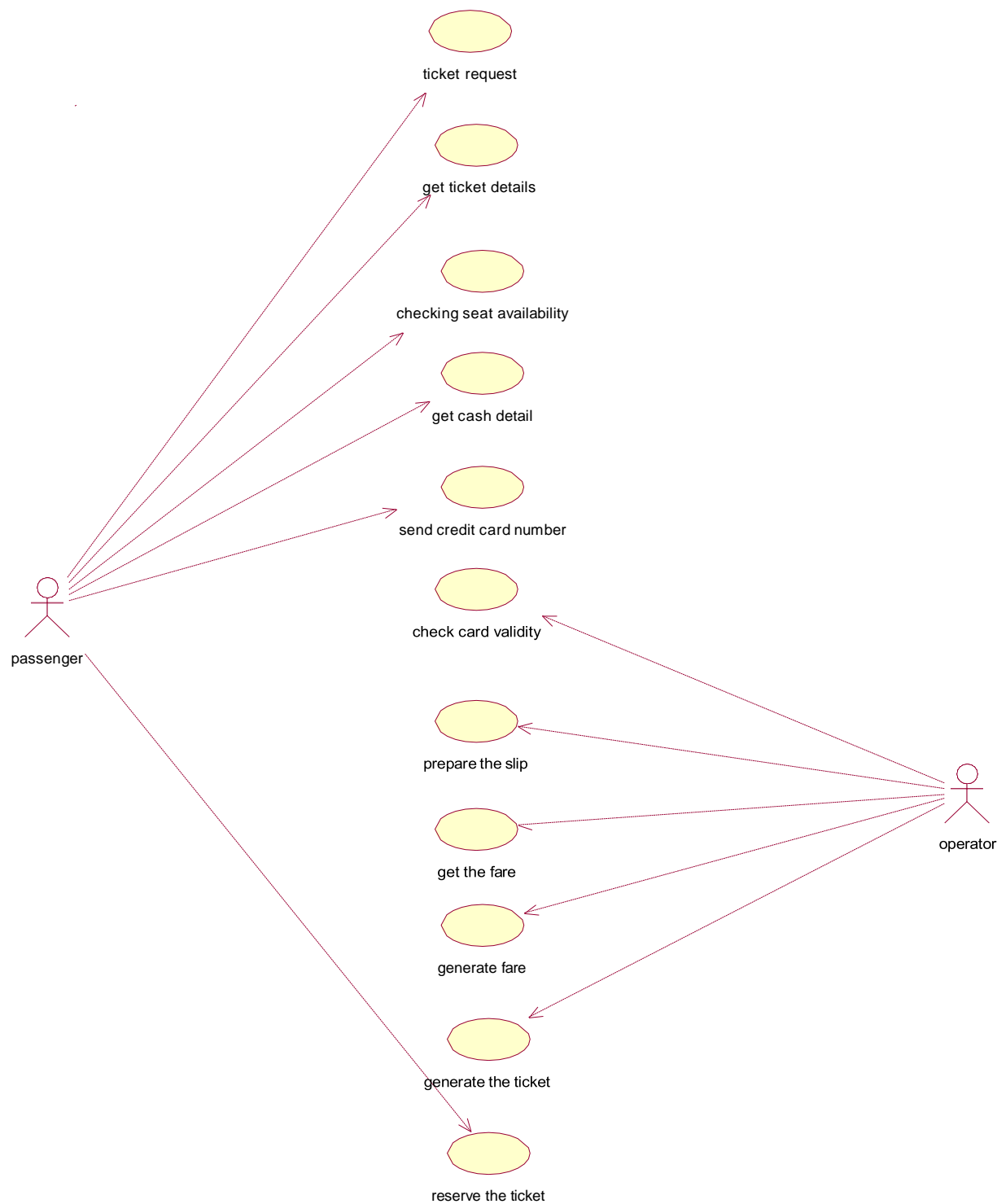
Identifying the use-case diagram involves the following steps.

- i. Choosing the system boundary.
 - ii. Identify the primary actor.
 - iii. Identify the primary goal for each primary actor.
 - iv. Define Use-case.
- i. **Choosing the System Boundary:**
- Hardware application : Computers for E-ticketing.
- Software application : Browsers, Document viewers for receiving the form.
- Person : passenger
- Organization : Ticketing reservation system.
- ii. **Identify the Primary Actor:**
- passenger
 - operator

iii. Identify the goals for each primary actor:

Actor	Goal
passenger	1.ticket request 2.reserve the ticket
operator	1.check card validity 2.prepare the slip 3.Generate the fare

Use-Case Diagram for E-ticketing:

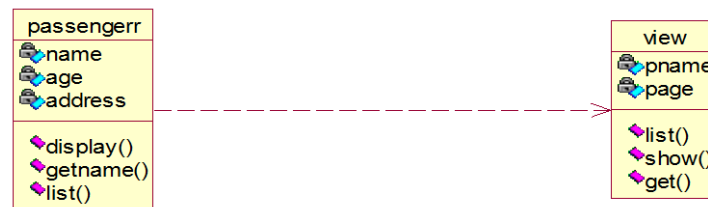


IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

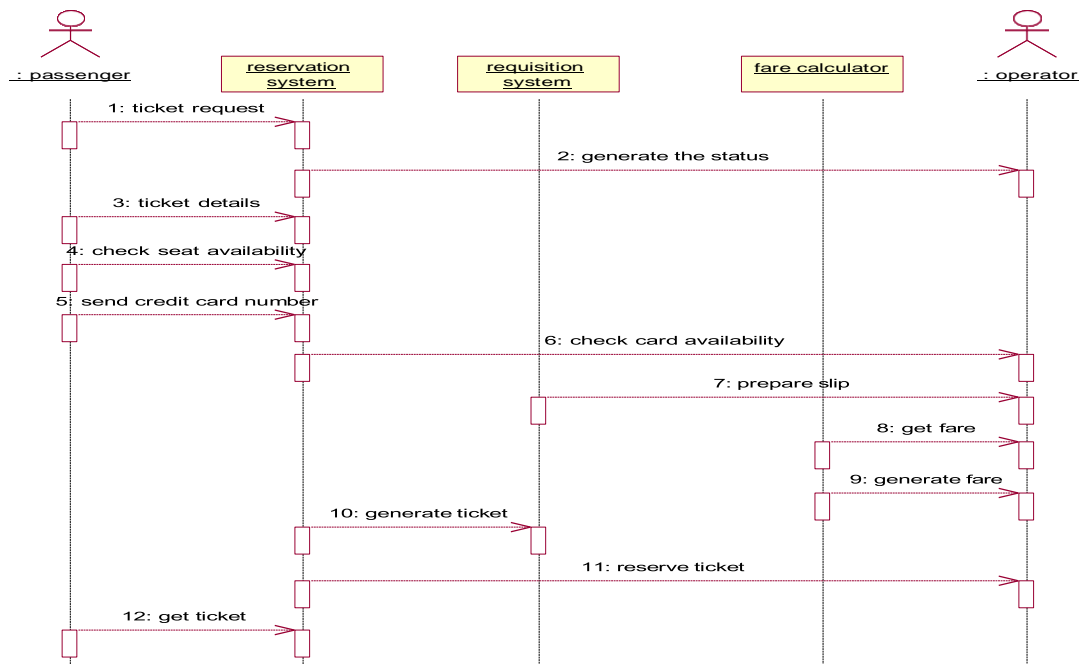
Identifying the Conceptual Classes using Category List

Conceptual class category	object
Role of people	Passenger, administer
Northworthy events	Reservation
Physical objects	Ticket
Description of things	Ticket details
Container of things	Railway Station, System
Things in a container	Ticket

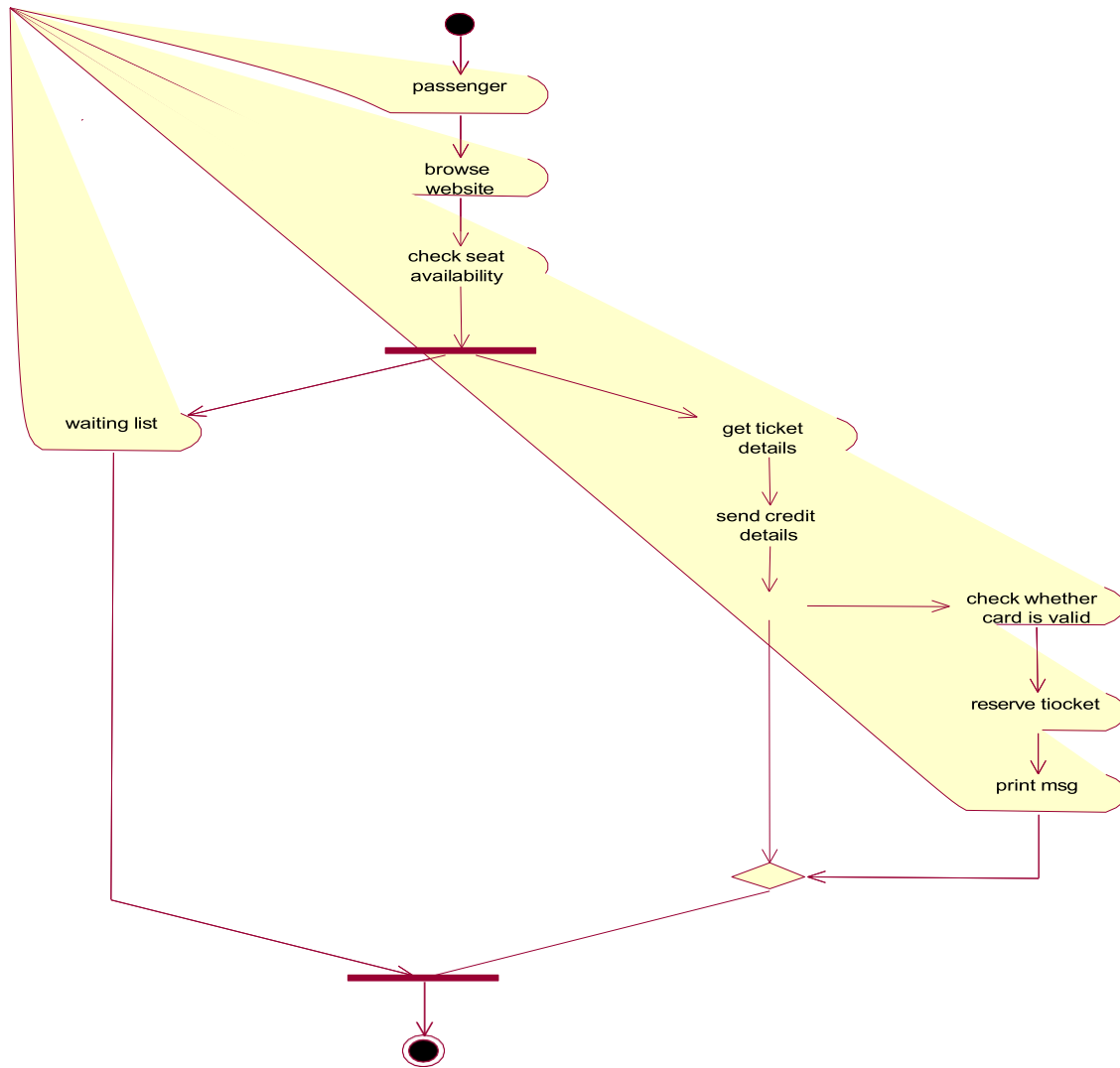
UML Class diagram



UML INTERACTION DIAGRAMS

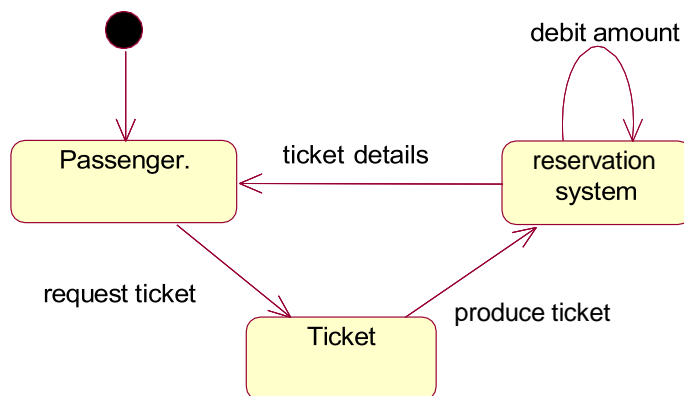


ACTIVITY DIAGRAM



Activity Diagram for E-ticketing

STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

Passenger.h

```
#ifndef PASSENGERR_H_HEADER_INCLUDED_AE9439ED
#define PASSENGERR_H_HEADER_INCLUDED_AE9439ED

class passengerr
{
public:
    display();
    getname();
    list();
private:
    name;
    age;
    address;
};
#endif /* PASSENGERR_H_HEADER_INCLUDED_AE9439ED */
```

RESULT:

Ex.No :7

Date: SOFTWARE PERSONNEL MANAGEMENT SYSTEM

AIM:

PROBLEM STATEMENT

Personnel Organizer Deluxe is flexible personnel management software for Windows users. Our software solution gives you an easy way to gather, organize and manage information about all employees in your company or organization. For the database novice, Organizer's intuitive interface and ready-to-use personnel management solutions make it easy to set up and use. Personnel Simple: complete database software solution that allows you to maintain data about all your employees in basic form.

Personnel Detailed: complete database software solution that allows you to maintain data about all your employees in as much detail as you want. For the power user, Organizer affords the simplicity of wizards that make it easy to set up and use employee management solutions or other database software solutions that you create.

Consider a software personnel management system the company or organization can provide the detail about the person who works in that organization. The company management can select the leader from one of the employee and give the work. The leader can select the worker to do the project. The person (user) collects the information about the employee who works in the organization. Collected information can be verified and submit to the management.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

Identify Use-case Diagram

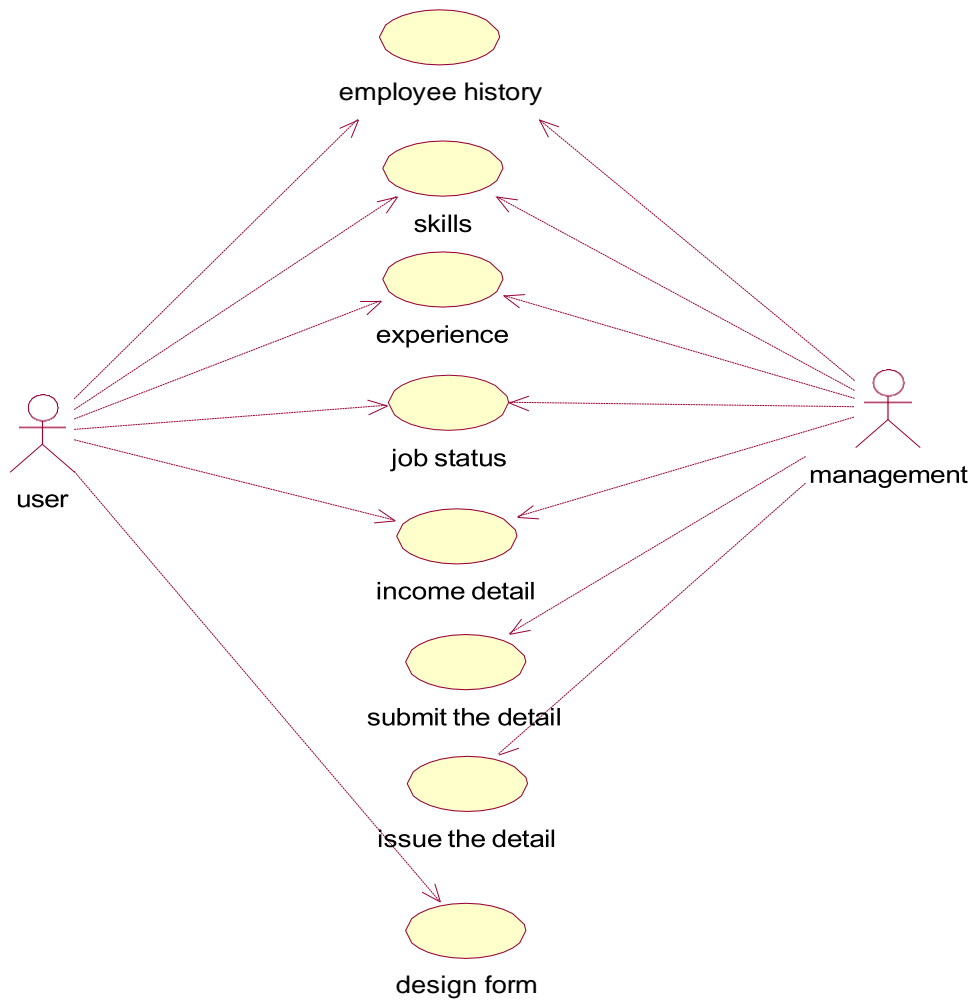
Identifying the use-case diagram involves the following steps.

- i. Choosing the system boundary.
 - ii. Identify the primary actor.
 - iii. Identify the primary goal for each primary actor.
 - iv. Define Use-case.
- i. **Choosing the System Boundary:**
- Hardware application : Computer
 - Software application : Browsers, c++
 - Person : user, worker
 - Organization : company
- ii. **Identify the Primary Actor:**
- User
 - Management

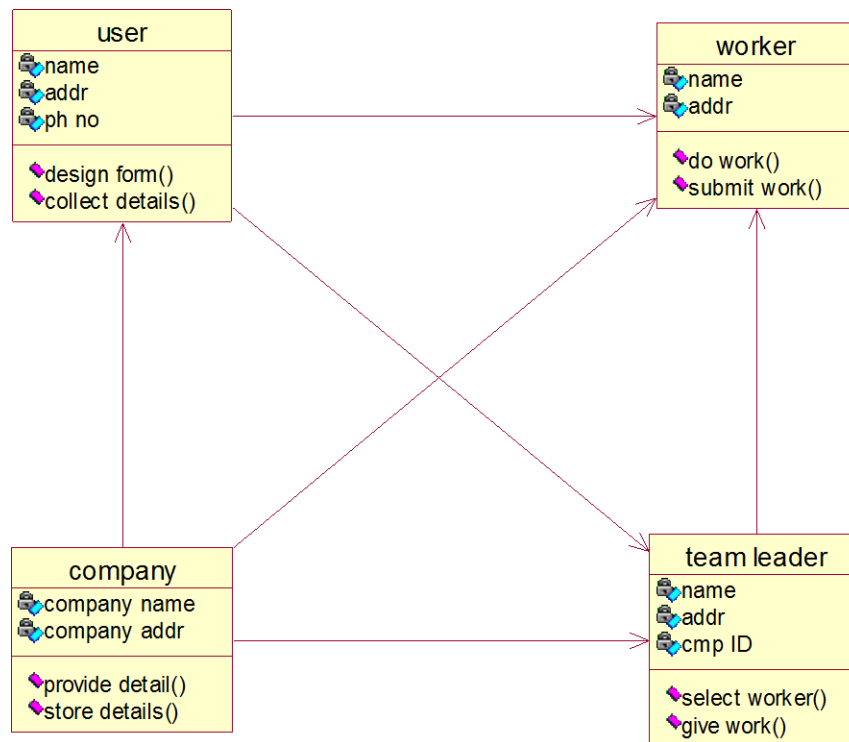
iii. Identify the goals for each primary actor:

S.No	Actor	Goals
1.	user	Select the employee, Get the details, Submit the details.
2.	management	Provide employee history, Select work leader, Give work to employee.

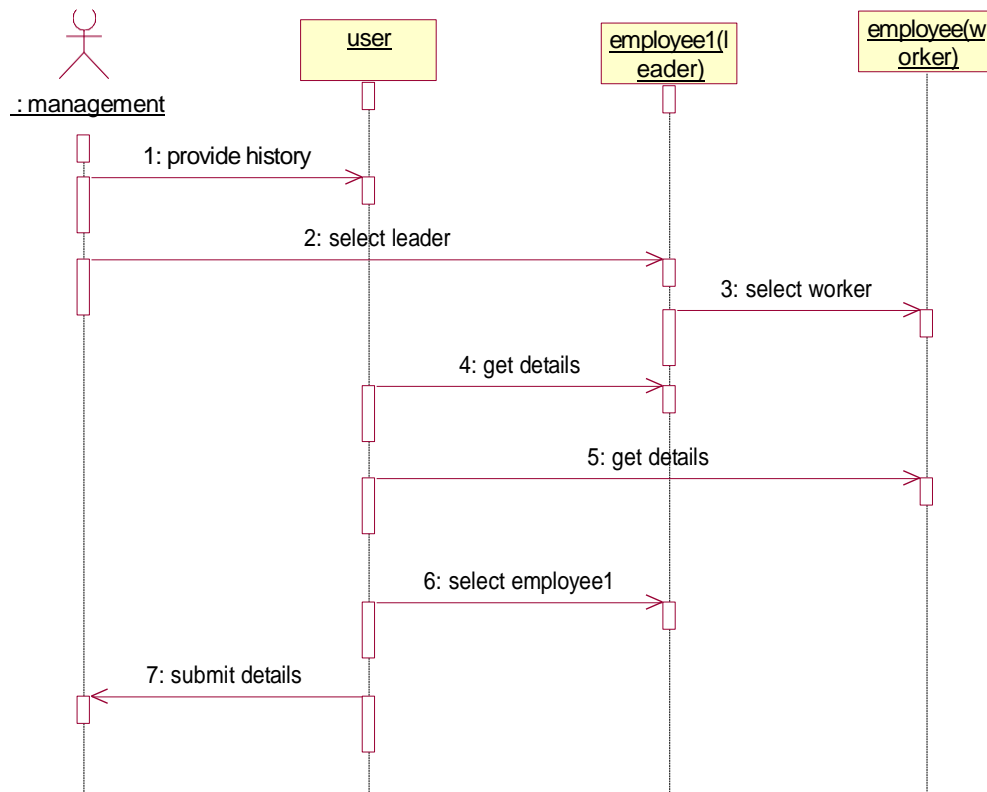
Use-Case Diagram for Software personnel management system:



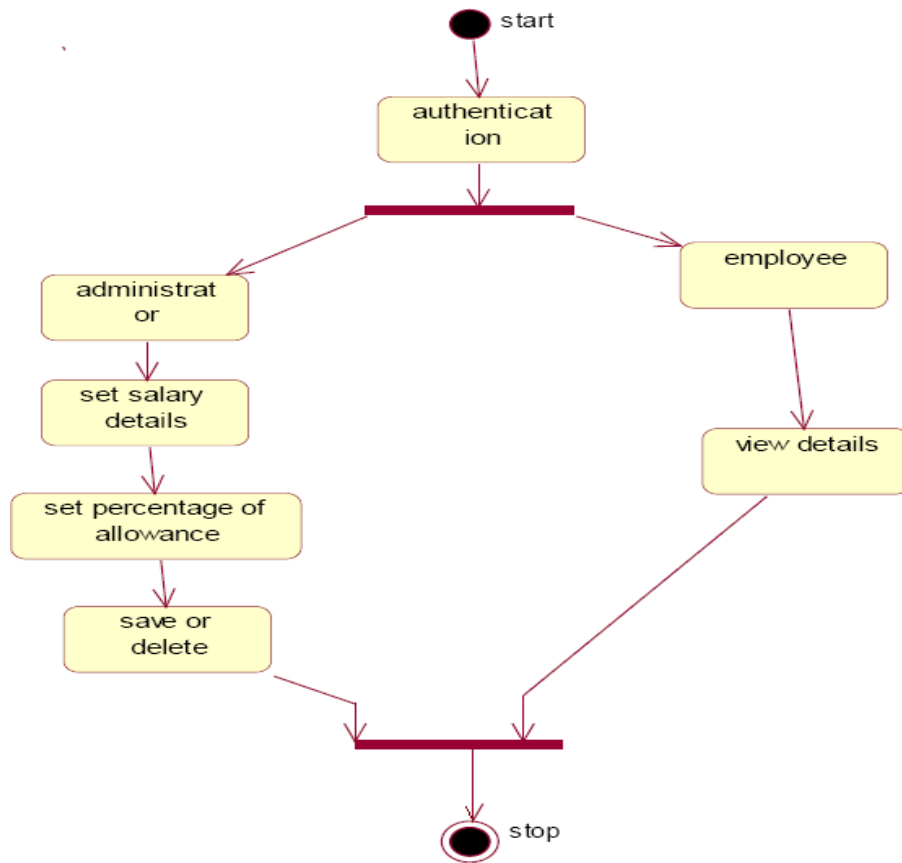
IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM



UML INTERACTION DIAGRAMS

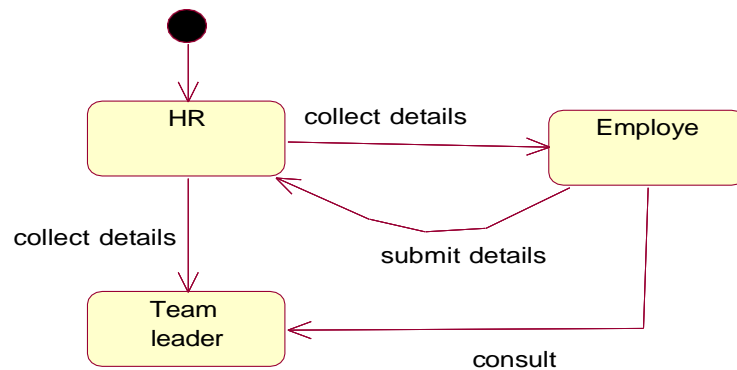


ACTIVITY DIAGRAM



Activity Diagram for Software personnel management system

STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

```
#include "company.h"

company::provide details()

{}

company::store details()

{}

#ifndef COMPANY_H_HEADER_INCLUDED_AE94254B
#define COMPANY_H_HEADER_INCLUDED_AE94254B

class company{

public:

    provide details();

    store details();

private:

    name;

    address;

};

#endif /* COMPANY_H_HEADER_INCLUDED_AE94254B */
```

RESULT:

Ex.No :8

Date:

CREDIT CARD SYSTEM

AIM:

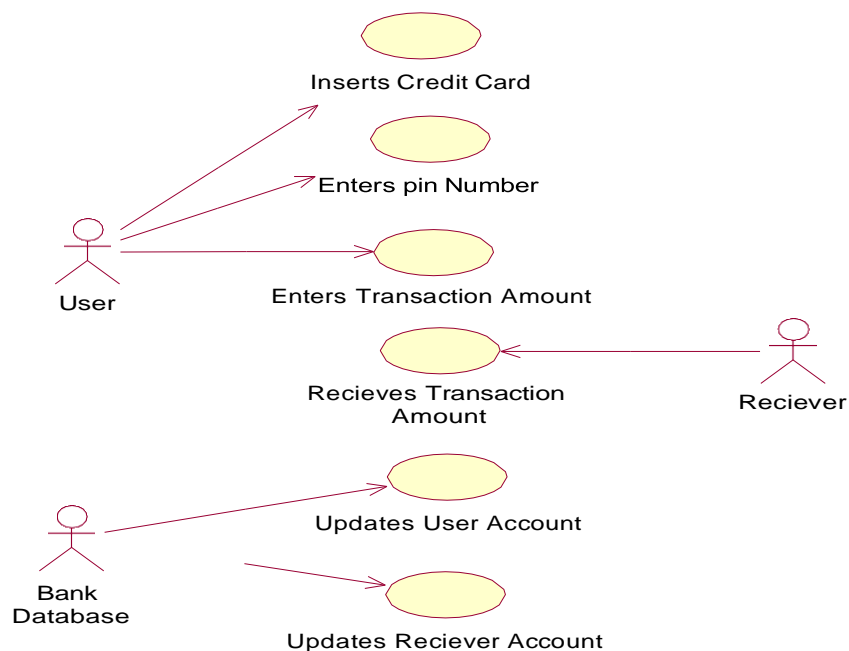
PROBLEM STATEMENT

A credit card is a small plastic card issued to users as a system of payment. It allows its holder to buy goods and services based on the holder's promise to pay for these goods and services. The issuer of the card creates a revolving account and grants a line of credit to the consumer from which the user can borrow money for payment to a merchant or as a cash advance to the user.

A credit card is different from a charge card: a charge card requires the balance to be paid in full each month. In contrast, credit cards allow the consumers a continuing balance of debt, subject to interest being charged. A credit card also differs from a cash card, which can be used like currency by the owner of the card.

- The User inserts the credit card into the system.
- The User then enters the pin Number.
- The Pin number is then validated.
- The amount to be transacted is then entered
- The transaction is then done.
- The account of the sender and receiver are validated

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

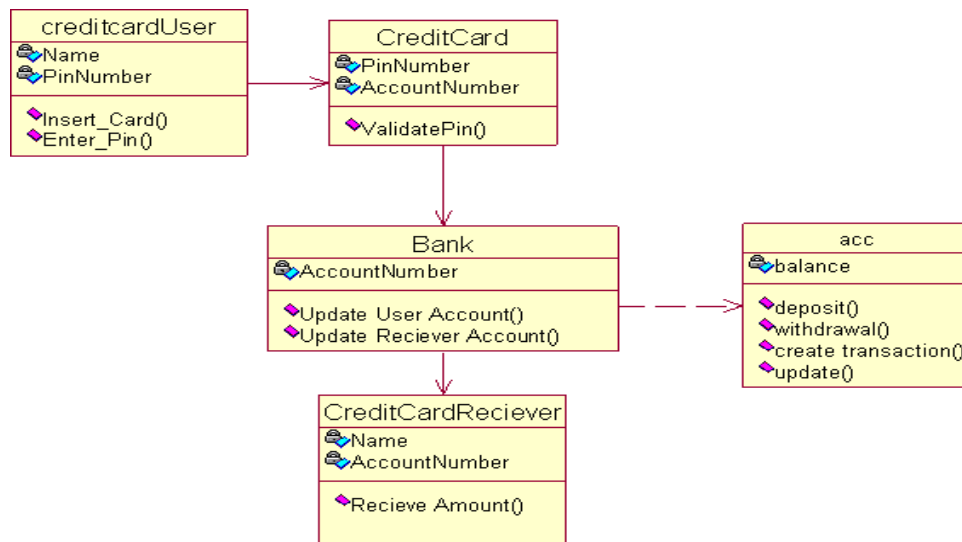


IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

Identifying the Conceptual Classes using Category List

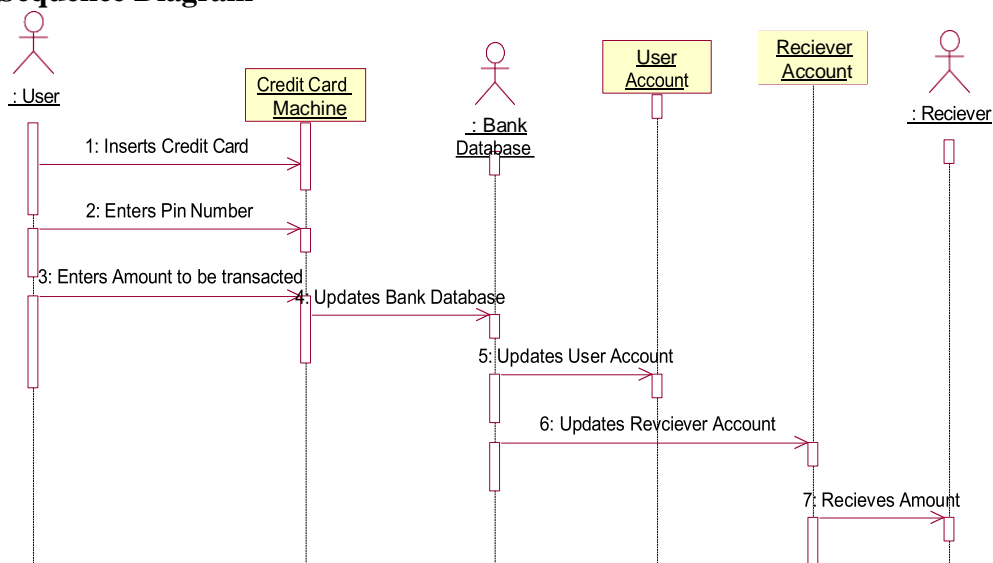
S.No	Category	Example
1.	Roles of People	User, Receiver
2.	Place of Event	Shop
3.	Physical Objects	Credit Card, Machine
4.	Container of Things	Bank

UML Class diagram

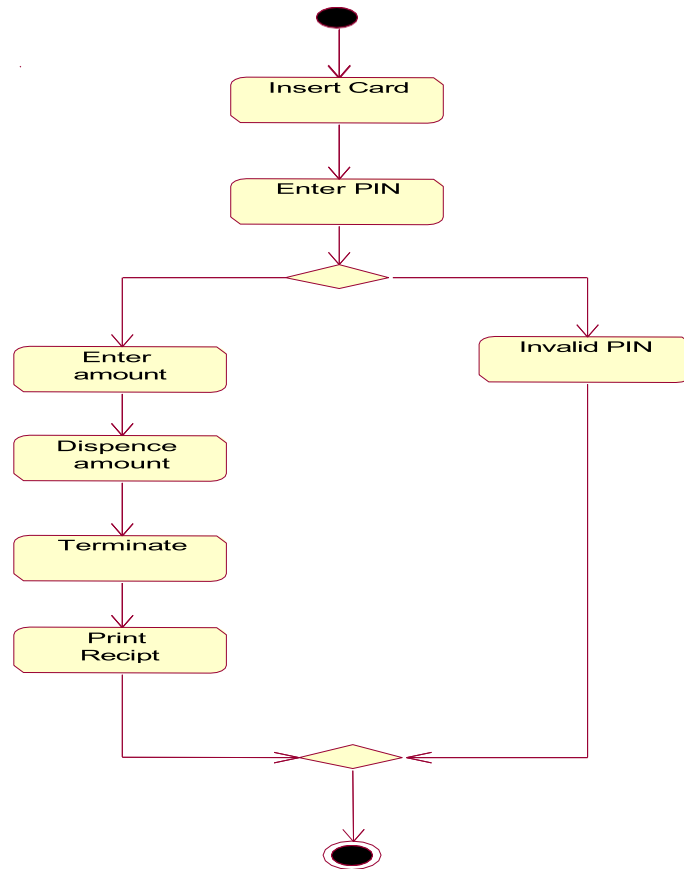


UML INTERACTION DIAGRAMS

UML Sequence Diagram

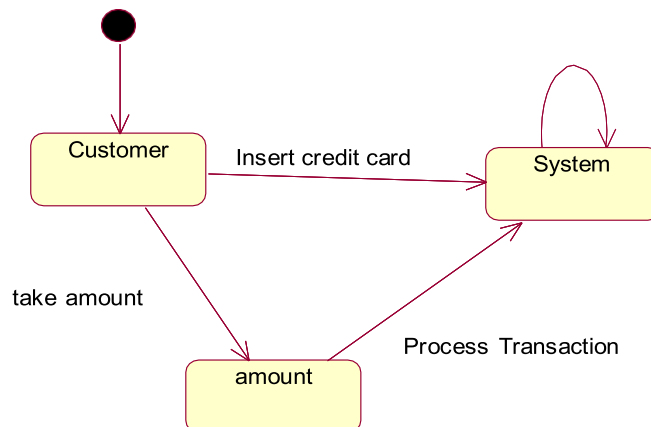


ACTIVITY DIAGRAM



Activity Diagram for Credit card processing

STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

```
/**
 * Class customer
 */
public class customer {

    public void name;
    public void age;
    // Constructors
    public customer () { };

    public void setName ( void newVar ) {
        name = newVar;
    }

    public void getName ( ) {
        return name;
    }
}
```

RESULT:

Ex.No :9

Date:

E-BOOK MANAGEMENT SYSTEM

AIM:

PROBLEM STATEMENT

- User can login to the system to download book.
- The users send the request message to the system.
- From the system the request message will be send to the server.
- Thus the users search the book with the help of browser.
- The server accepts the request message and searches the result for the same.
- The corresponding response message will be send from the server to the user.
- With the help of the response message the user can download the book what they need.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

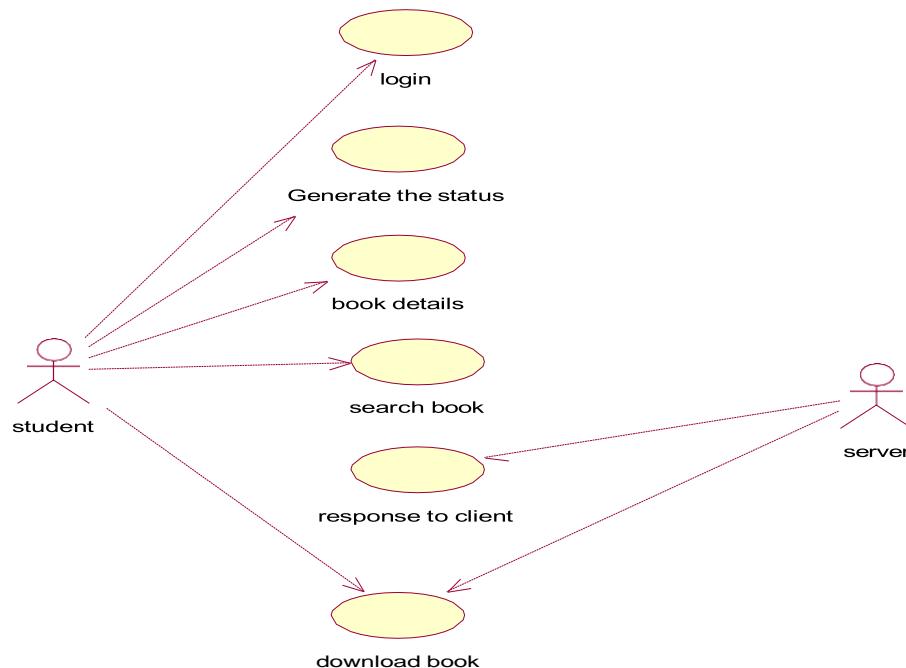
Actors Involved:

- student
- server

Identify the goals for each primary actor:

Actor	Goal
Student	1.login 2.generate the status 3.search book 4. Downloads book
Server	1.Respose the client

Use-Case Diagram for e-book management system

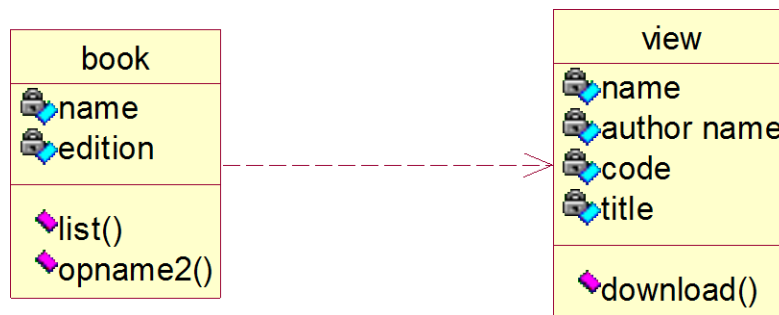


IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

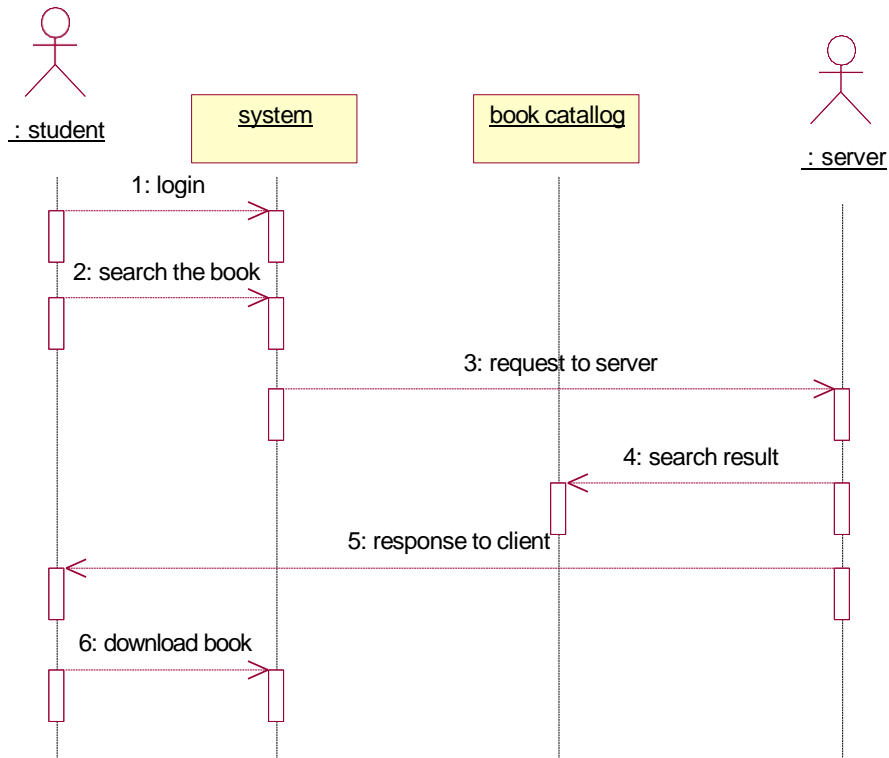
Identifying the Conceptual Classes using Category List

Conceptual class category	Object
Role of people	Student
Place of transaction	System
Physical objects	Book,system
Container of things	System
Things in a container	Book

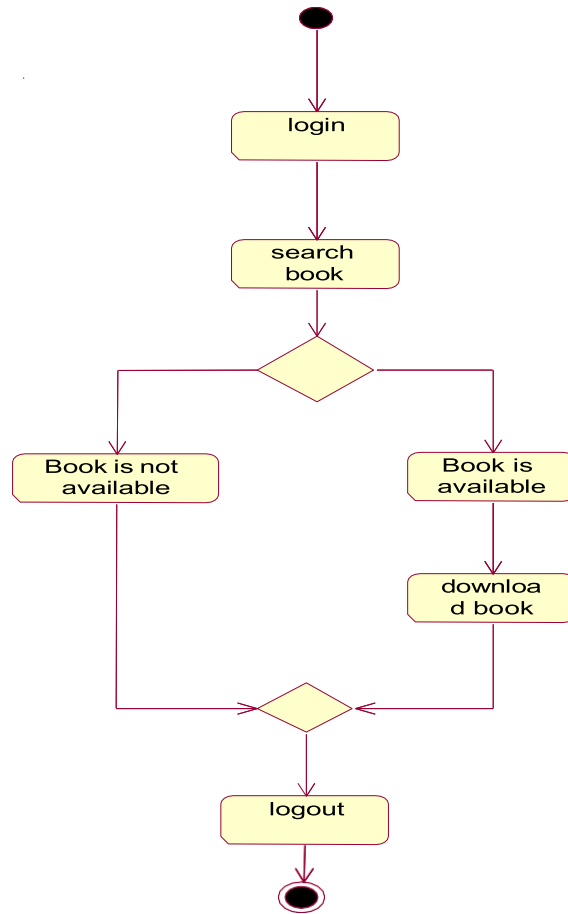
UML Class diagram



UML INTERACTION DIAGRAMS

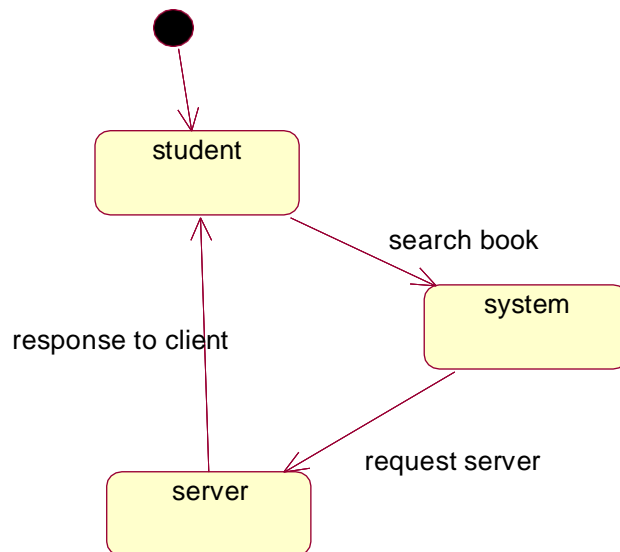


ACTIVITY DIAGRAM



Activity Diagram for e-book management system

STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

```
public class customer {  
  
    public void name;  
    public void age;  
  
    // Constructors  
    public customer () { };  
  
    // Methods  
  
    // Accessor methods  
    * Set the value of name  
    * @param newVar the new value of name  
    */  
    public void setName ( void newVar ) {  
        name = newVar;  
    }  
    * Get the value of name  
    * @return the value of name  
    */  
    public void getName ( ) {  
        return name;  
    }  
}
```

RESULT:

Ex.No :10

Date:

RECRUITMENT SYSTEM

AIM:

PROBLEM STATEMENT

- A Candidate refers job portals, advertisements and other networking referrals and
- Applies for the required job by submitting a bio-data or profile information.
- The company HR sends the call letter.
- The candidates are subjected to a list of tests starting from initial screening to personal interview.
- The results of each test are validated in the company database and the candidates are short listed.
- The finalist's pre-employment records are checked and the appointment order is given.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

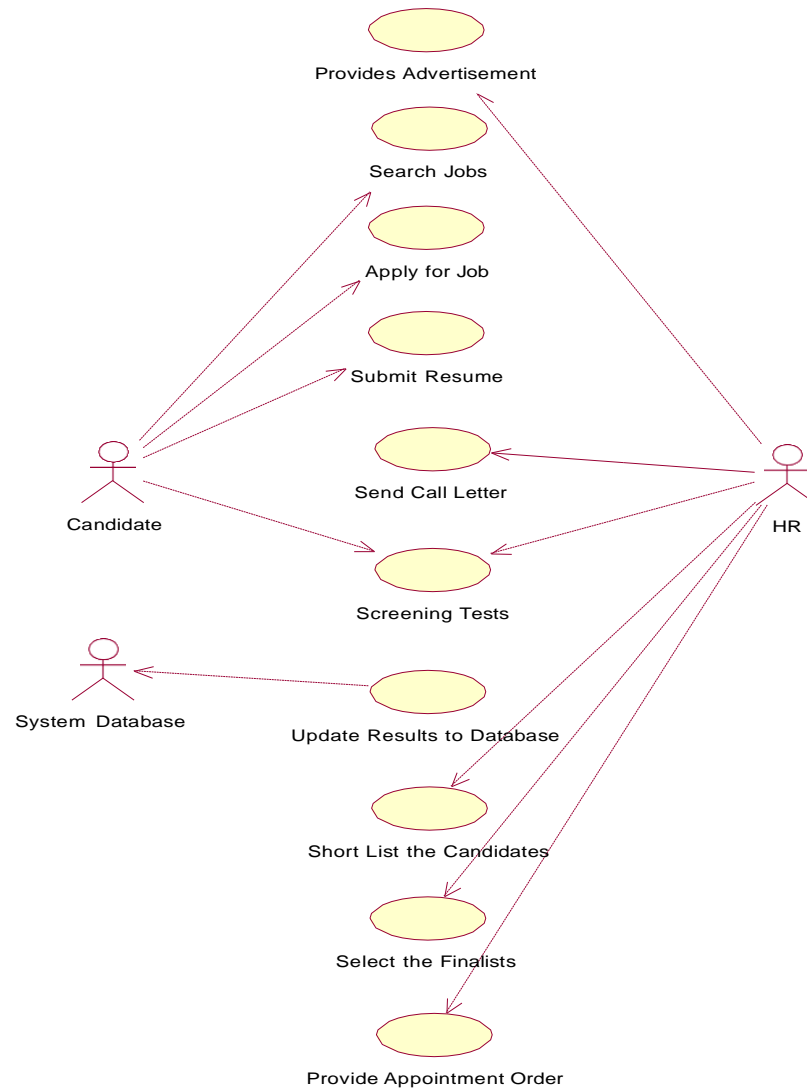
Actors Involved:

- Candidate.
- HR
- System Database

iii. Identify the goals for each primary actor:

S.No	Actor	Goals
1.	Candidate	Searches Job, Applies for Job, Provides Bio-data, Attends Screening Tests
2.	HR	Provides Advertisements, Sends Call Letters, Conducts Screening Tests, Updates results to System Database, Shortlists the Candidates, Selects the finalists, Provides Appointment Order.
3.	System Database	Stores the results.

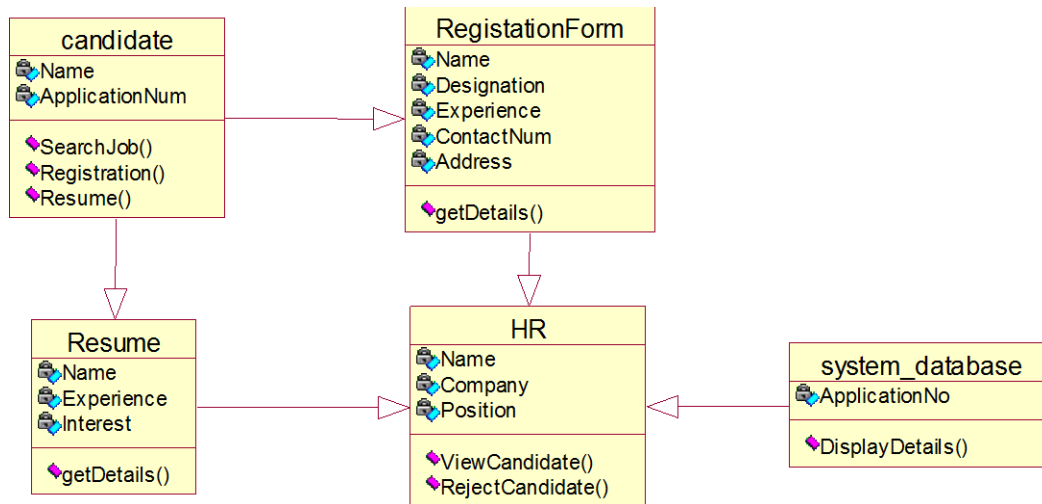
Use-Case Diagram for Recruitment system



IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

Identifying the Conceptual Classes using Category List

S.No	Category	Example
1.	Roles of People	Candidate, HR
2.	Place of Event	Company
3.	Physical Objects	Registration Form, Resume
4.	Description of Things	Candidate Bio data
5.	Catalogs	Company Details
6.	Container of Things	Company
7.	Schedule Manuals	Interview schedule time

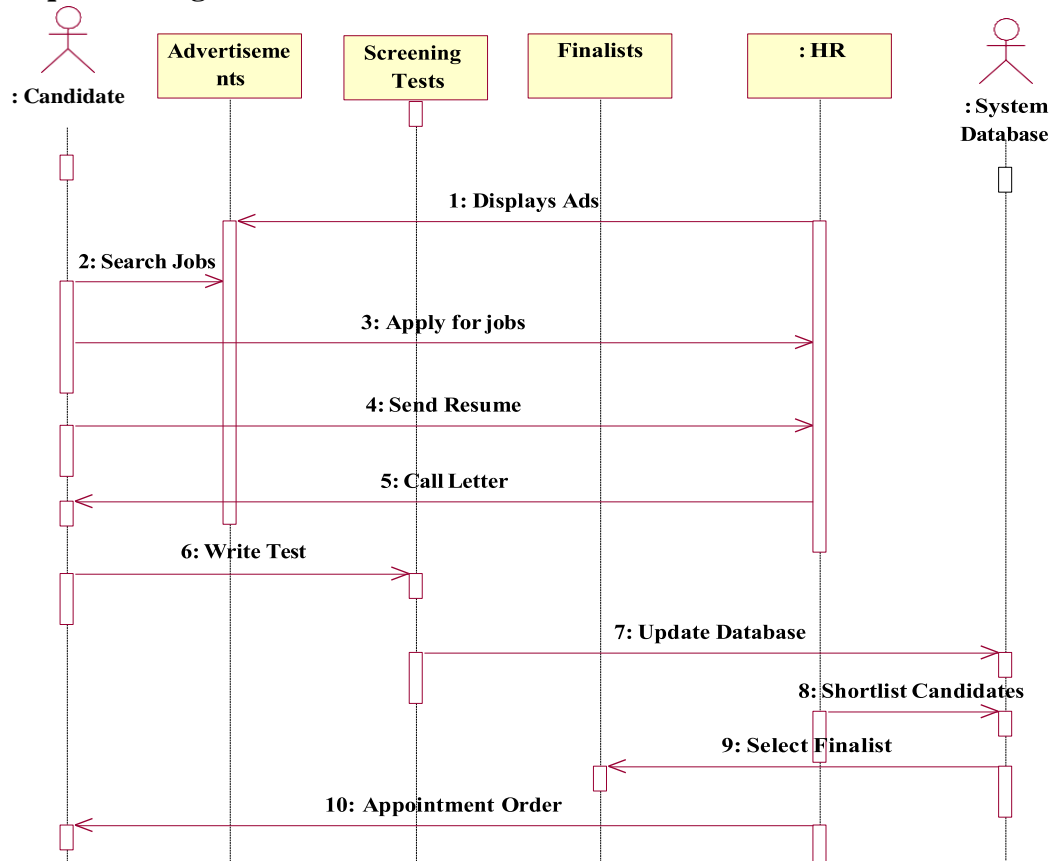


UML INTERACTION DIAGRAMS

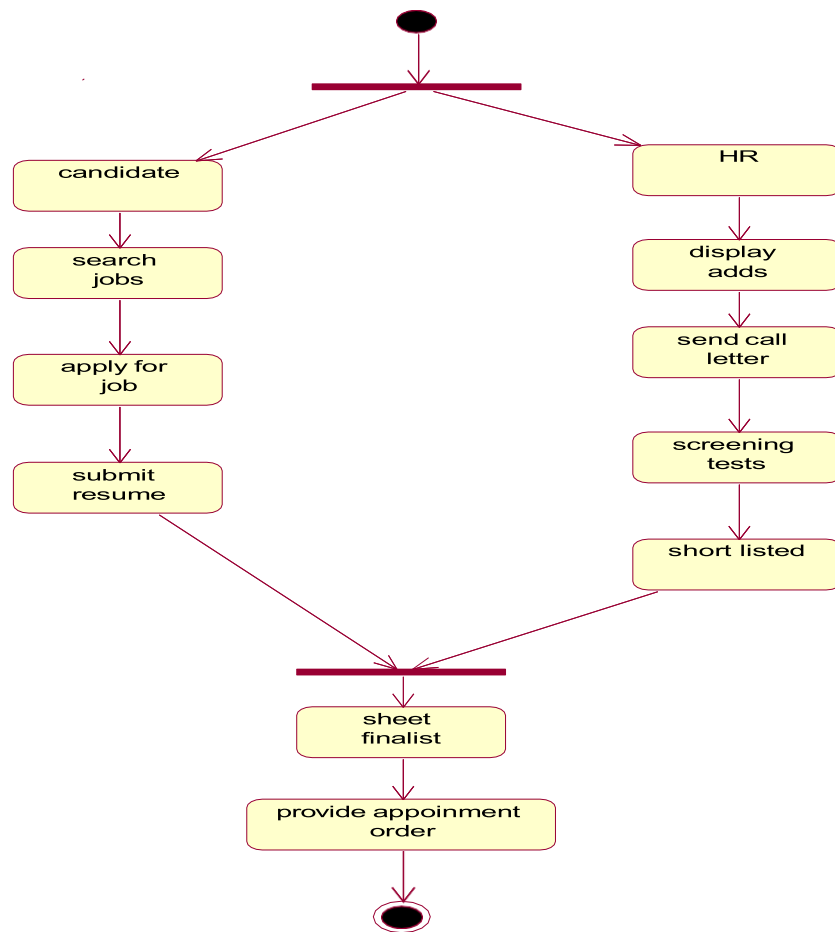
Interaction diagrams come in two forms, both present in the UML.

- Sequence Diagram
- Collaboration Diagram

UML Sequence Diagram

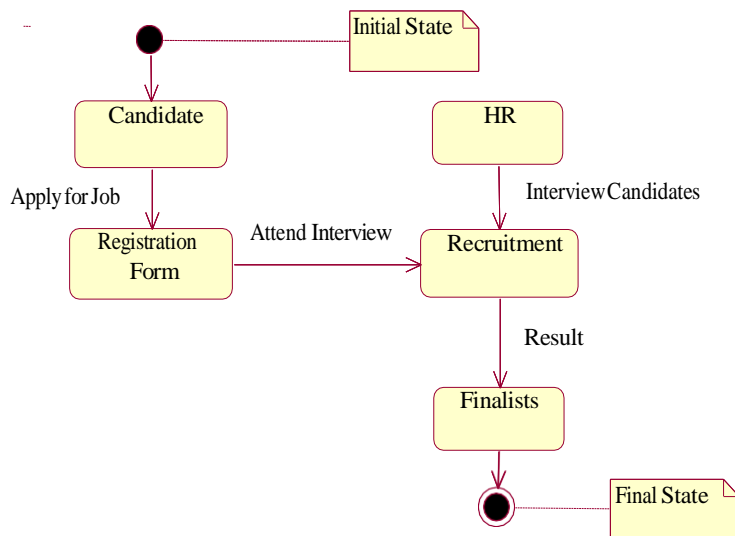


ACTIVITY DIAGRAM



Activity Diagram for Recruitment system

STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

Class Registrationform

```
public class Registrationform {  
    public void Name1;  
    public void Designation;  
    // Constructors  
    public Registrationform () { };  
    // Methods  
    // Accessor methods  
    * Set the value of Name1  
    * @param newVar the new value of Name1  
    */  
    public void setName1 ( void newVar ) {  
        Name1 = newVar;  
    }  
}
```

RESULT:

Ex.No:11

Date:

CONFERENCE MANAGEMENT SYSTEM

AIM:

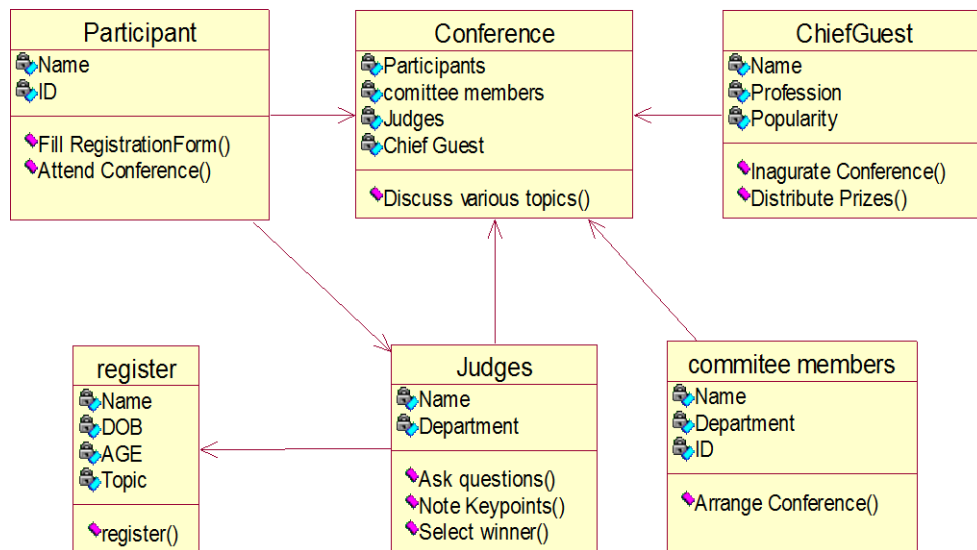
PROBLEM STATEMENT

- Announce the date of the conference
- Members and panel arrive
- Participants of the conference should register and pay the registration fees.
- The member start their discussion
- Final versions of papers that are accepted should be collected. .
- Collecting final accepted versions
- Finally, support to participants during the conference duration should be provided
- The head of the conference will tell the area of interest
- The member starts their discussion
- The head gather their options and note it down
- Selecting the best of all options
- Identify the requirements
- Implement them in their domain
- Develop a successful project

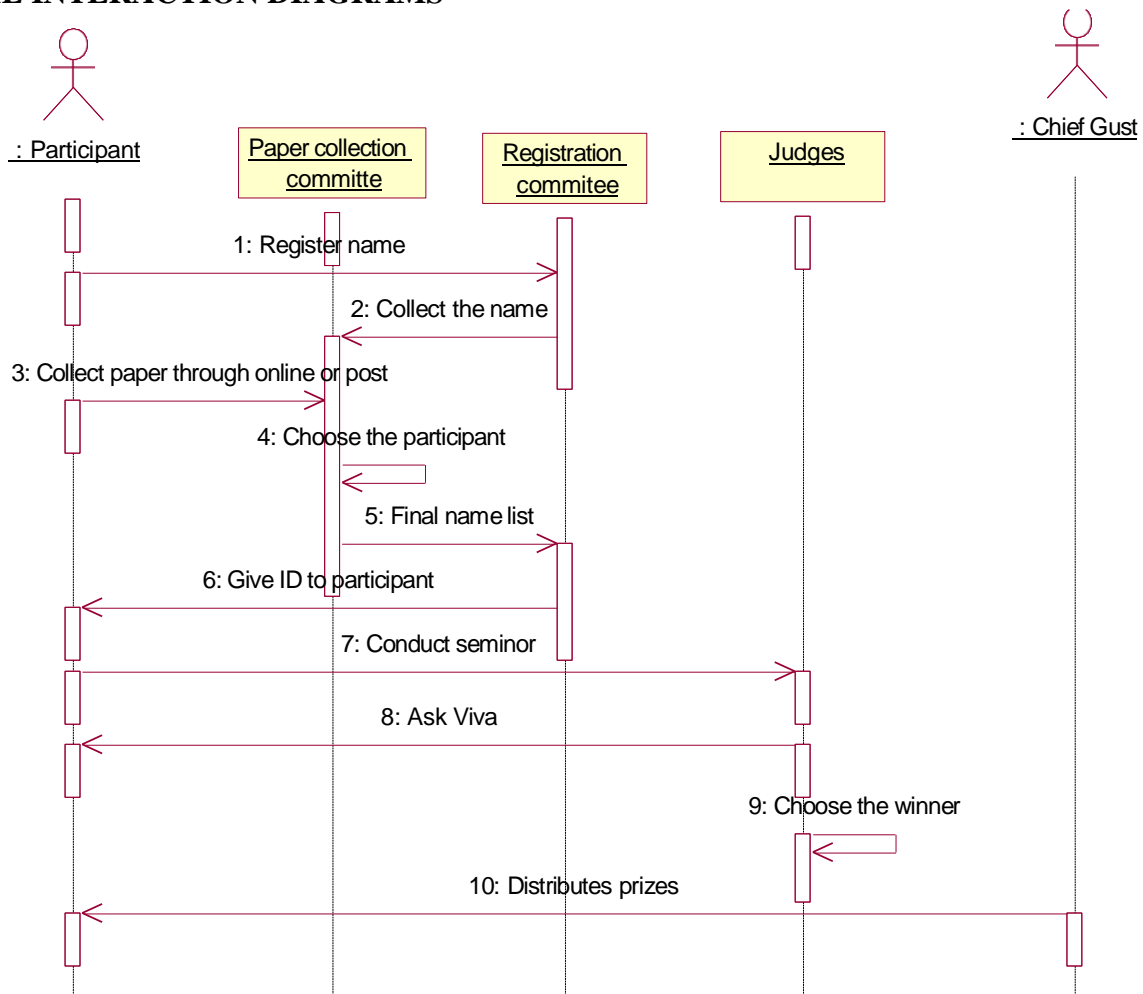
IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL



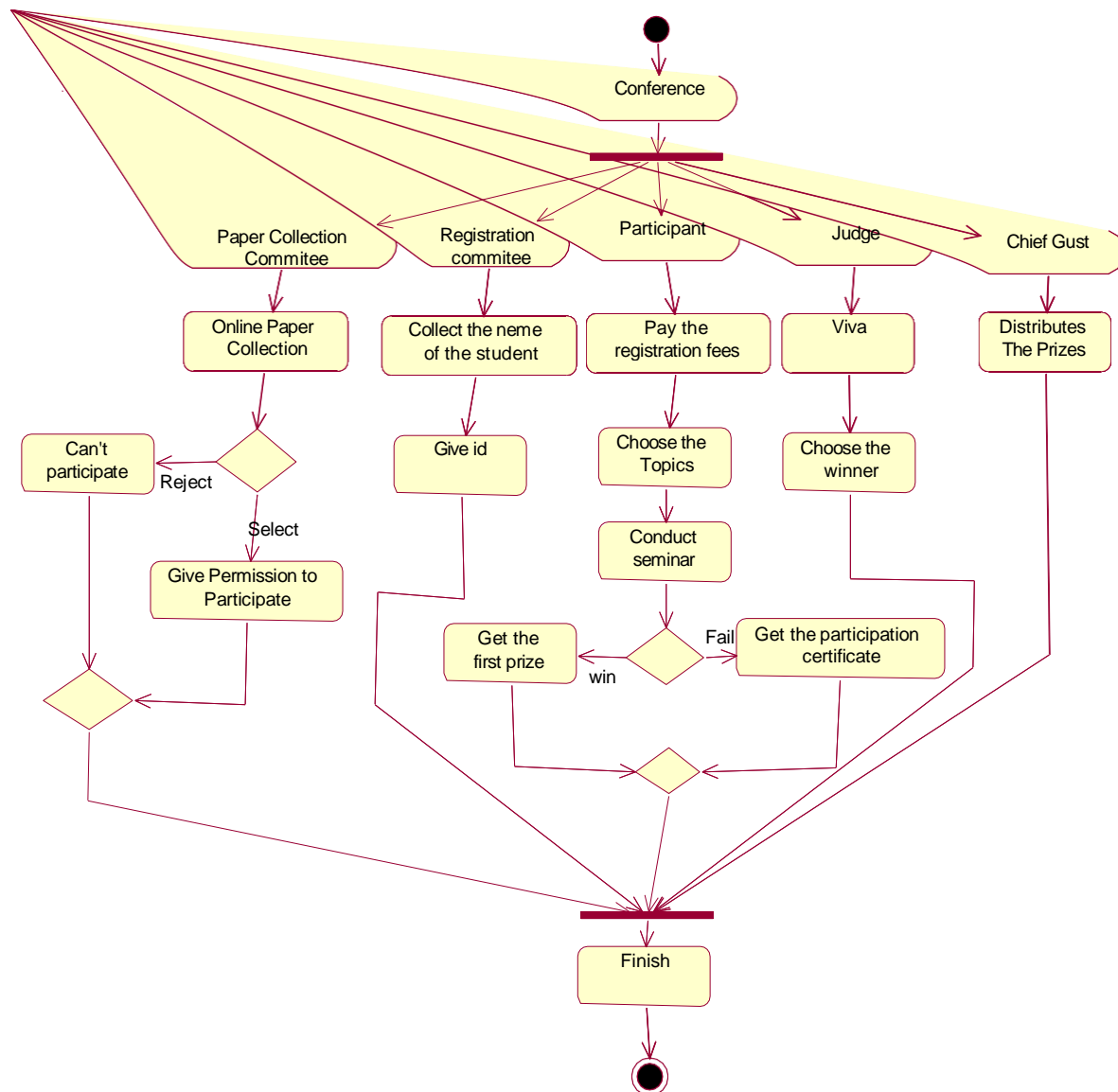
IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM



UML INTERACTION DIAGRAMS

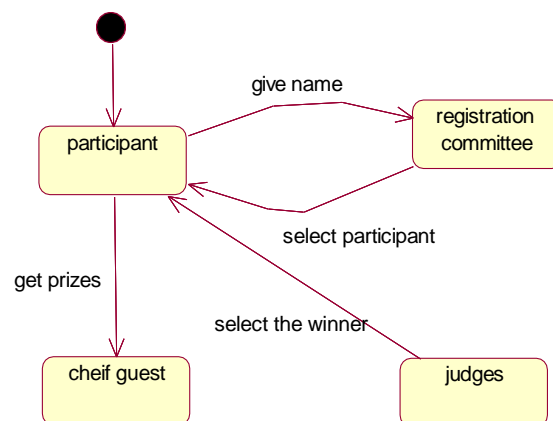


ACTIVITY DIAGRAM



Activity Diagram for Conference management system

STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

* Class Conference

*/

```
public class Conference {
```

```
    public void Participants;
```

```
    public void CommitteeMembers;
```

```
    public void Judges;
```

```
    public void ChiefGuests;
```

```
    public Conference () { };
```

* Set the value of Participants

* @param newVar the new value of Participants

*/

```
    public void setParticipants ( void newVar ) {
```

```
        Participants = newVar;
```

```
    }
```

RESULT:

Ex.No:12

Date:

BPO MANAGEMENT SYSTEM

AIM:

PROBLEM STATEMENT

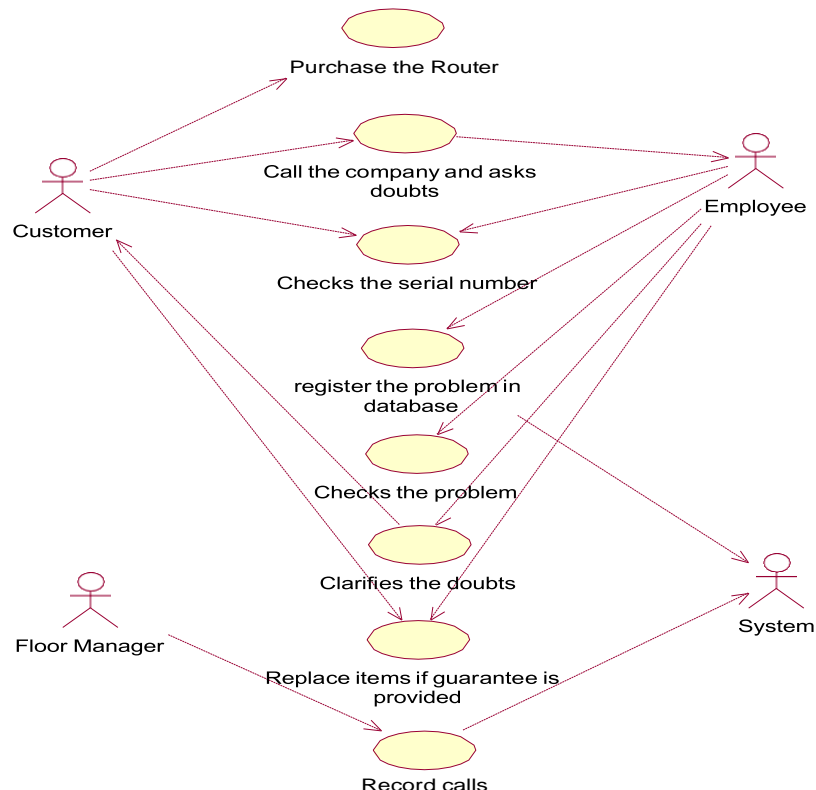
A BPO management system is a centralized office used for the purpose of receiving and transmitting a large volume of requests by telephone. A call centre is operated by a company to administer incoming product support or information inquiries from consumers. Outgoing calls for telemarketing, clientele, product services, and debt collection are also made. In addition to a call Centre, collective handling of letters, faxes, and e-mail at one location is known as a call center.

A BPO system is often operated through an extensive open workspace for call center agents, with work stations that include a computer for each agent, a telephone set/headset connected to a telecom switch one or more supervisor stations.

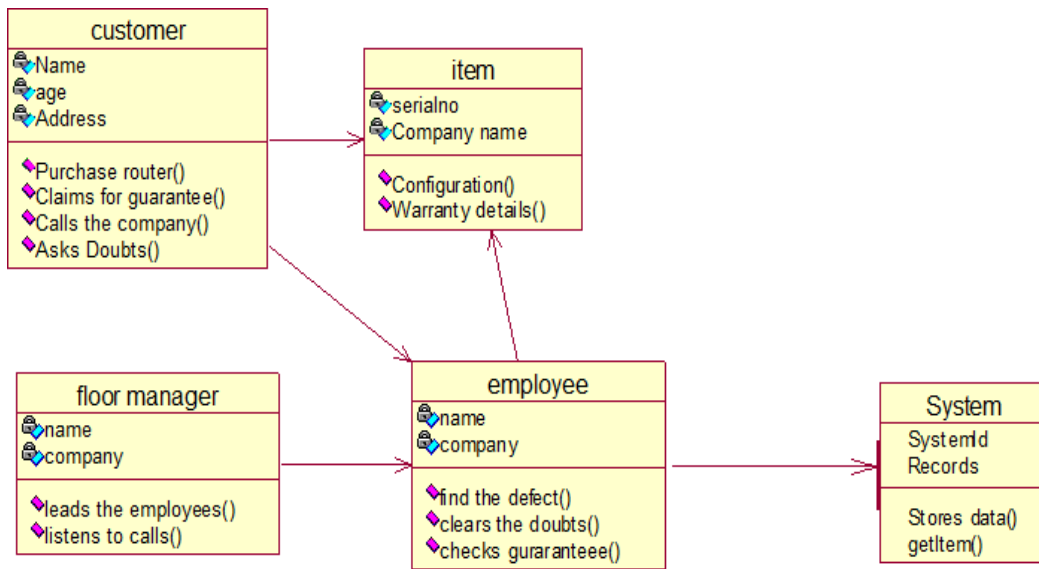
It can be independently operated or networked with additional centers, often linked to a corporate computer networks, including, microprocessors microcomputers and LAN's. Increasingly, the voice and data pathways into the centre are linked through a set of new technologies called computer telephone Integration (CTI).

Most major businesses use BPO management system to interact with their customers. Examples include utility companies, mail order catalogue retailers, and customer support for computer hardware and software. Some businesses even service internal functions through call centers.

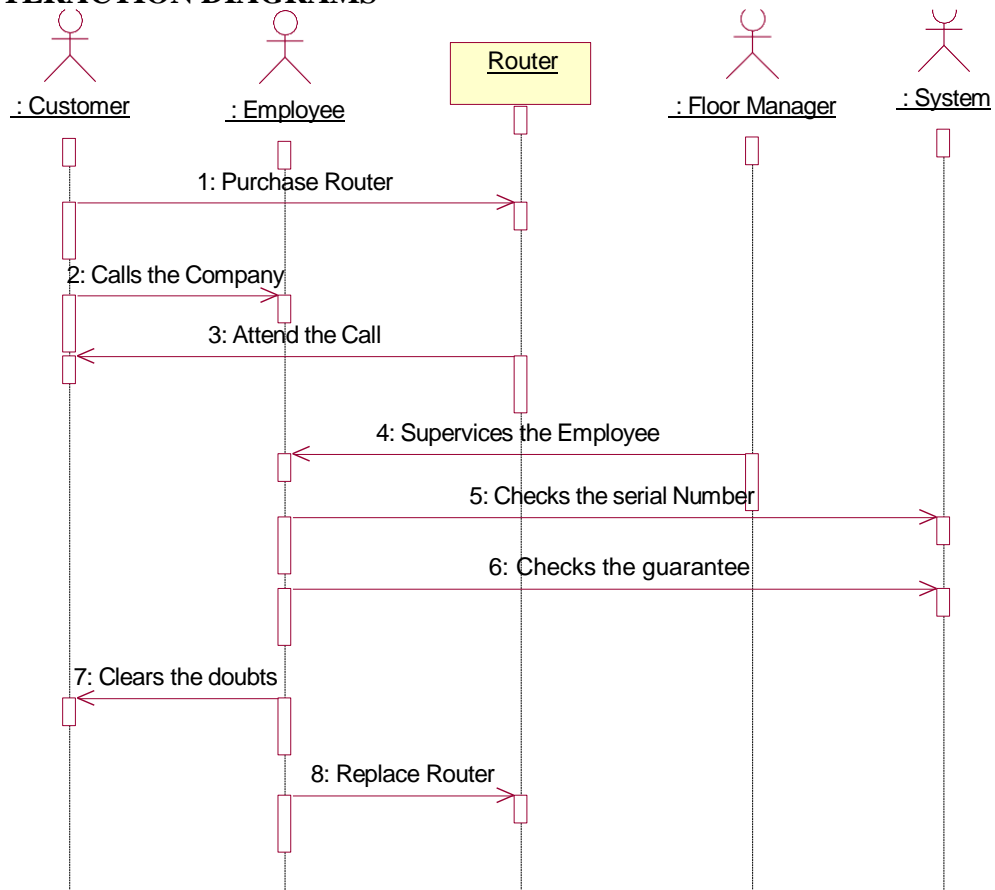
IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL



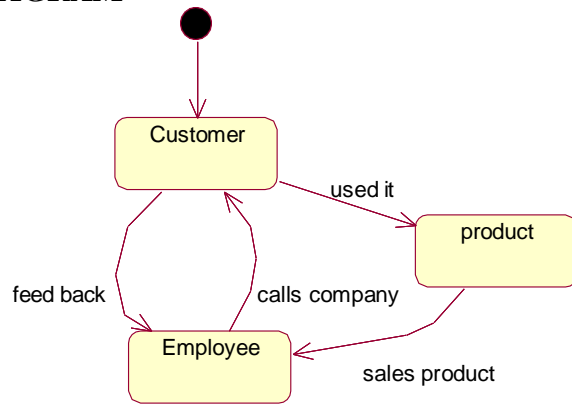
IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM



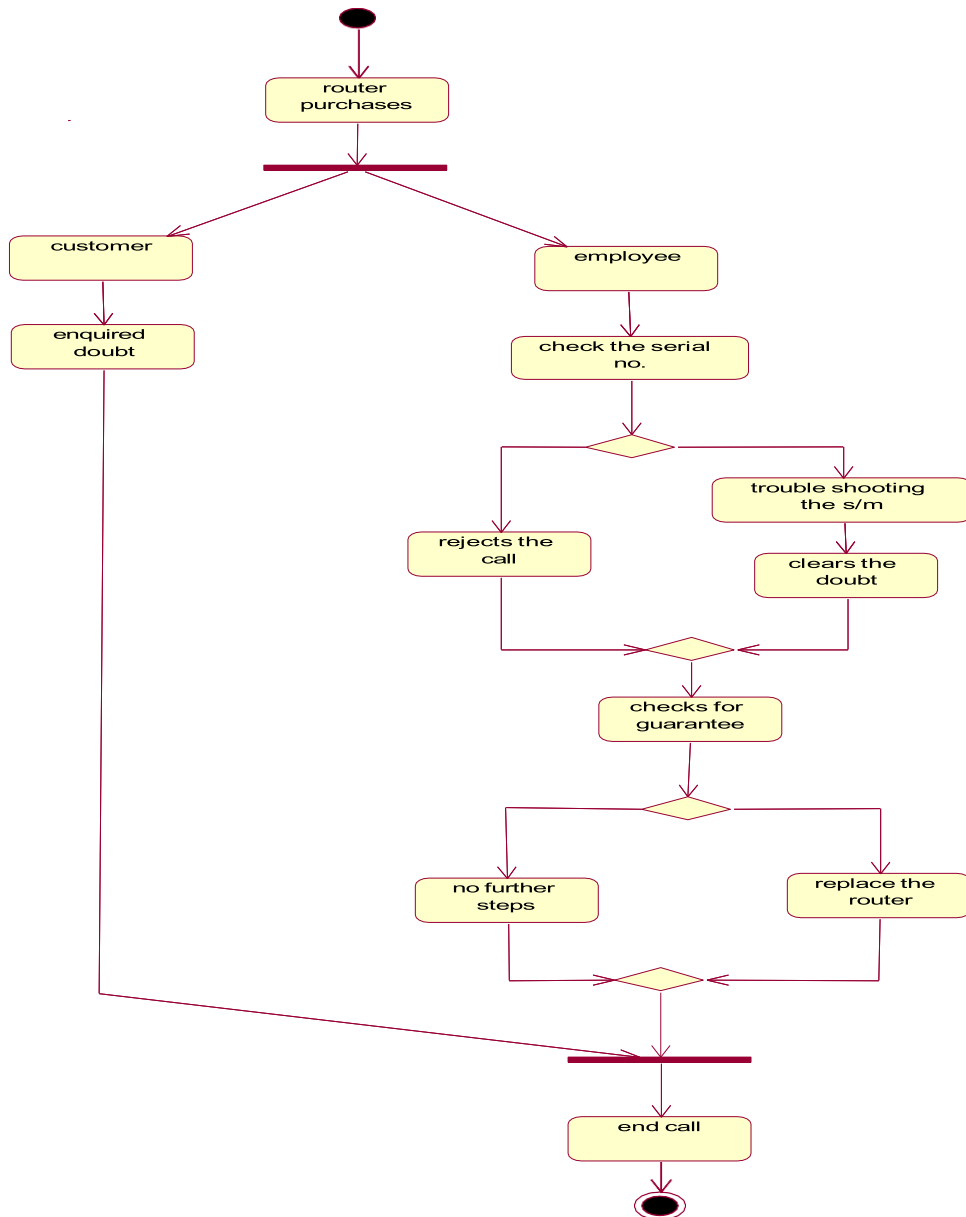
UML INTERACTION DIAGRAMS



STATE CHART DIAGRAM



ACTIVITY DIAGRAM



IMPLEMENTATION OF THE CODE

* Class customer

*/

```
public class customer {
```

```
    public void name;
```

```
    public void age;
```

```
    //
```

```
    // Constructors
```

```
    //
```

```
    public customer () { };
```

```
    public void setName ( void newVar ) {
```

```
        name = newVar;
```

```
    }
```

```
    public void getName ( ) {
```

```
        return name;
```

```
    }
```

RESULT:

Ex.No :13

Date:

FOREIGN TRADING SYSTEM

AIM:

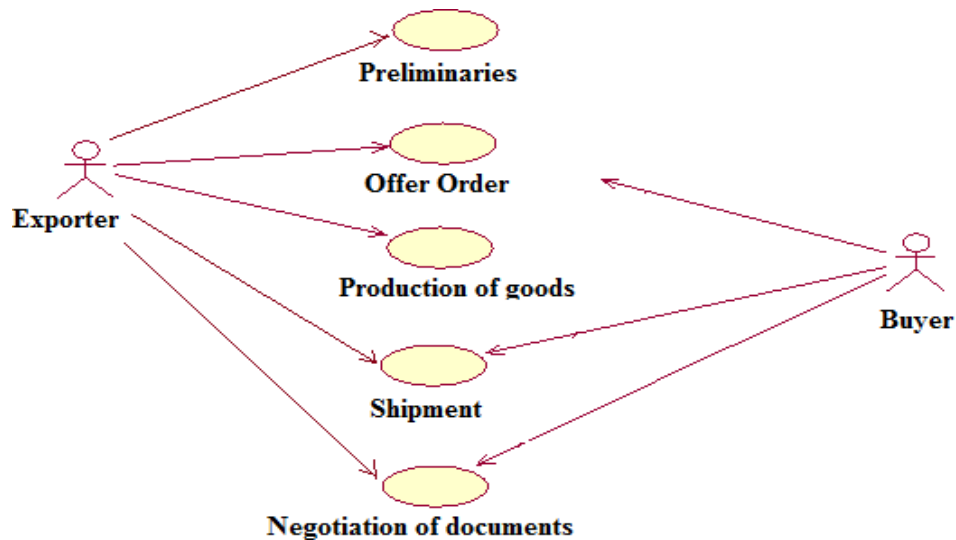
PROBLEM STATEMENT

The main activity of international marketing is the export-import procedure. This procedure involves the actual and operational procedure of export and import trade. It also involves documentation, procedures, rules and regulations imposed by both the exporting and importing countries. These procedures include excise clearance, foreign exchange, etc.

Foreign trading system is the interface between the exporter and buyer. It aims at improving the efficiency in the production, export process and reduce the complexities involved in it to the maximum possible extent.

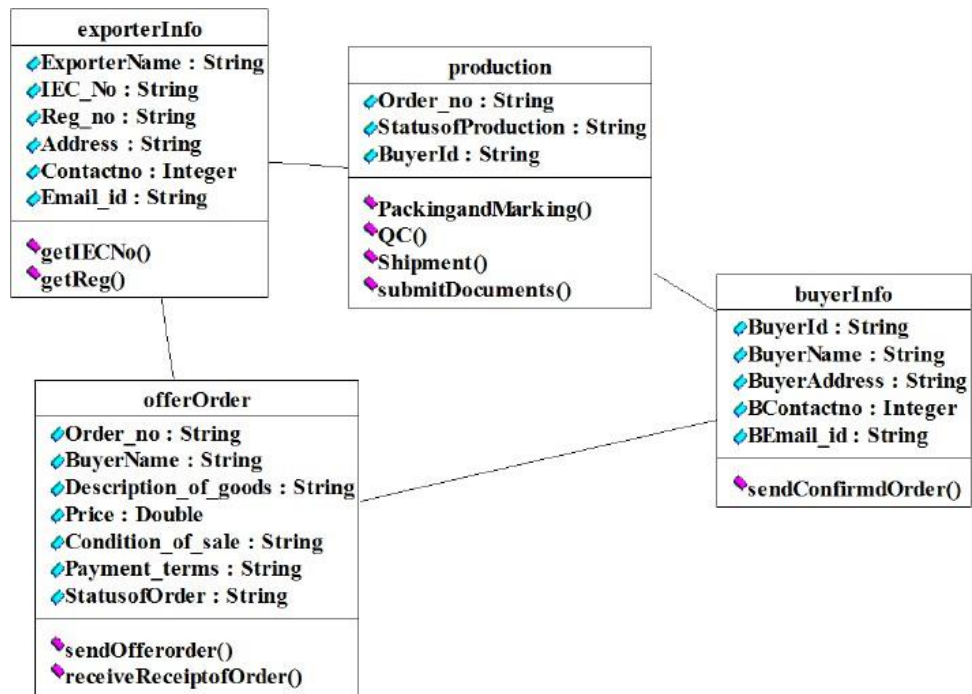
The System provides an online interface to the buyer where they can fill in their personal details and submit the necessary documents (may be by scanning). The authority concerned with the production and shipment of goods can use this system to reduce his workload and process the application in a speedy manner.

IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL

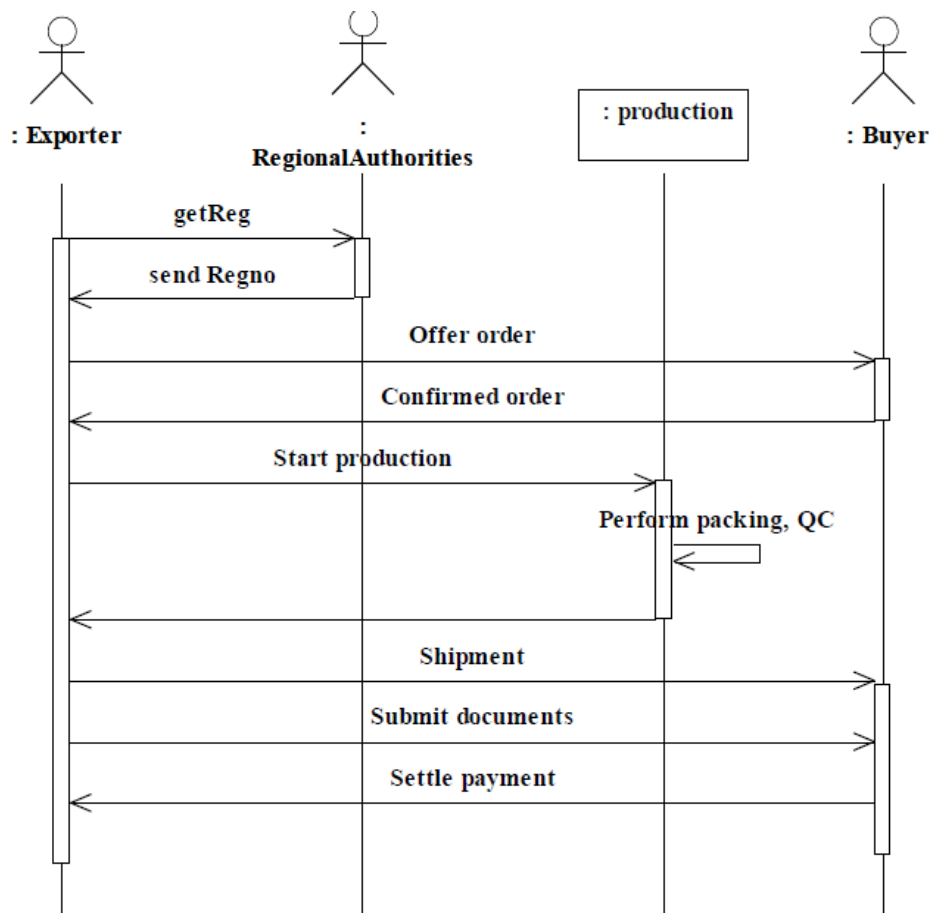


IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM

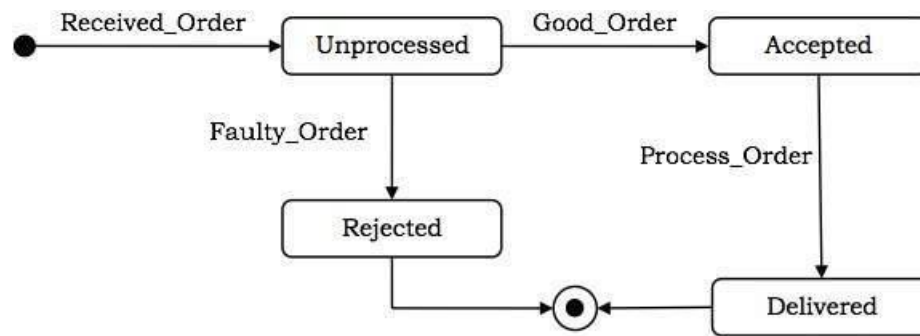
A class diagram is a picture for describing generic descriptions of possible systems. Class diagrams and collaboration diagrams are alternate representations of object models. Class diagrams contain classes and object diagrams contain objects, but it is possible to mix classes and objects when dealing with various kinds of metadata, so the separation is not rigid.



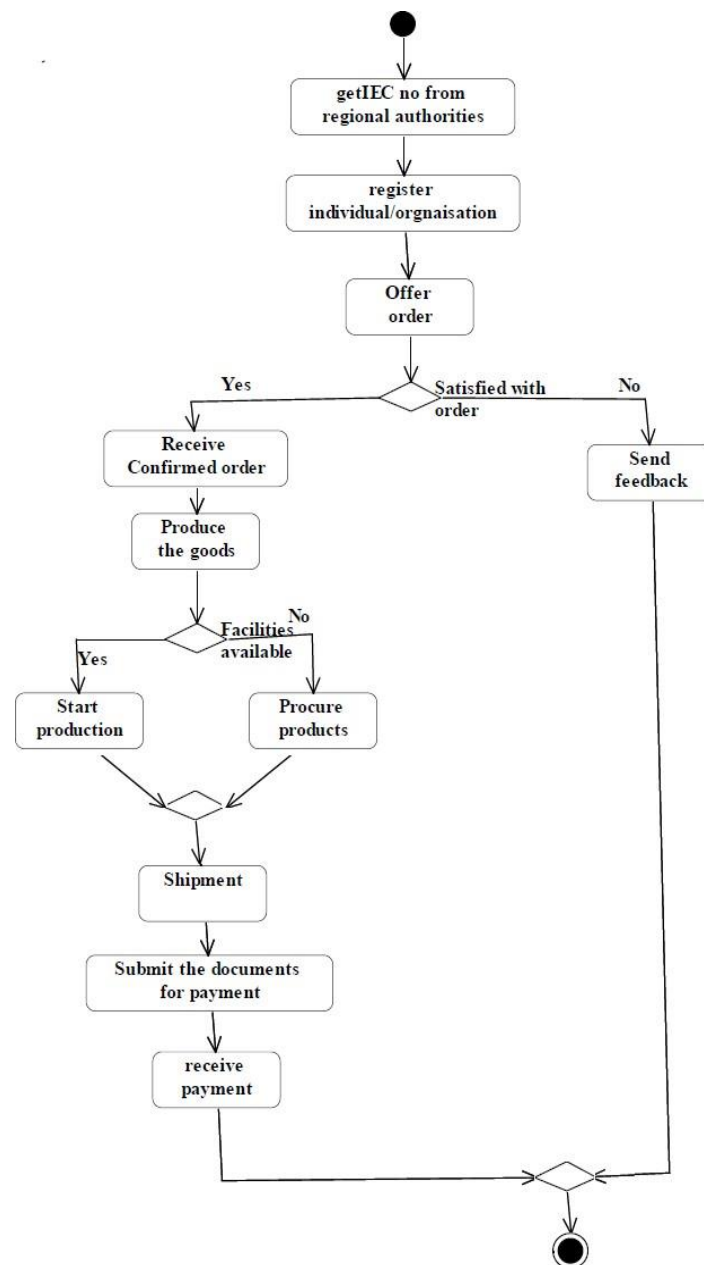
UML INTERACTION DIAGRAMS



STATE CHART DIAGRAM



ACTIVITY DIAGRAM



IMPLEMENTATION OF THE CODE

BuyerInfo.java

```
public class BuyerInfo  
{  
    private String BuyerId;  
    private String BuyerName;  
    private String BuyerAddress;  
    private Integer BcontactNo;  
    private String BEmailId;  
    public Production theProduction;  
  
    public BuyerInfo()  
    {  
    }  
    public void SendConfirmedOrder()  
    {  
    }  
}
```

RESULT:

Ex.No :14

Date:

LIBRARY MANAGEMENT SYSTEM

AIM:

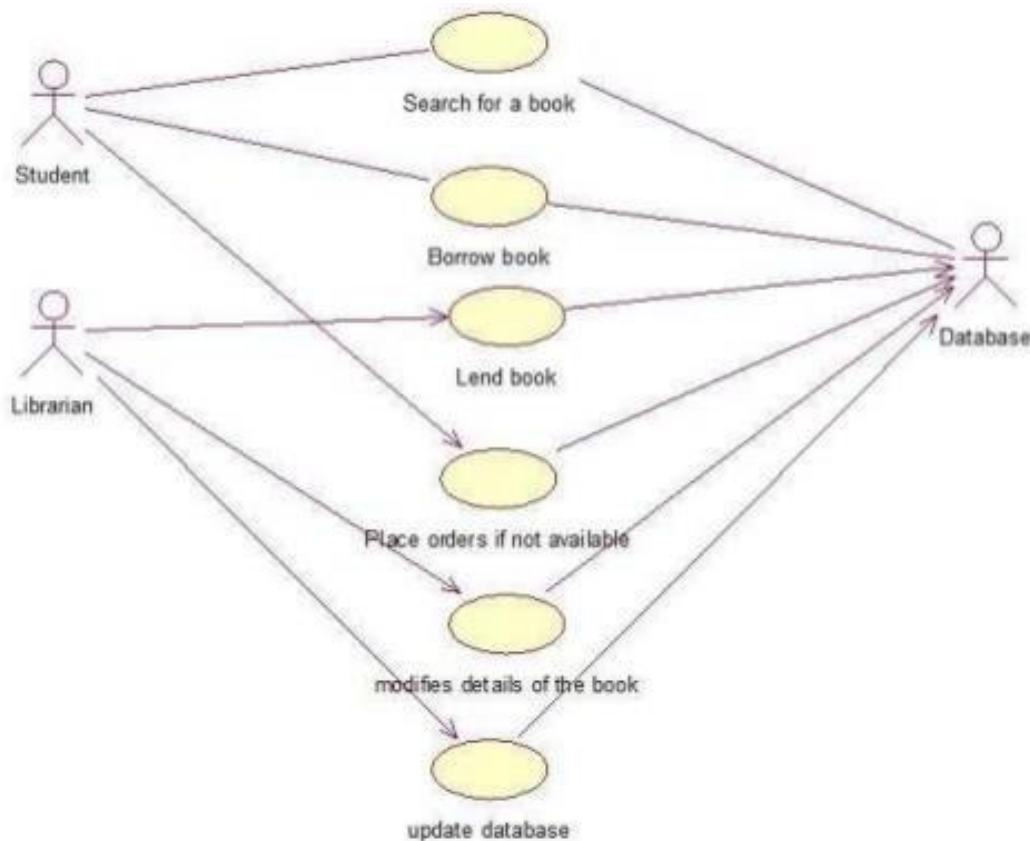
PROBLEM STATEMENT

The library lends books and magazines to borrowers, who are registered in the system, as are the books and magazines. The library handles the purchase of new titles for the library. Popular titles are bought in multiple copies. Old books and magazines are removed when they are out of date or in poor condition.

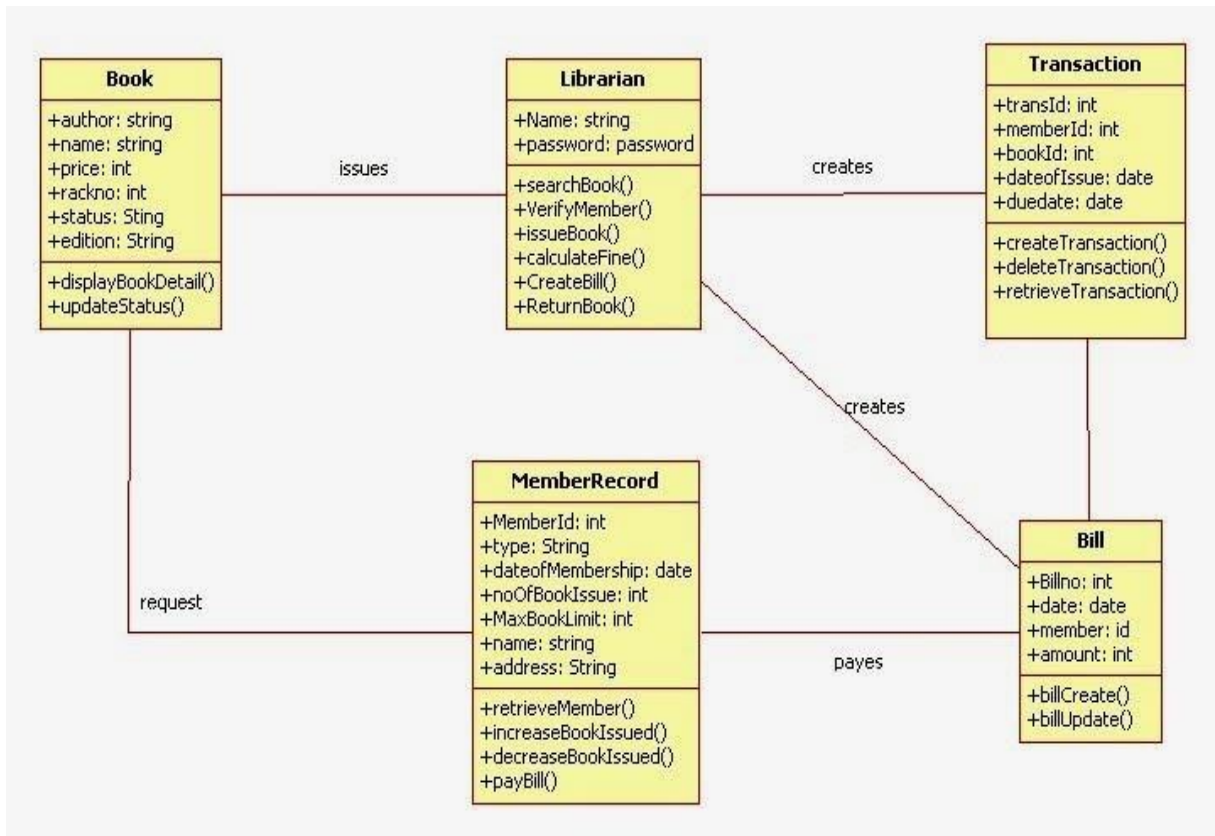
The librarian is an employee of the library who interacts with the customers (borrowers) and whose work is supported by the system. A borrower can reserve a book or magazine that is not currently available in the library, so that when it's returned or purchased by the library, that borrower is notified.

The reservation is cancelled when the borrower checks out the book or magazine or through an explicit canceling procedure. The librarian can easily create, update, and delete information about the titles, borrowers, loans, and reservations in the system.

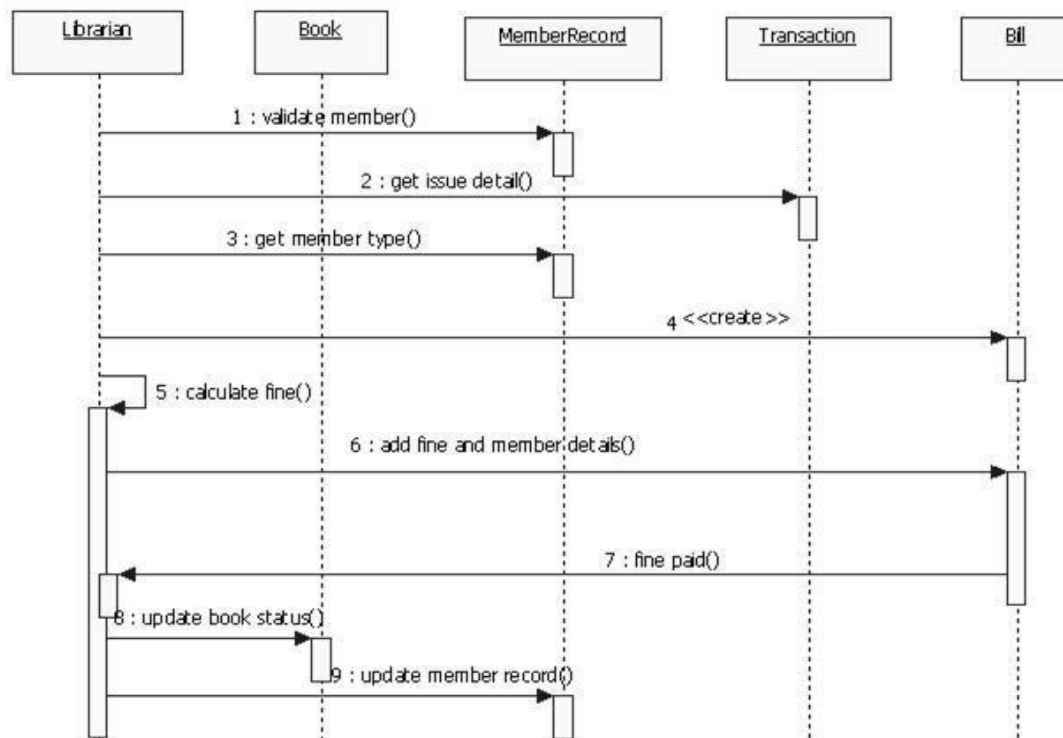
IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL



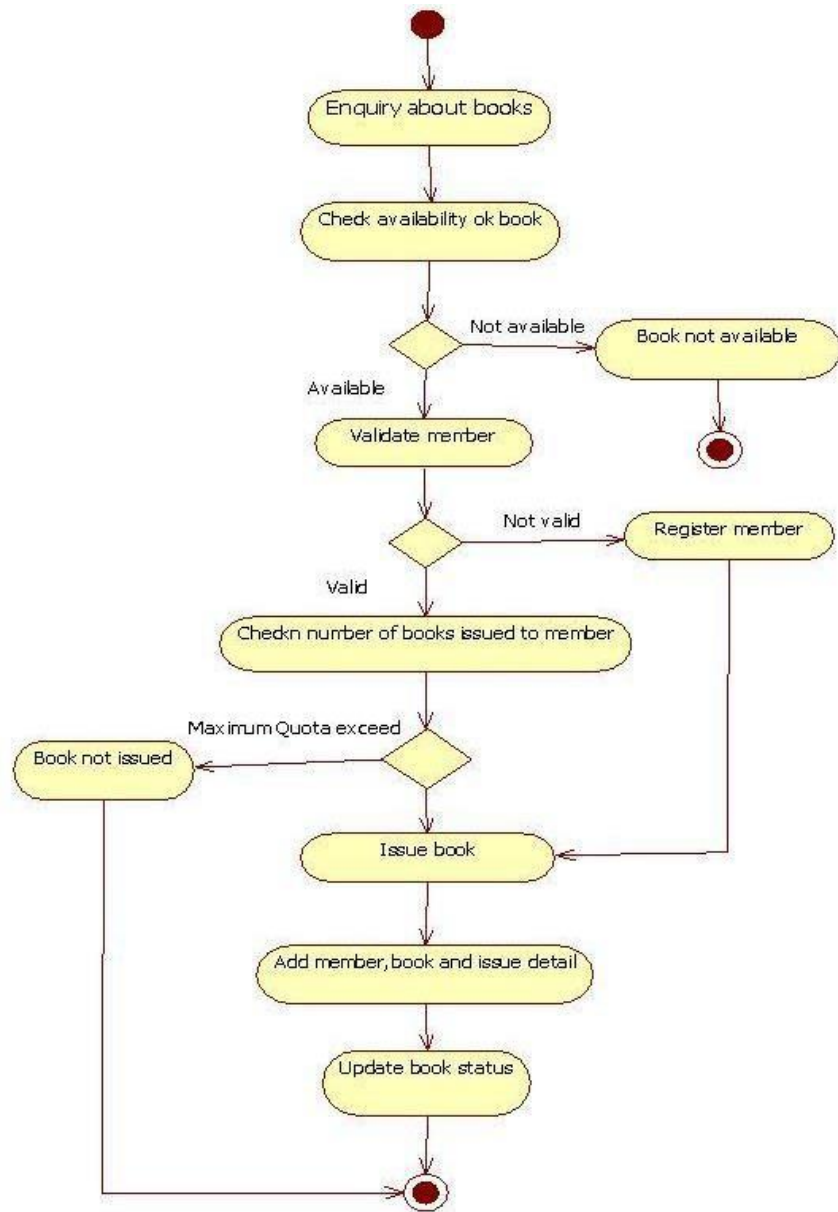
IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM



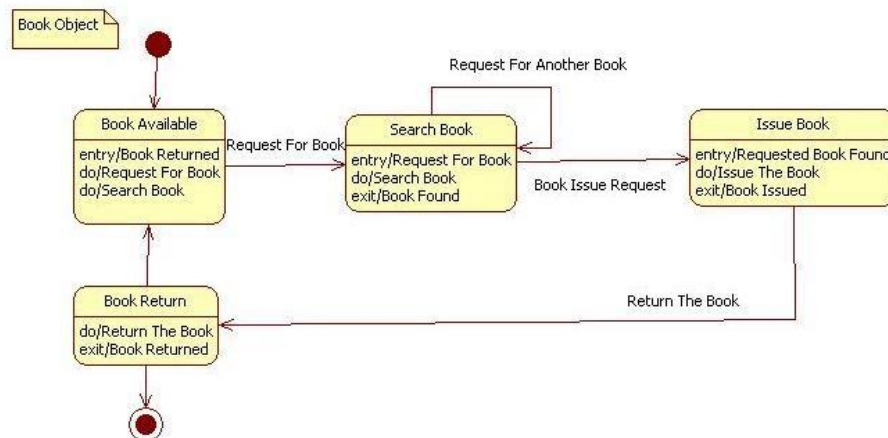
UML INTERACTION DIAGRAMS



ACTIVITY DIAGRAM



STATE CHART DIAGRAM



IMPLEMENTATION OF THE CODE

```
public class Book {  
  
    //  
  
    // Fields  
  
    //  
  
    public author name name;  
  
    public void code;  
  
    public void title;  
  
    //  
  
    // Constructors  
  
    //  
  
    public book () { };  
  
    public void setName ( author name newVar ) {  
  
        name = newVar;  
  
    }  
}
```

RESULT:

Ex.No :15

Date:

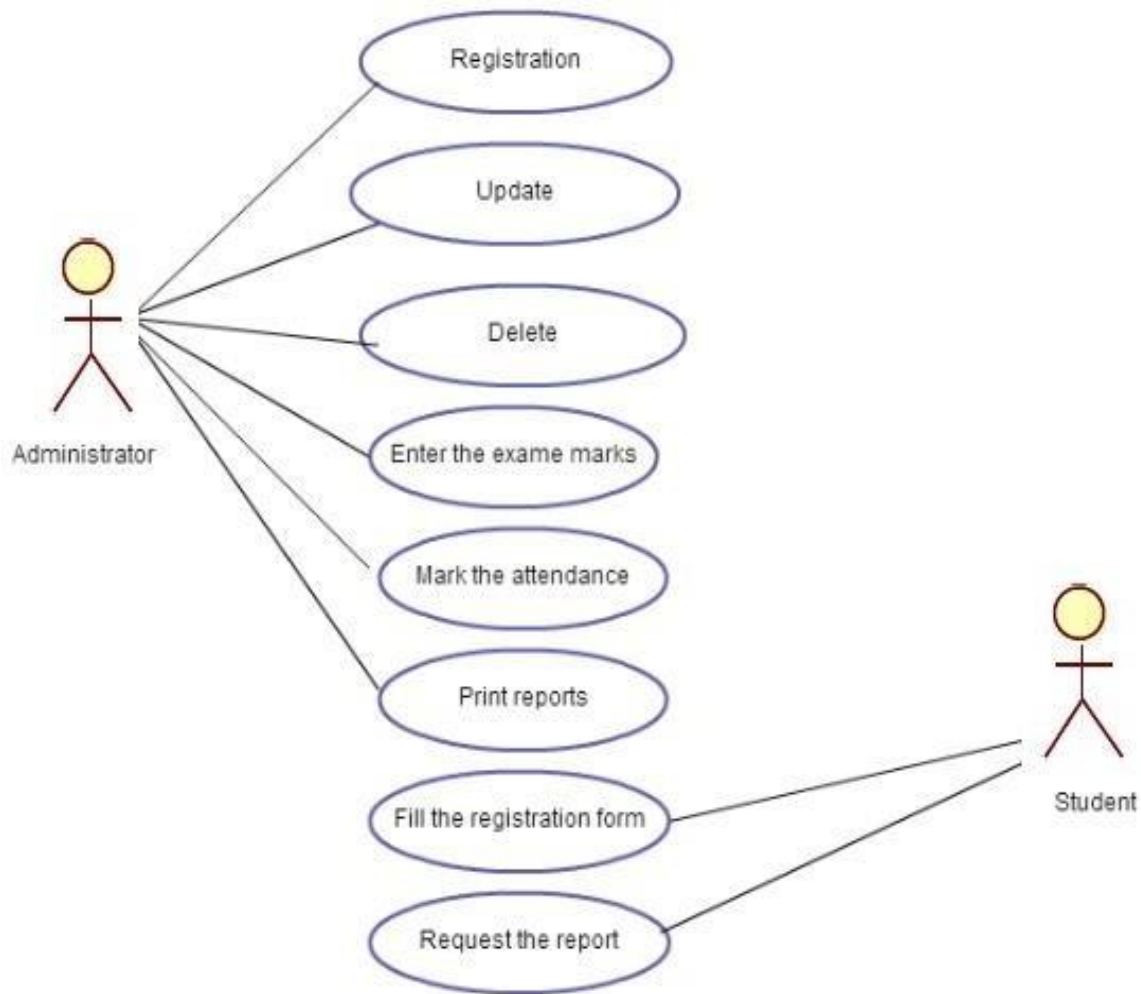
STUDENT INFORMATION SYSTEM

AIM:

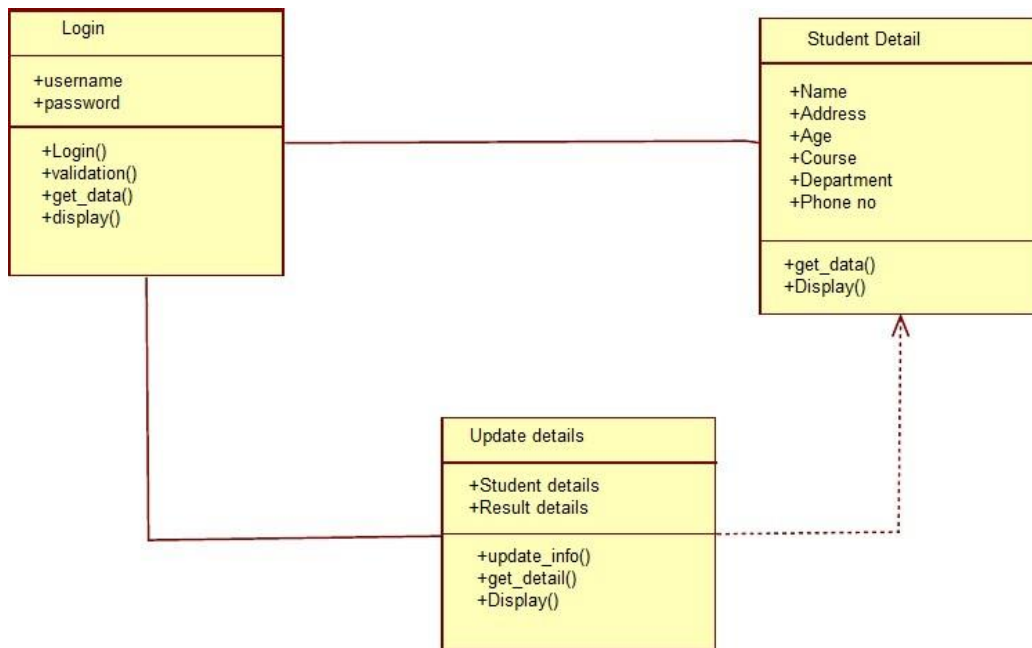
PROBLEM STATEMENT

A Student information system (SIS) is a software application for educational establishment to manage student data. Student information systems provide capabilities for entering student test and other assessment scores, building student schedules, tracing student attendance and managing many other student-related data needs in a school college or university.

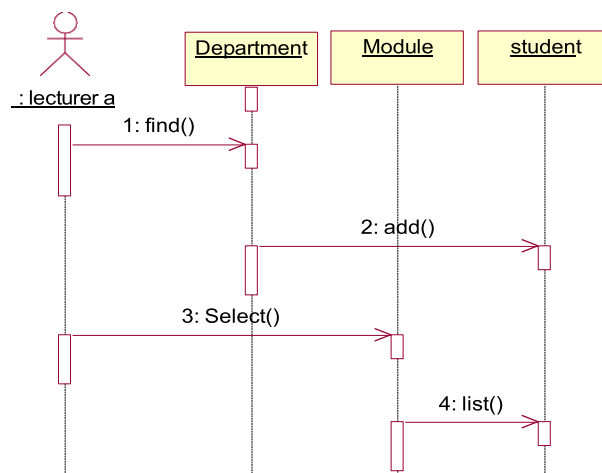
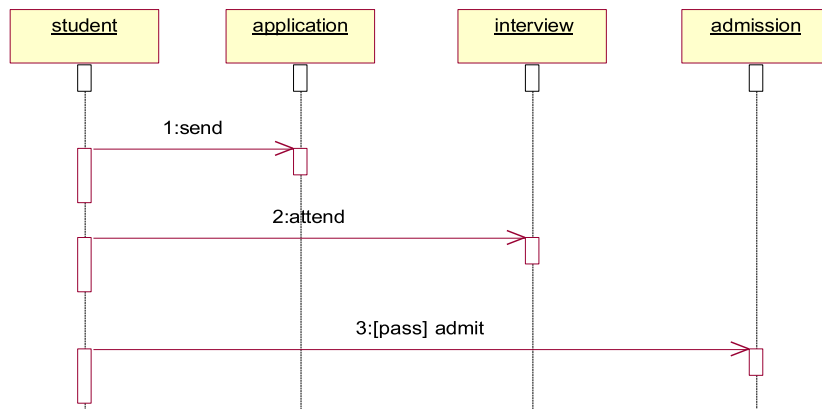
IDENTIFY THE USE-CASES AND DEVELOP THE USE-CASE MODEL



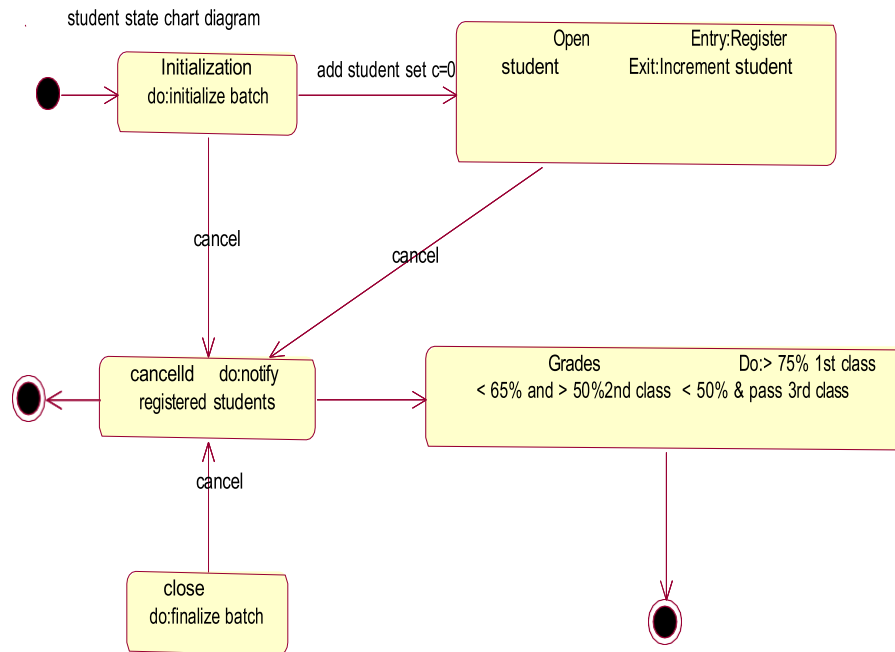
IDENTIFY THE CONCEPTUAL CLASSES AND DEVELOP A DOMAIN MODEL WITH UML CLASS DIAGRAM



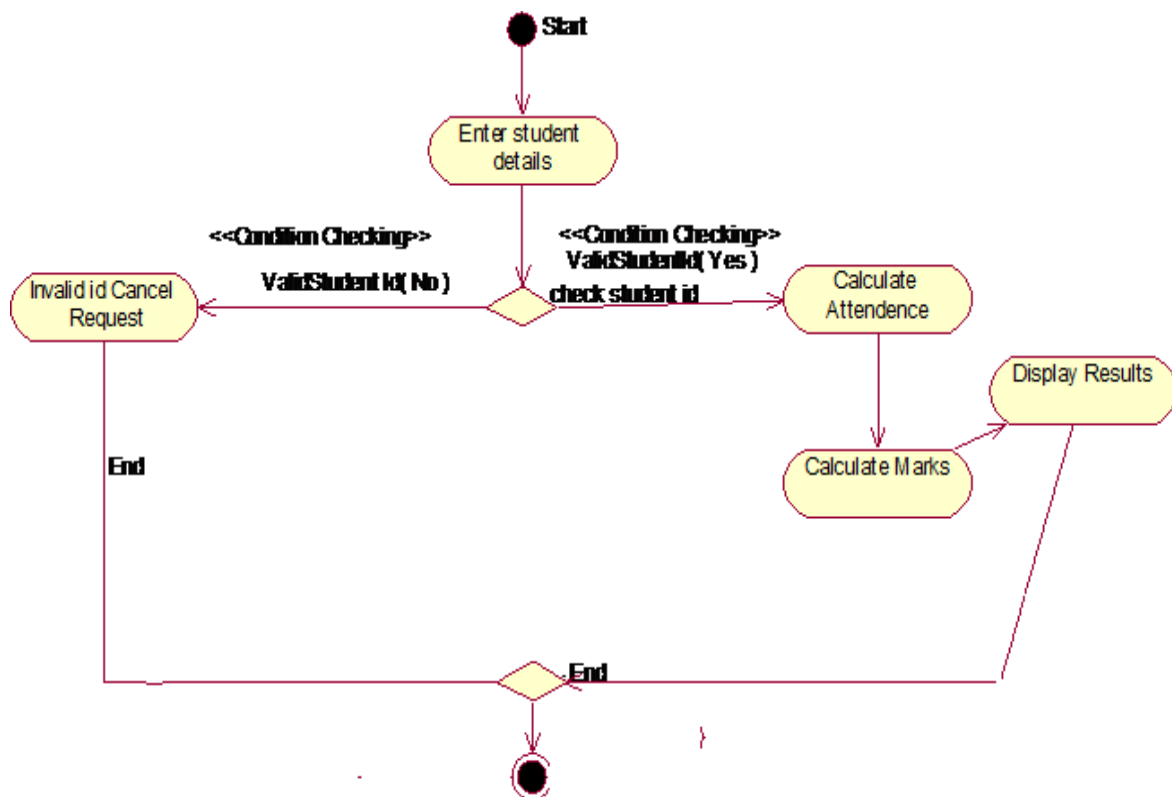
UML INTERACTION DIAGRAMS



STATE CHART DIAGRAM



ACTIVITY DIAGRAM



IMPLEMENTATION OF THE CODE

* Class student

*/

```
public class student {
```

```
    public void name;
```

```
    public void age;
```

```
    public student () { };
```

```
    public void setName ( void newVar ) {
```

```
        name = newVar;
```

```
    }
```

```
    public void getName ( ) {
```

```
        return name;
```

```
    }
```

RESULT: