

Lab Instructions

- 1. Please do enter the lab with proper dress code.**
- 2. Maintain silence in the lab.**
- 3. Keep the bags & other belongings outside the lab.**
- 4. Follow strict timings of the lab hours.**
- 5. Students do not enter the lab without the presence of concern staff in-charge.**
- 6. Students should have their observation during lab hours.**
- 7. Students should enter the lab according to the slot/batch assigned.**
- 8. Any Xerox materials / Books are not allowed.**
- 9. Please do proper Switch ON/OFF of the systems.**
- 10. Do not use/open any other applications or games other than the lab requirements.**
- 11. Usage of any removable media/disk is prohibited.**
- 12. Proper entry should be made in the lab register along with timings.**
- 13. Any system issues please report to your staff incharge.**
- 14. Students are responsible for any damage of the system they use.**
- 15. Arrange the chairs while leaving the lab.**

DEPARTMENT OF INFORMATION TECHNOLOGY

INSTITUTE VISION

To be a pioneer in engineering and management education by providing amicable ambience and being a centre for learning thereby perpetuating versatile professionals in engineering and management suitable for industry, higher education and research.

INSTITUTE MISSION

- Developing, nurturing and building professionals with technical, research and managerial expertise, intellectual proficiency, right attitude and moral ethics.
- Building infrastructure that earmarks the institution to be the most preferred educational destination and provides the right ambience for creative and innovative teaching - learning process.
- Enabling learners to broaden their horizon and helping to meet global challenges.

VISION OF THE DEPARTMENT

To provide quality education to the students and impart IT excellence by building strong academic environment and to enable the essential skills in the innovators and entrepreneurs making them as proficient professionals for industrial consultancy.

MISSION OF THE DEPARTMENT

- Developing innovative, competent computer engineers through planning, in-depth analysis, and hands-on problem solving in the quest for the students to work with emerging technologies.
- Inculcating a learning environment and upgrade the knowledge of students through value based education and promotes the academic excellence.
- Providing a leadership and ethical values for effective deliberate and tactical planning to cater the needs of the Society and to Prepare students for professional career and higher studies and promoting innovative research and education programs in the IT field.

PROGRAMME OUTCOMES (POs)

PO1 - Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems

PO2 - Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3 - Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations

PO4 - Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5 - Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6 - The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7 - Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 - Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9 - Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 - Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 - Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12 - Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAMME SPECIFIC OUTCOMES (PSOs)

PSO1: Have proficiency in programming skills to design, develop and apply appropriate techniques, to solve complex engineering problems.

PSO2: Have knowledge to build, automate and manage business solutions using cutting edge technologies.

PSO3: Have excitement towards research in applied computer technologies.

List of Experiments

LIST OF EXPERIMENTS:

- 1. Identify a software system that needs to be developed.**
- 2. Document the Software Requirements Specification (SRS) for the identified system.**
- 3. Identify use cases and develop the Use Case model.**
- 4. Identify the conceptual classes and develop a Domain Model and also derive a Class Diagram from that.**
- 5. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence and Collaboration Diagrams**
- 6. Draw relevant State Chart and Activity Diagrams for the same system.**
- 7. Implement the system as per the detailed design**
- 8. Test the software system for all the scenarios identified as per the usecase diagram**
- 9. Improve the reusability and maintainability of the software system by applying appropriate design patterns.**
- 10. Implement the modified system and test it for various scenarios.**

SUGGESTED DOMAINS FOR MINI-PROJECT:

- 1. Passport automation system.**
- 2. Book bank**
- 3. Exam registration**
- 4. Stock maintenance system.**
- 5. Online course reservation system**
- 6. Airline/Railway reservation system**
- 7. Software personnel management system**
- 8. Credit card processing**
- 9. e-book management system**
- 10. Recruitment system**
- 11. Foreign trading system**
- 12. Conference management system**
- 13. BPO management system**
- 14. Library management system**
- 15. Student information system**

TABLE OF CONTENTS

S.NO.	EXPERIMENT TITLE
1.	Passport automation system
2.	Book bank
3.	Exam Registration
4.	Stock maintenance system
5.	Online course reservation system
6.	Airline/Railway reservation system
7.	Software personnel management system
8.	Credit card processing
9.	E-book management system
10.	Recruitment system
11.	Foreign trading system
12.	Conference Management System
13.	BPO Management System
14.	Library management system
15.	Student information system
CONTENT BEYOND SYLLABUS	
1	ATM SYSTEM

EX. NO:1

PASSPORT AUTOMATION SYSTEM

AIM:

To design a Passport Automation System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN:-

To simplify the process of applying passport, software has been created by designing through ARGO-UM tool. Initially the applicant login the passport automation system and submits his details. These details are stored in the database and verification process done by the passport administrator, regional administrator and police the passport is issued to the applicant.

➤ **PROBLEM STATEMENT:-**

1. 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.

2. 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.

3. 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 (Ministry of External Affairs) office.

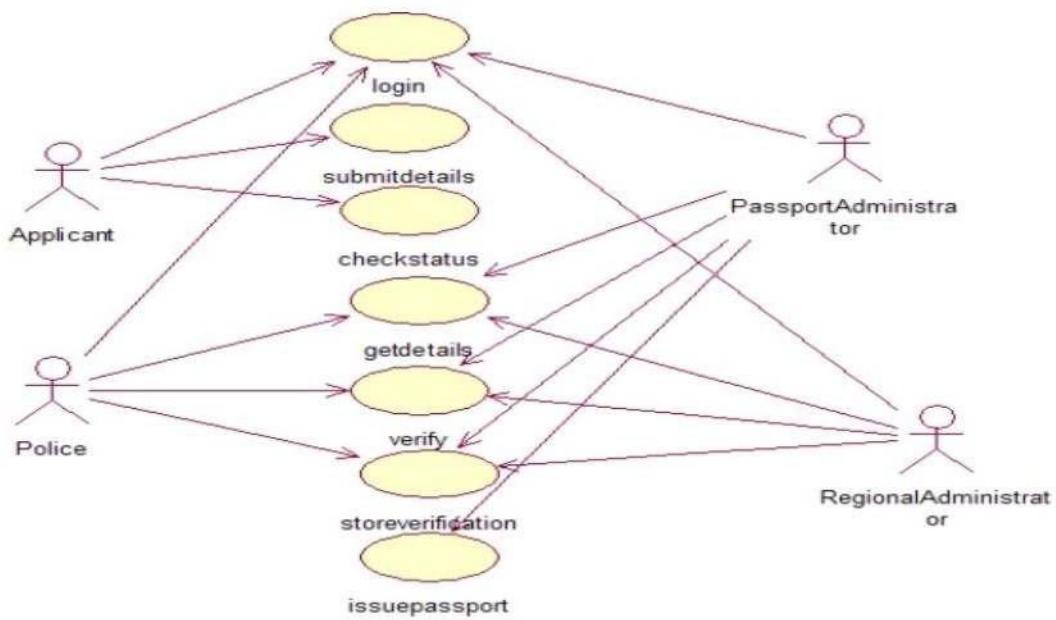
4. The application is then processed manually based on the report given by the system, and any forfeiting identified can make the applicant liable to penalty as per the law.

5. The system forwards the necessary details to the police for its separate verification whose report is then presented to the administrator. After all the necessary criteria have been met, the original information is added to the database and the passport is sent to the applicant.

UML USE CASE DIAGRAM:-

Description:

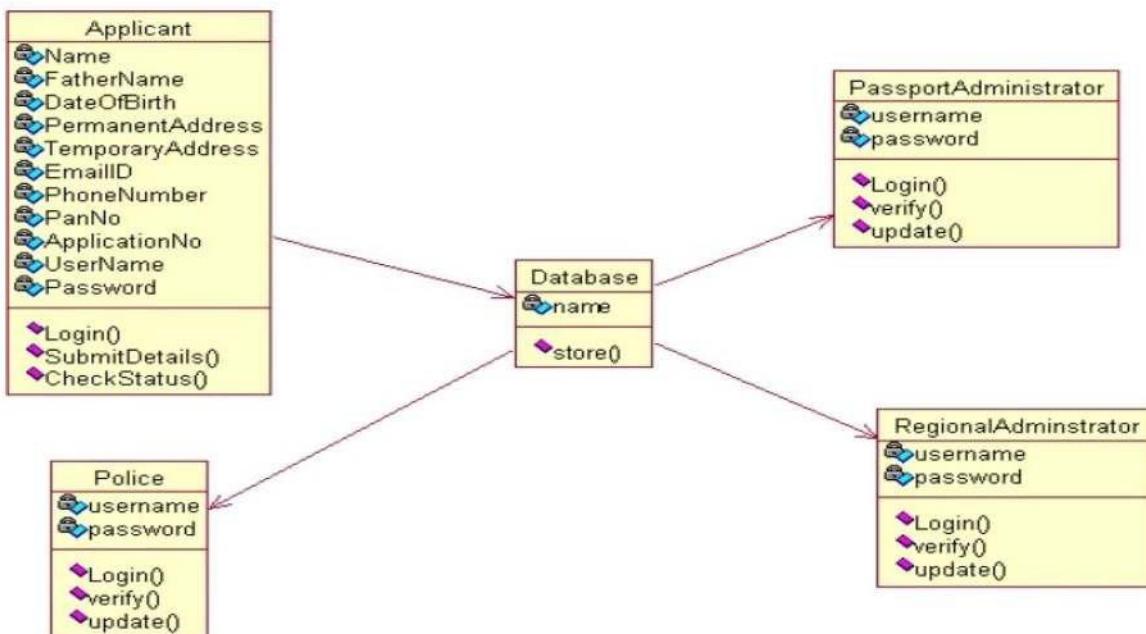
A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM

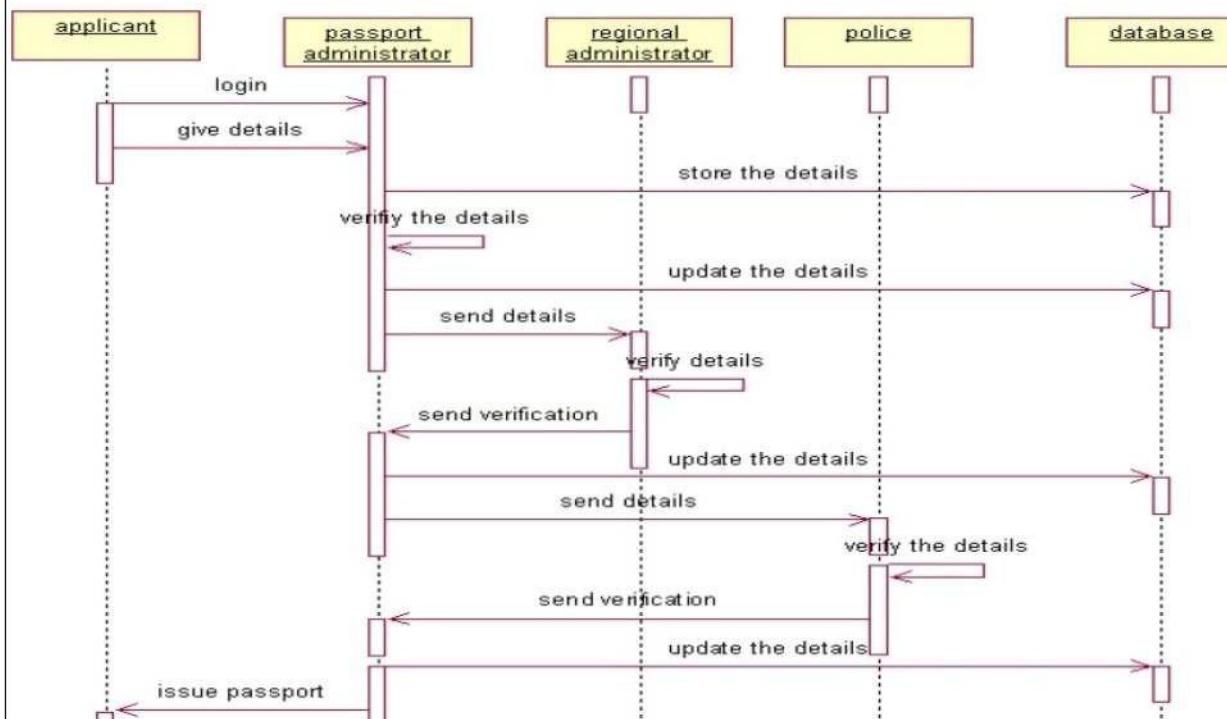
APPLICANT
THE DATABASE
REGIONAL ADMINISTRATOR
PASSPORT ADMINISTRATOR
THE POLICE

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM

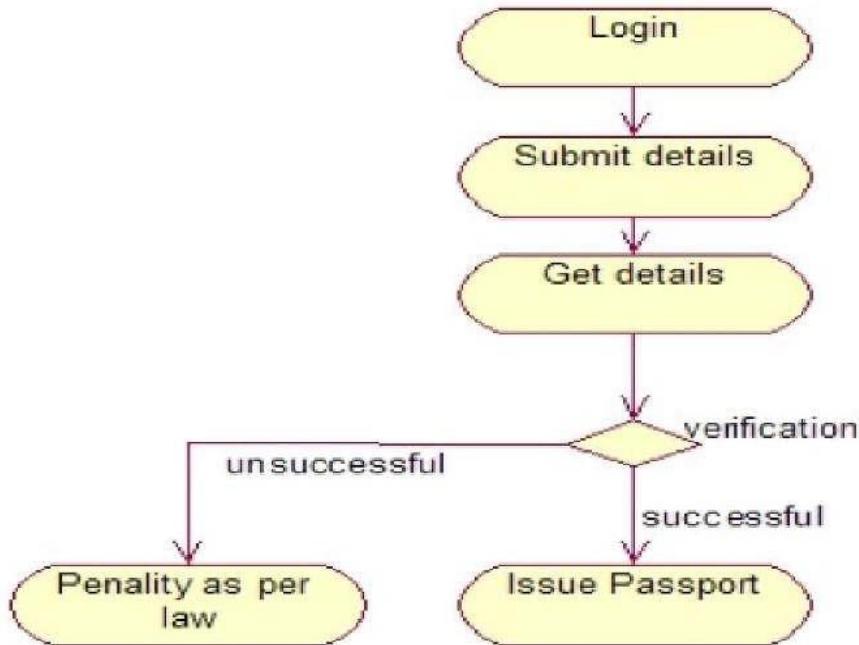
- APPLICATION
- PASSPORT ADMINISTRATION
- REGIONAL ADMINISTRATION

- POLICE
- DATABASE

UML ACTIVITY DIAGRAM:-

Description:-

Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency.



QUESTION

Activity diagram Describes:-

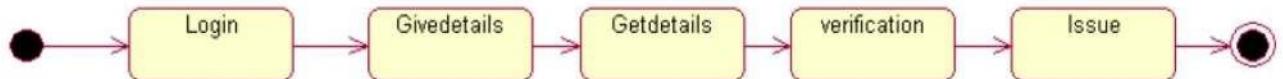
- How activities are coordinator to provide a service.
- The events needed to achieve some operation.
- How events in a single use case relate to one another.

DOCUMENTATION OF ACTIVITY DIAGRAM

- The activities in the passport automation system are login, submit details, get details, issue passport and penalty and verification.
- In the login activity applicant give username and password and then login into the passport automation system after then fill the details that are required for application.

UML STATECHART DIAGRAM:-

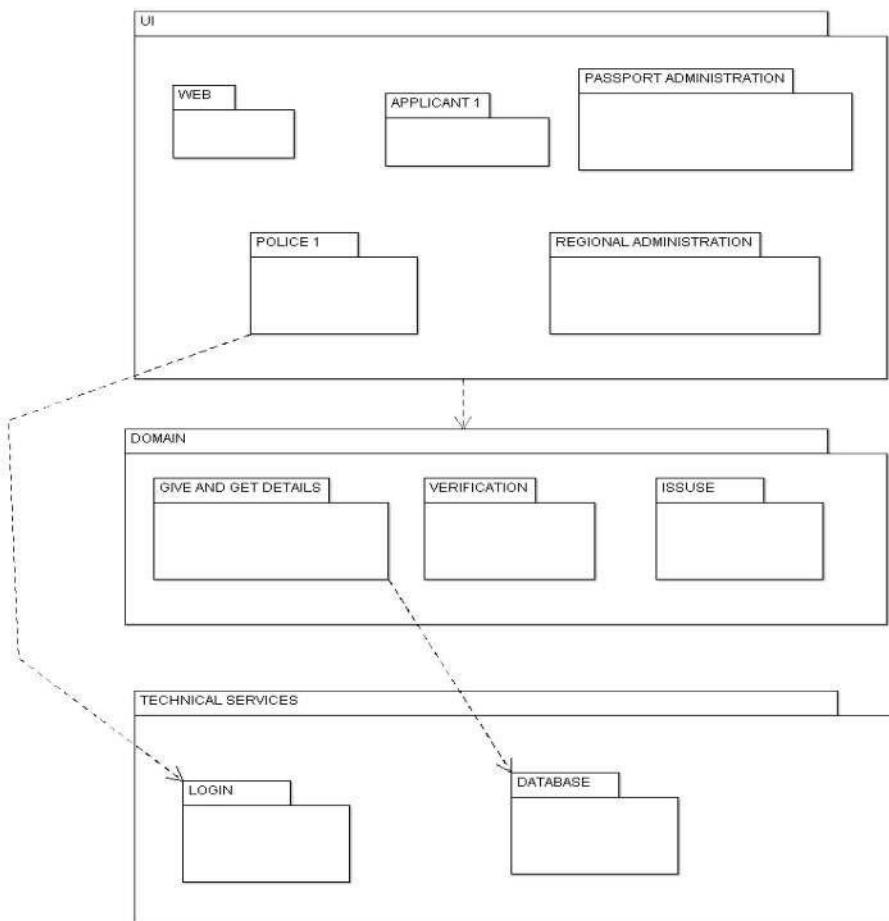
A state is represented as round box which may contains one or more compartments in it.



PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers.

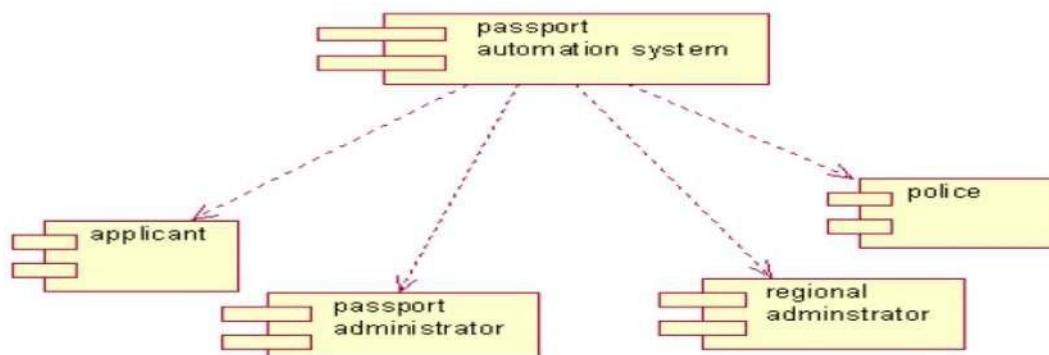


DOCUMENTATION OF THE PACKAGE DIAGRAM:-

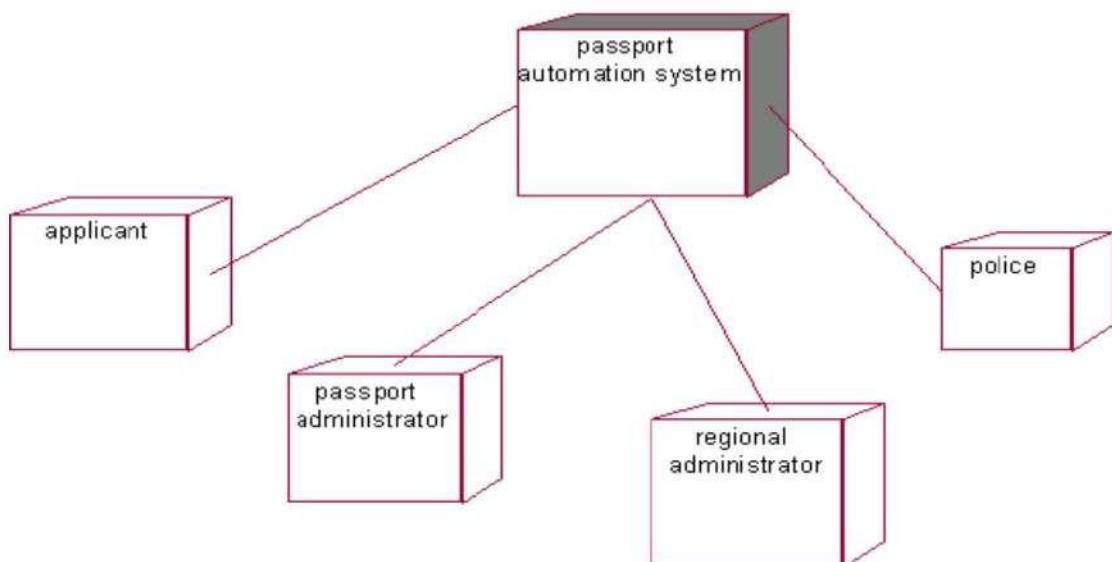
- **User-interface**
Web, applicant, police, regional administration.
- **Domain**
Give and get details, Verification, Issues.
- **Technical Services**
Login, Database.

UM 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 system. It is represented by boxed figure. Dependencies are represented by communication association.



UM 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.



UM TECHNICAL SERVICE LAYER:-

ID	NAME	AGE	ADDRESS	PHONE NUMBER
1	Rajahs	19	Lovelorn	2343423
2	Tejesh	25	Chennai	45645645
3	Suren	42	Madurai	24254466

ID	NAME
1	Rajesh
2	Tejesh
3	Suren

ID	Appoint men t	Applicant ID	Date	Time
1	123	345	12-09-2016	4
2	124	234	13-09-2016	5
3	145	445	15-09-2016	7

SAMPLE CODE:-**APPLICATION**

```

import java.util.Vector;
public class application {
public char name;
private char father name;
public int Afterbirth;
private varchar permanent address;
private varchar Temporary_address;
public varchar email;
public int Phone number;
public varchar pan No;
public varchar Application No;
public varchar Username;
public varchar password;
public Vector my Database;
public void login() {
}
public void submit details() {
}
public void checking status() {
}
}

```

DATABASE

```

import java.util.Vector;
public class Database {
public char name;
public Vector my application;
public Vector mypassportAdministration;
public Vector myregionalAdministrator;
public Vector mypolice;
public void store() {
}
}

```

USER INTERFACE LAYER:-

The screenshot displays two windows of the Online Passport Automation System:

GIVE YOUR DETAILS

This window contains fields for personal information:

- NAME: [Input Field]
- PASSWORD: [Input Field]
- DATE OF BIRTH: [Input Field] (7/2/1990)
- RESIDENCE ADDRESS: [Input Field] (Residence)
- EMPLOYMENT ADDRESS: [Input Field] (Residence)
- PHONE NO.: [Input Field] (9876543210)
- E-MAIL ID: [Input Field] (pankaj@abc.com)
- TC NO.: [Input Field] (000)

STATUS OF THE PASSPORT

This window shows the status of the passport application:

- APPLICATION NO: [Input Field]
- NAME: [Input Field] (Pankaj)
- PASSPORT ADMINISTRATOR: [Input Field] (Success)
- REGIONAL ADMINISTRATOR: [Input Field] (Success)
- POC: [Input Field] (Success)

YOUR APPLICATION HAS REGISTERED SUCCESSFULLY

This window displays a success message and the application number:

- YOUR APPLICATION NUMBER: [Input Field] (1234567890)

RESULT:

Thus the Passport Automation System has been done successfully by Argo- UML tool.

EX.NO:2

BOOK BANK MANAGEMENT SYSTEM

AIM:

To design a Book Bank System by using Argo-UML tool.

PROBLEM STATEMENT:-

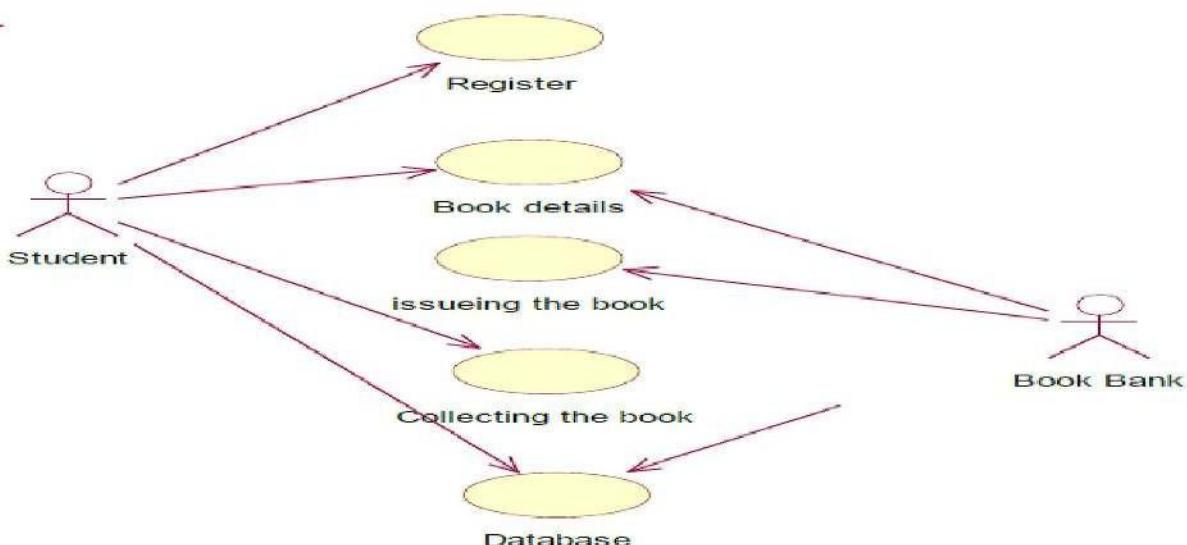
To simplify the process of applying Book Bank, software has been created by designing through ARGONAUT tool. The process of members registering and purchasing books from the book bank are described sequentially through following steps:

- a. First the member registers himself if he was new to the book bank.
- b. Old members will directly select old member button..
- c. They select their corresponding year.
- d. After selecting the year they fill the necessary details and select the book and he will be directed towards administrator
- e. The administrator will verify the status and issue the book.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

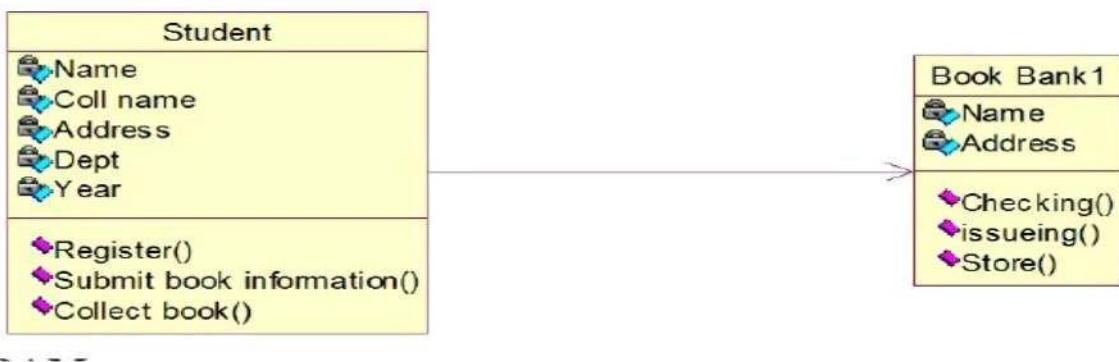
Actor: - students, Book bank admin, user

Use case: - Students details, Date of issue, Date of return, no of books taken, Check availability.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

STUDENTS

Registration, Request for book, Return previous Books

COMPUTER

Student's record, issue, return, check availability, order books, verify id, renewal by online, username and password.

ADMIN:

Verify students, check for book availability, Issue books, Order for new author, maintain students details

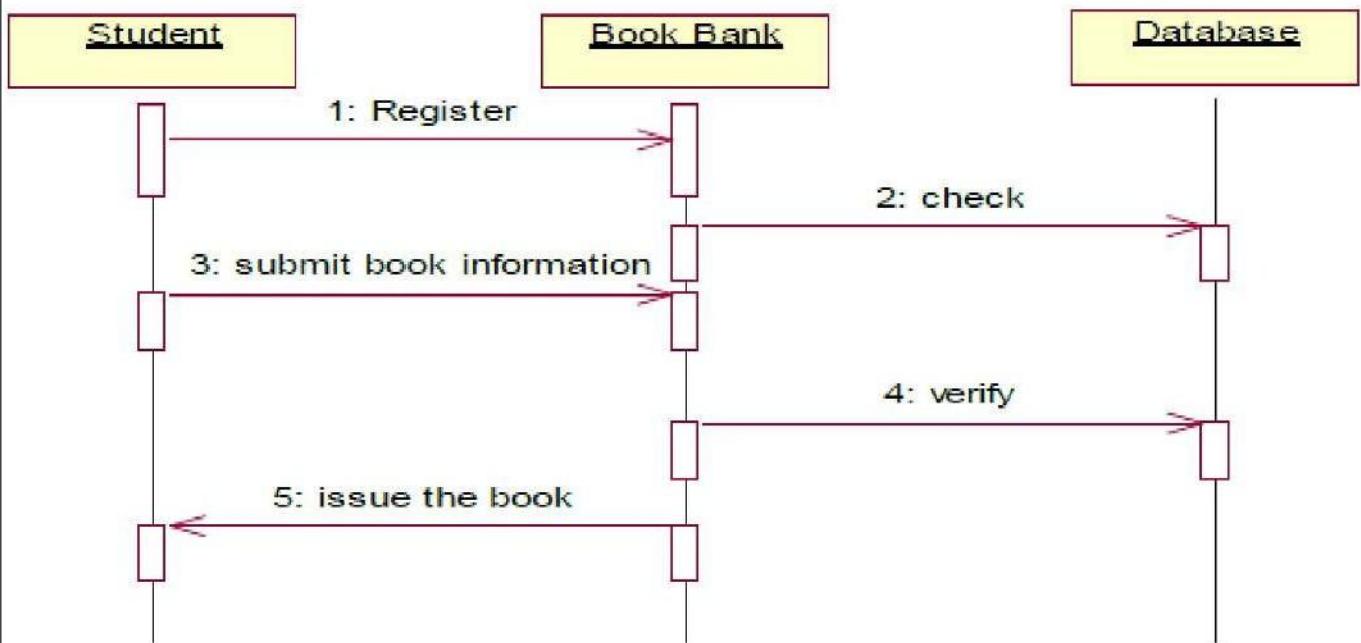
UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

1. User
2. Admin
3. Computer

UML STATE CHART DIAGRAM:-

Description:-

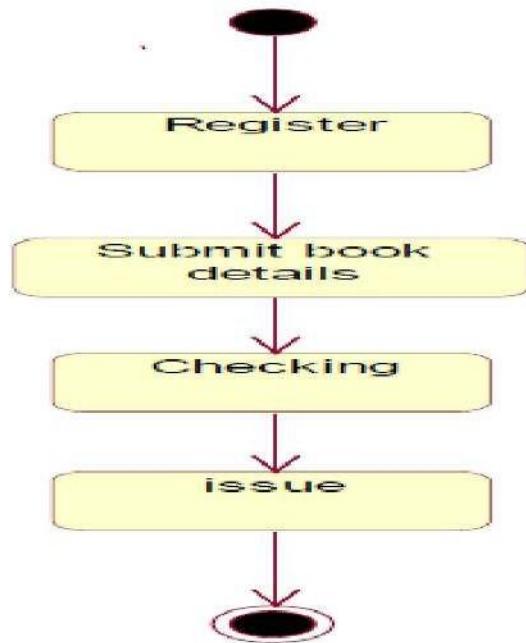
Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

How activities are coordinator to provide a service.

The events needed to achieve some operation.

How events in a single use case relate to one another.

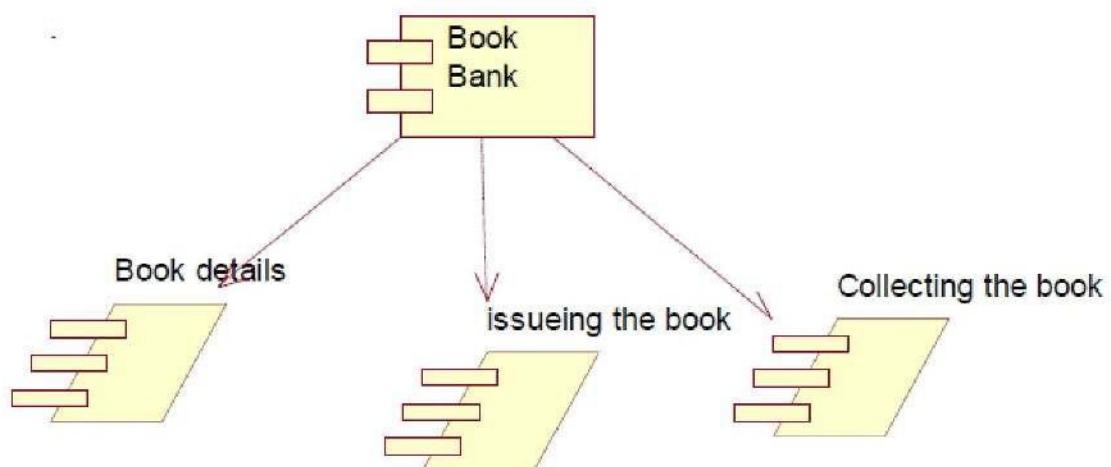


DOCUMENTATION OF STATE CHART DIAGRAM:-

- Registration.
- Year of Selection.
- Check the availability.
- Select the books
- Inundation
- Issue books
- Verification
- Authentication

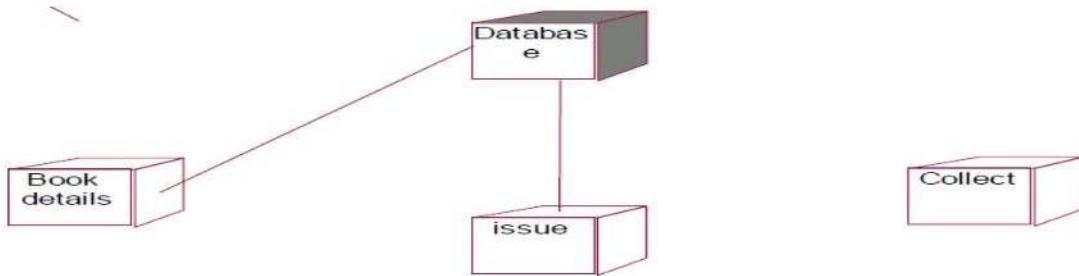
UML COMPONENT DIAGRAM:

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



UML DEPLOYMENT DIAGRAM:

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



UML TECHNICAL SERVICE LAYER:-

S . N o	Std-Id	Book No	Date-of-issue
1.	15680	1234	01-06-2017
2.	15692	4102	15-07-2017
3.	15682	2011	20-06-2017

S. No	Book- No	Std-Id	Date of return
1.	1234	15680	15-06-2017
2.	4102	15692	30-07-2017
3.	2011	15682	05-07-2017

SAMPLE CODE:-

```

import java.util.Vector;
public class computer {
    public int std_id;
    public varchar book_list;
    public Vector my student;
    public Vector my student;
    public Vector my Admin;
    public Vector my Admin;
    private void students records() {
    }
    private void issue() {
    }
    public void return() {
    }
    public void check_availability() {
    }
    public void order_books() {
  
```

```

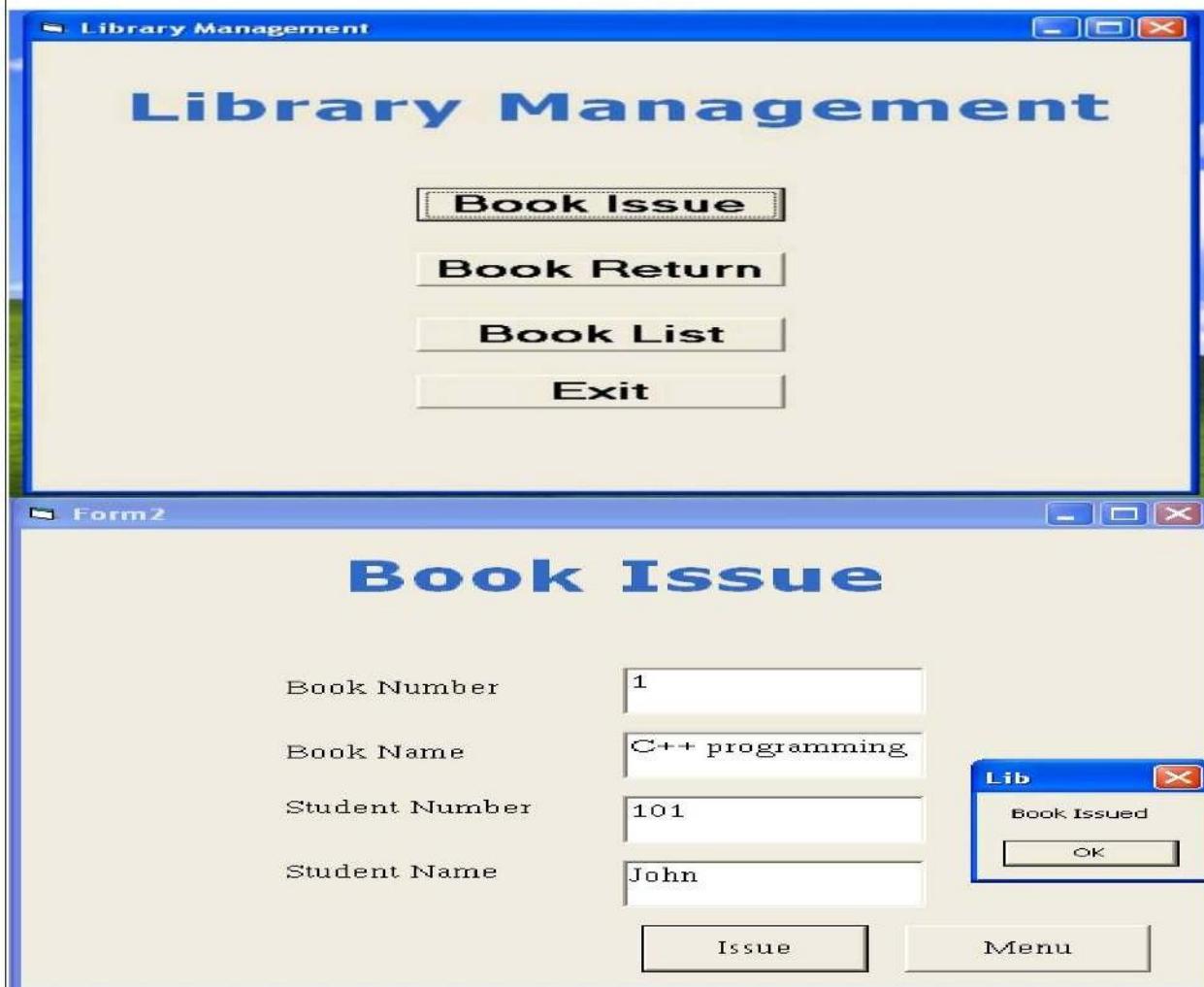
import java.util.Vector;
public class student {
    public varchar stud_id;
    public varchar date_of_issue;
    public archival date_of_return;
    public Vector my computer;
    public void request_for_books() {
    }
    public void return_previous_book() {
  
```

```

}
public void verify_if() {
}
public void check_the_availability() {
}
public void enter_the_book_issue() {
}
public void renewal_by_online() {
}
public void username&password() {
}
}

```

USER INTERFACE LAYER:-



Result:-

Thus the Book Bank Management has been done successfully by using Argo-UML.

EX.NO: 3	EXAM REGISTRATION
----------	-------------------

AIM:

To design Exam Registration System by using Argo-UML tool. .

PROBLEM ANALYSIS AND PROJECT PLAN

To simplify the process of applying Exam Registration, software has been created by designing through ARGO-UM tool.

The exam registration is an application in which applicant can register themselves for the exam. The details of the students who have registered for the examination will be stored in a database and will be maintained. The registered details can then be verified for any fraudulent or duplication and can be removed if found so. The database which is verified can be used to issue hall tickets and other necessary materials to the eligible students.

PROBLEM STATEMENT

Exam registration system is used in the effective dispatch of registration from to all of the students this system adopts a comprehensive approach to minimize the manual work and schedule resources, time in cogent manner the core of the system is to get the online registration from (with details such as name , oregano , etc.,) filled by the student whose statement is verified for it is genuineness by the exam registration system with respect to the already existing information in the database. This forms the first and fore most step in the processing of exam application.

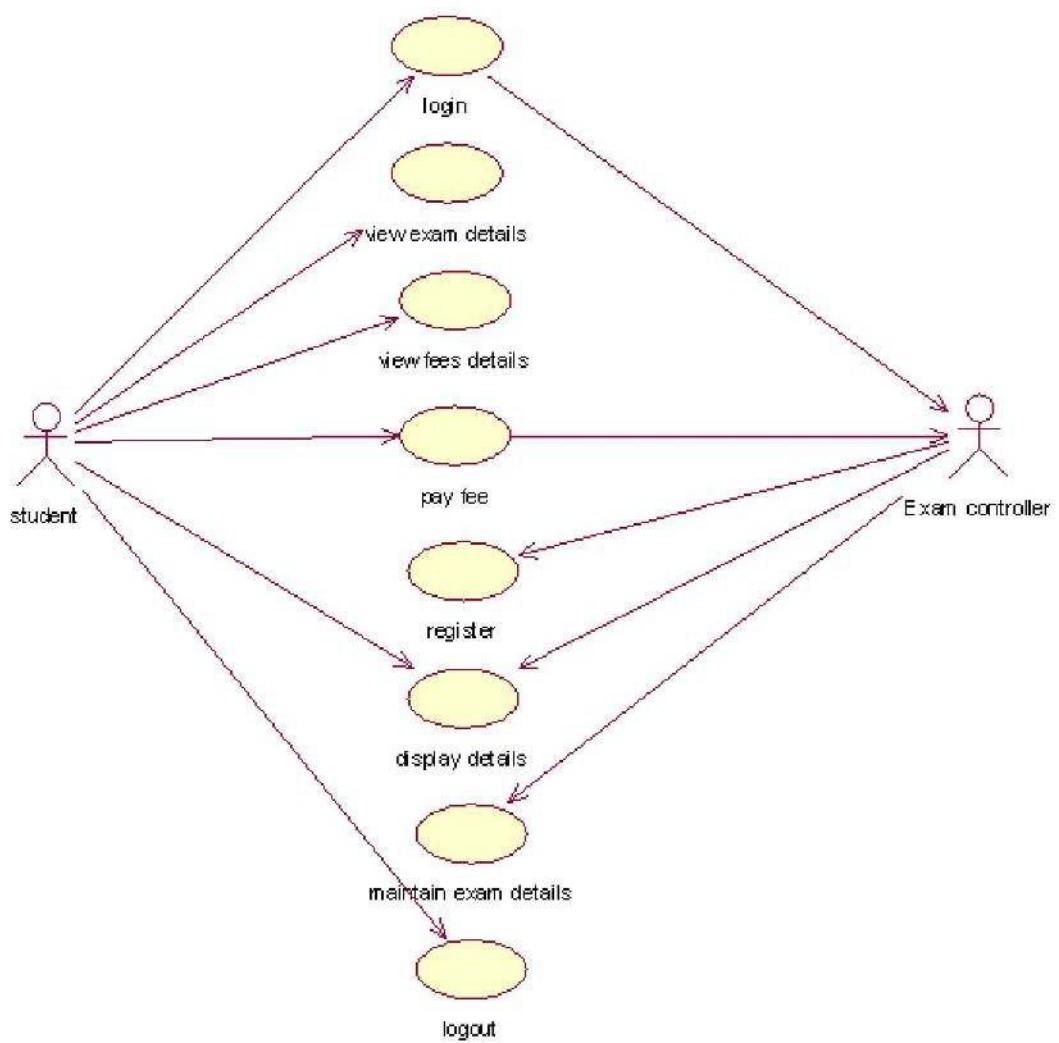
The process of students accessing the registration application and applying for the examination by filling out the form with proper details and then the authorities verify those details given for truth and correctness are sequenced through steps

- a. The students access exam registration application.
- b. They fill out the form with correct and eligible details.
- c. They complete the payment process.
- d. The authorities verify or check the details.
- e. After all verification the exam registration database is finalized.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

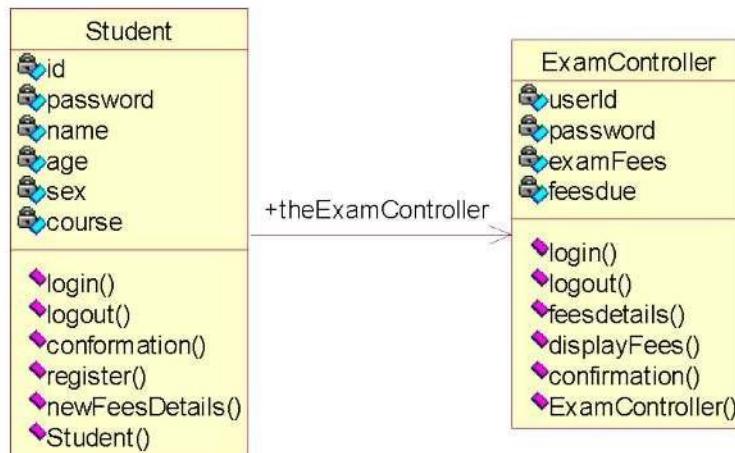
The actors in this use case diagram are Students, Interface and Database. The use cases are the activities performed by actors.

- a. Student fills outs the form in the form filling process.
- b. The interface checks and validates registered details.
- c. Then the database is searched for details and verified.
- d. Database stores the details and returns acknowledgment.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

- Candidate
- Fee details
- Fee proof
- Admin class
- Application Generation

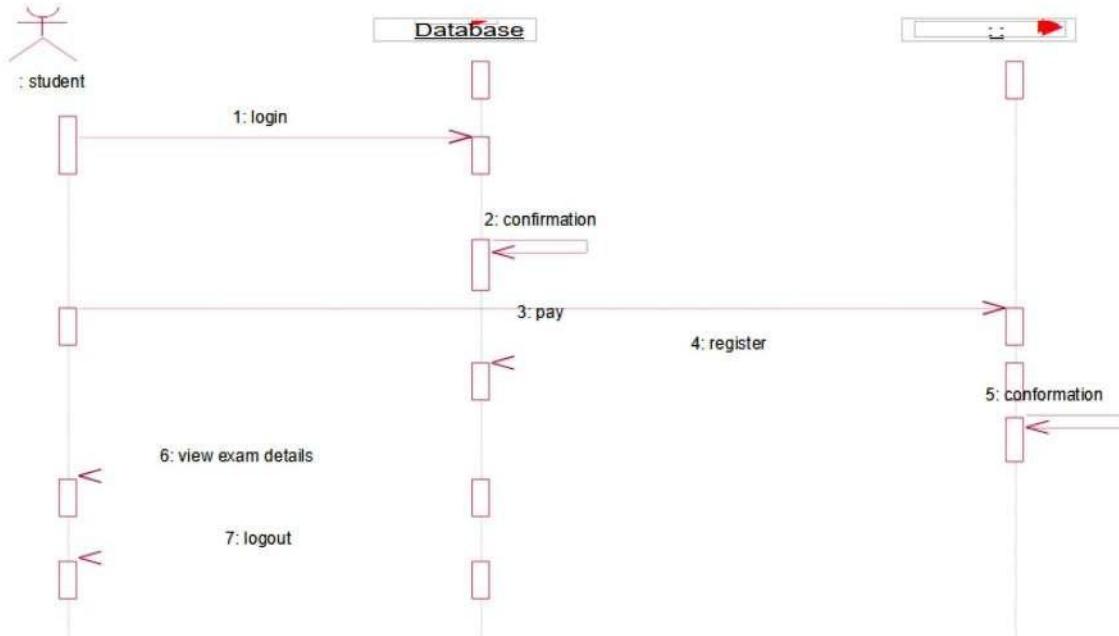
UML INTERACTION DIAGRAM:-

Description:-

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.

It's basically used for the purpose of security or safety. It contains user, database for the authentication status is verified from the database. The candidate will fill out the form with correct and eligible details.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

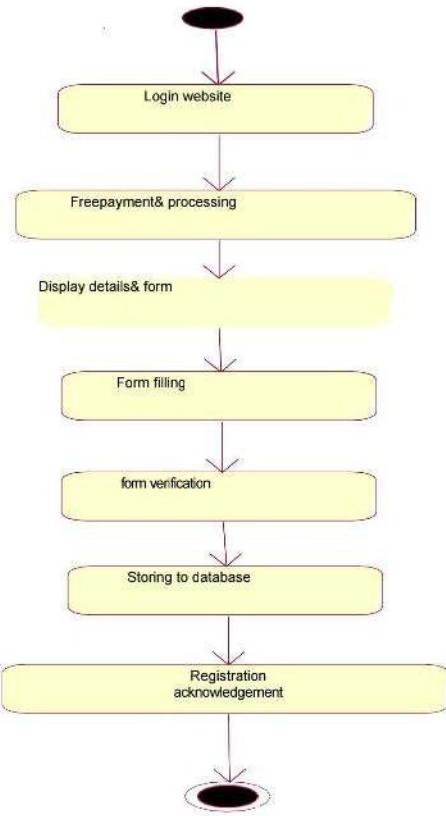
The sequence diagram describes the sequence of steps to show

- The candidate selects the exam and enters their detail in the online registration form.
- Then the candidate selects the payment mode like Credit card, Debit card and Net banking.
- If the payment is not completed then it goes for repayment and after completion of the payment the application will be generated.
- The detail of exam is viewed by the candidate.

UML ACTIVITY DIAGRAM:-

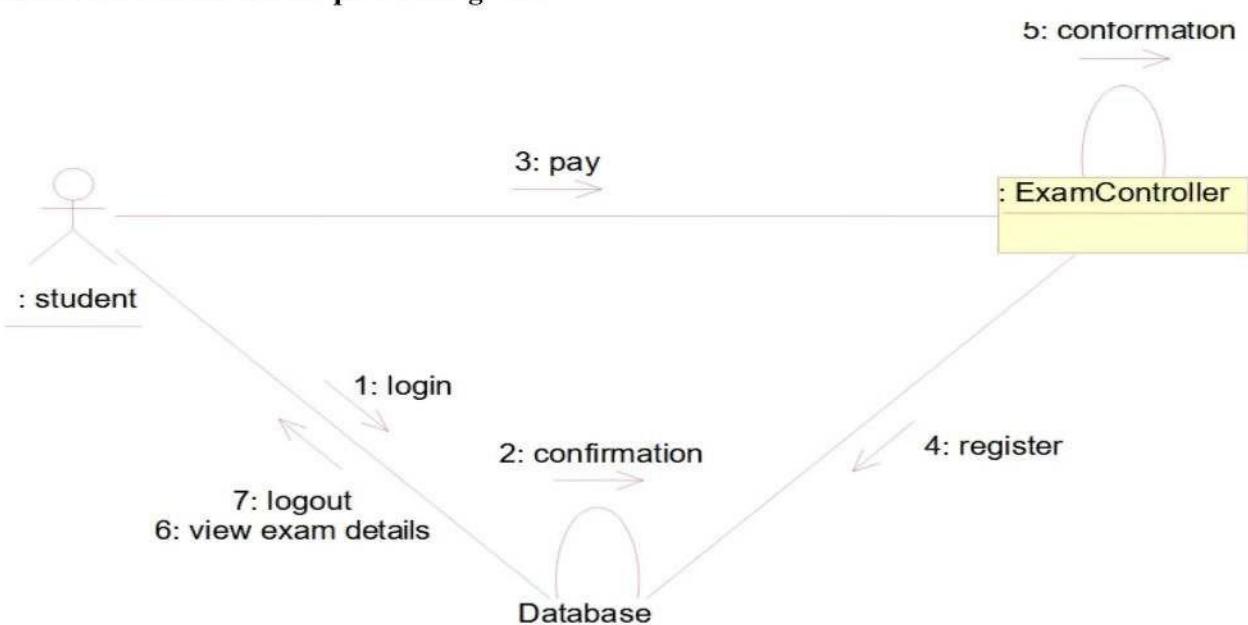
Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency.

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. An activity diagram shows the overall flow of control.



UML COLLABORATION DIAGRAM:-

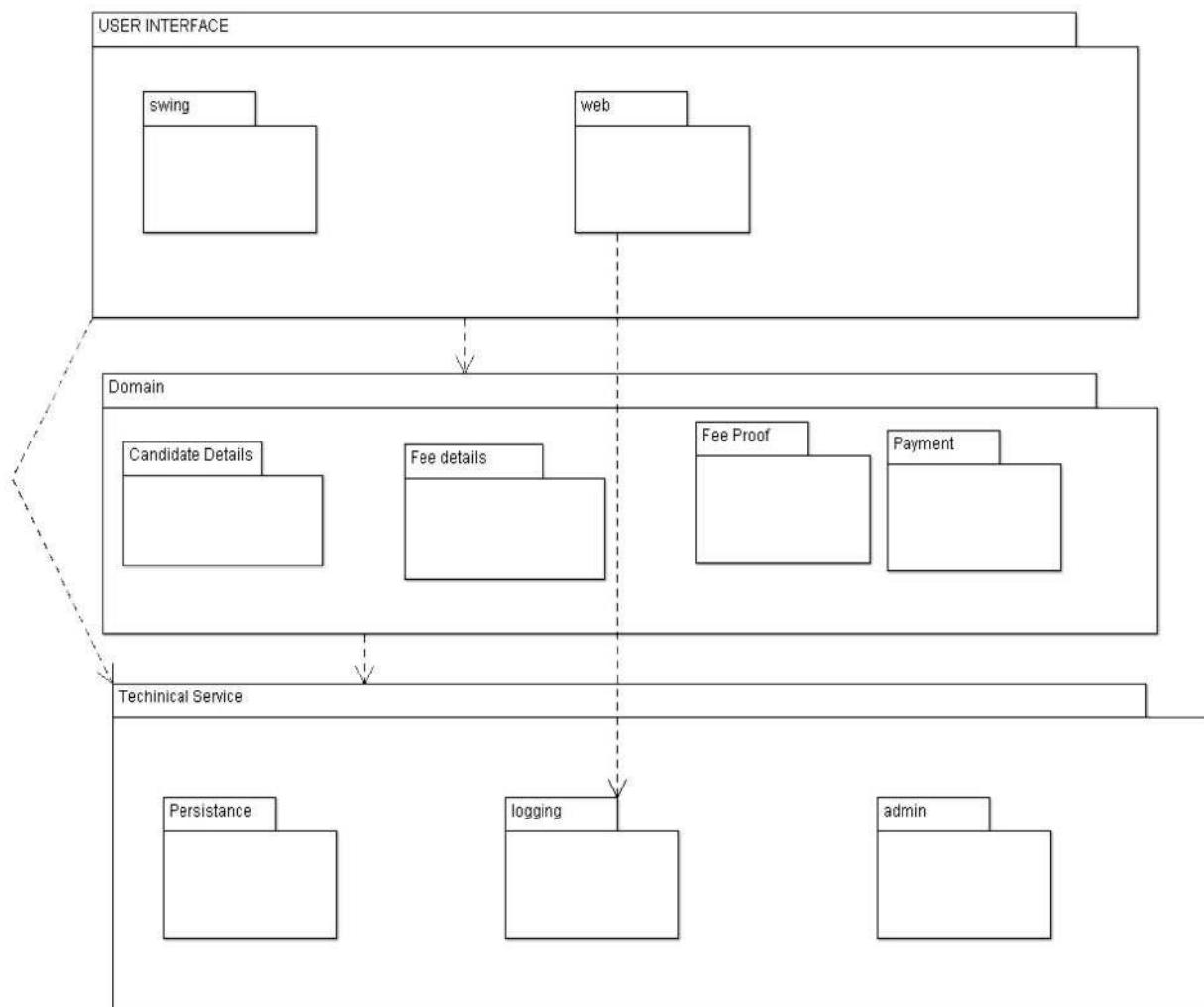
In collaboration diagram the object can be placed in anywhere on the diagram. The collaboration comes from sequence diagram.



UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



DOCUMENTATION OF PACKAGE DIAGRAM:

The three layers in the exam registration system are

- **User interface layer** – consists of swing and web. This layer describes how the candidates login to the application form.

- Domain layer – shows the activities that are performed inside the exam registration system. All the activities of candidate and admin are performed in this layer.
- Technical service layer – The sourcing and updating the details are performed in this layer.

UML TECHNICAL SERVICE LAYER:-

Name	Date of Birth	Qualification	Gender	Select Exam	Phone Number
Saravana	01-05-1997	BE	M	GROUP 1	9087349490
Surya	06-07-1998	BE	M	GROUP 2	8337849320
Vino	09-08-1998	BE	M	GROUP 3	7334739390

Register No	Name	Application No	Hall Ticket	Center	Date of Examiner
510615104078	Saravana	1014327	654789	SOC COLLEGE	06-10-2017
501615104088	Surya	1014329	698547	CACHET	05-11-2107
510615104105	Vino	1014320	647895	CIT	05-12-2017

SAMPLE CODE:-

PERSONAL INFO:-

```
import java.util.Vector;

public class personal Info {
    public String peeress_red;
    public string verify;
    public Vector my candidate;
    public void save() {
    }
    public void modify() {
    }
    public void new Operation() {
    }
}
```

ADMIN:-

```
import java.util.Vector;

public class admin {
    public string candidate Info;
    public Vector my Fees Details;
    public Vector my Debit card;
    public void save() {
    }
    public void Discard() {
    }
}
```

USER INTERFACE LAYER:-

The screenshot shows a web browser window titled "Exam Registration" with the URL "file:///C:/Users/d.balu/Desktop/Online%20exam%20registration/Registration%20form.html". The page is titled "ONLINE REGISTRATION FORM". It contains various input fields for user information:

- Select Exam: Bank Exam
- Name: Lakshmi D
- Email_id: lakshmi@gmail.com
- Father name: A dharalingam
- Address: 123.abc street, chennai-28
- DOB: 12-03-1989
- Gender: Male (radio button selected)
- Religion: Hindu
- Community: MBC
- Nativity: Tamil Nadu
- Qualification: ME
- Others: Type Writing
- Verification: 1324253 | 54758876

A "Submit" button is located at the bottom right of the form.

The screenshot shows a web browser window titled "Exam Registration" with the URL "file:///C:/Users/d.balu/Desktop/Online%20exam%20registration/netbanking.html". The page is titled "Net Banking Mode". It contains three input fields:

- Bank name: KVB
- Account No: 87502597908
- Amount: 500

A "submit" button is located at the bottom right of the form.

Result:-

Thus the Exam Registration has been done successfully by using Argo-UML.

EX.NO: 4	STOCK MAINTENANCE SYSTEM
----------	--------------------------

AIM:

To design Stock maintenance System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN

To simplify the process of applying stock maintenance system, software has been created by designing through ARGO-UM tool. This software is designed for supporting the computerized Stock Maintenance System. In this system, the customer can place order and purchase items with the aid of the stock dealer and central stock system. This order is verified and the items are delivered to the customer.

The process of Stock Maintenance System is that the customer login to the particular site to place the order for the customer product. The Stock Maintenance System are described sequentially through steps

- The customer login to the particular site.
- They fill the customer details.
- They place the order for their product.
- The vendor login and views the customer details and orders.

Functionalities of Customer:

To purchase an item, Constraints checked here includes

- Maximum price of the item.
- Number of item left.
- Conflicts in timing with other orders of some items.

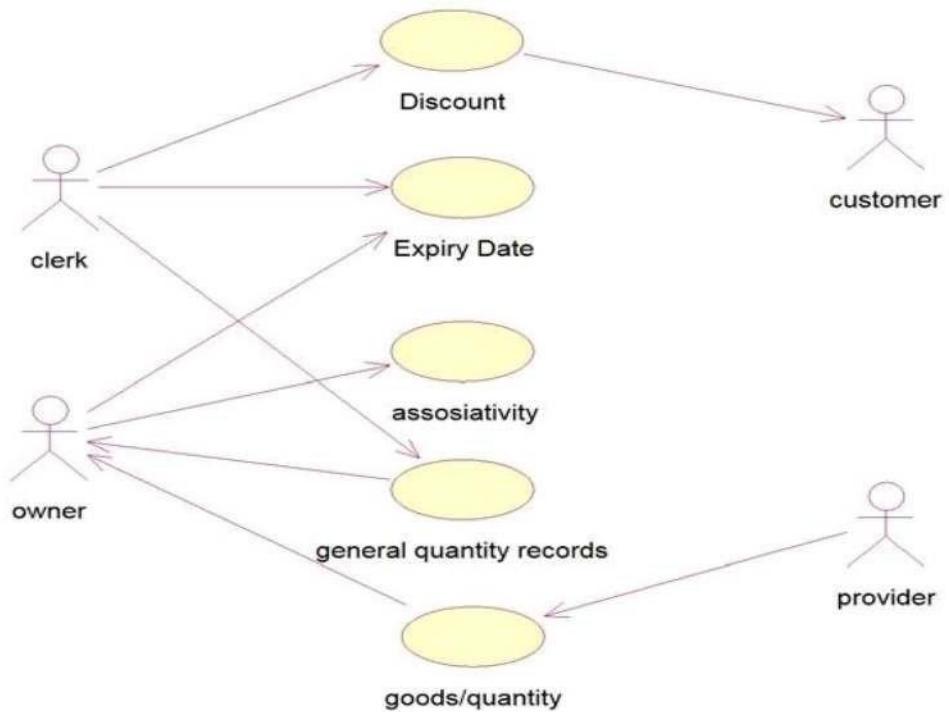
Functionalities of Stock Manger:

- Can view items currently available.
- Can view information of the entire customer who has order the items.
- Conflicts with the items available at the stock room and the customer order.
- Can edit the items orders such as number of items left or to increase or decrease the number of availability.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

ACTOR: - Stock Manager, Manager, Admin

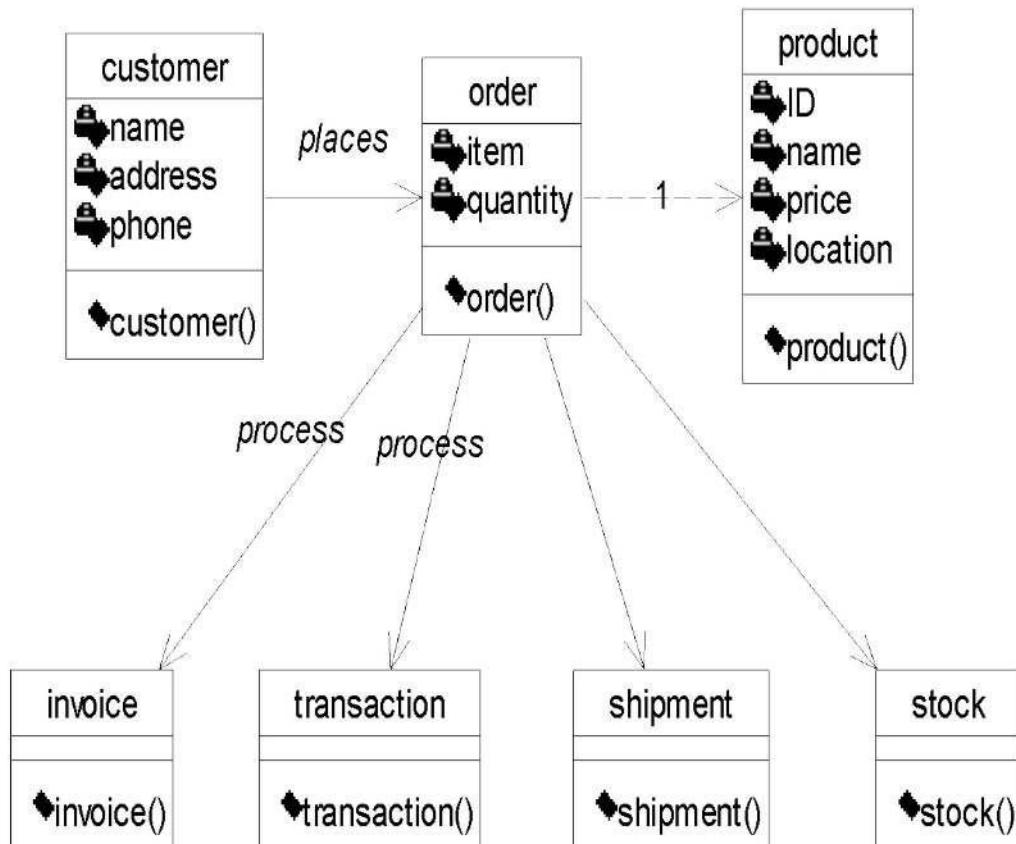
Use Case: - Stock Clearance, login, Stock, Export, Stock List, New Stock, and Payment.

The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

- Login
- Manager
- Stock Clearance
- payment
- Stock

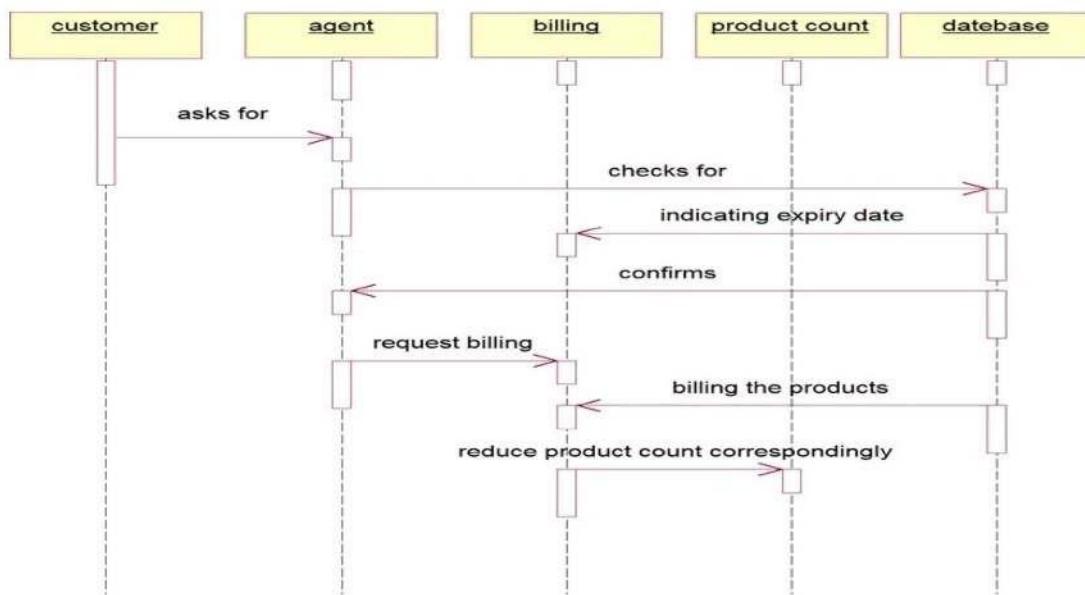
The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

- Customer
- Manager
- Stock Manager
- Admin

It can be 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.

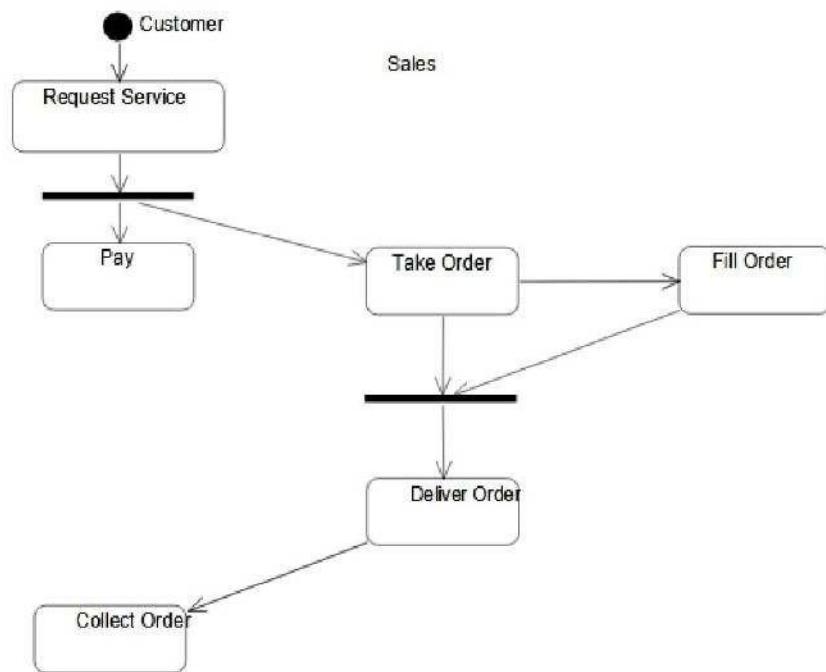
UML ACTIVITY DIAGRAM:-

Description:-

Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

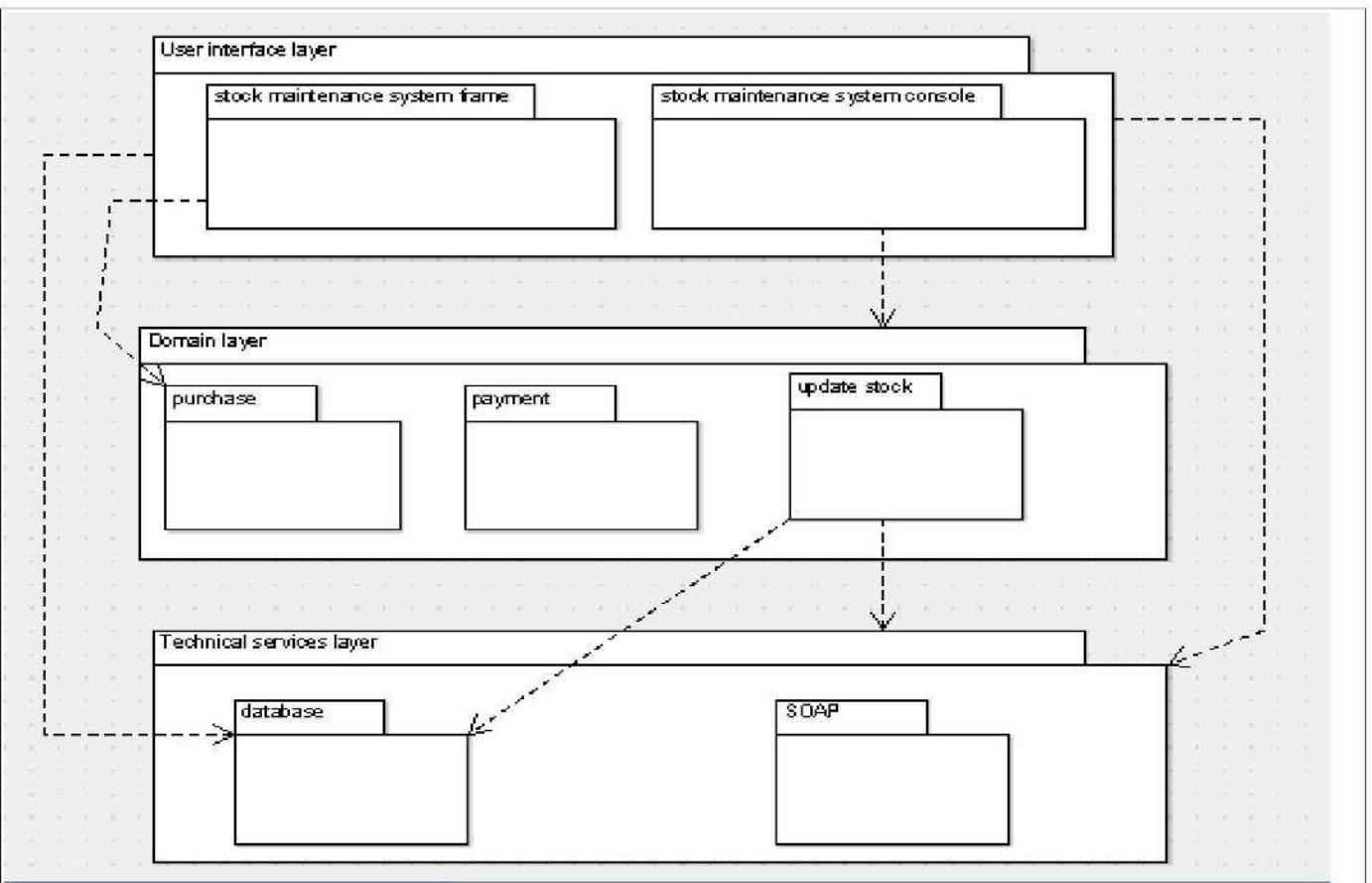
- How activities are coordinator to provide a service.
- The events needed to achieve some operation.
- How events in a single use case relate to one another.
-



UML PACKAGE DIAGRAM:-

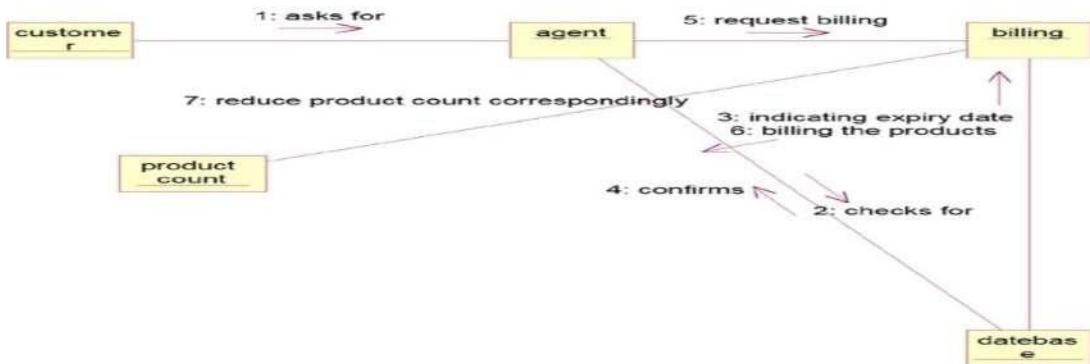
Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



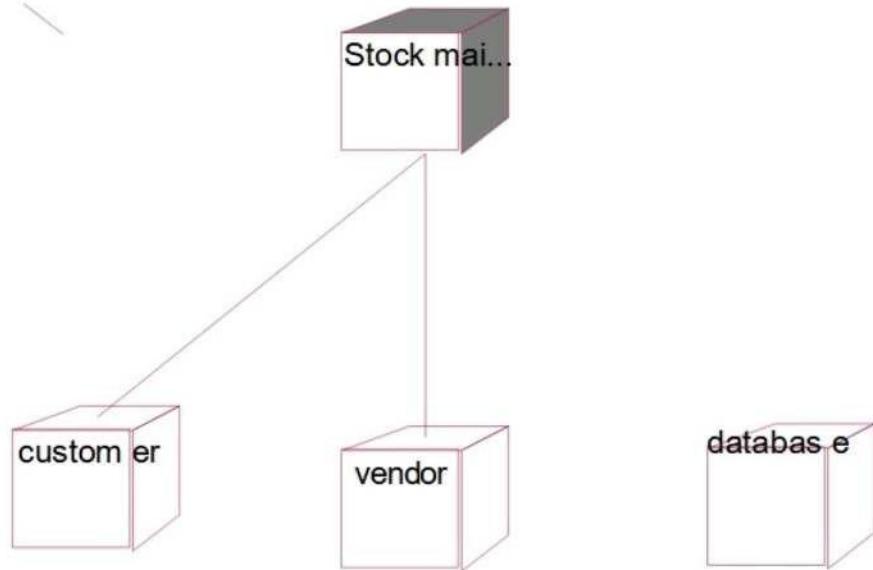
UML COLLABORATION DIAGRAM:-

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.



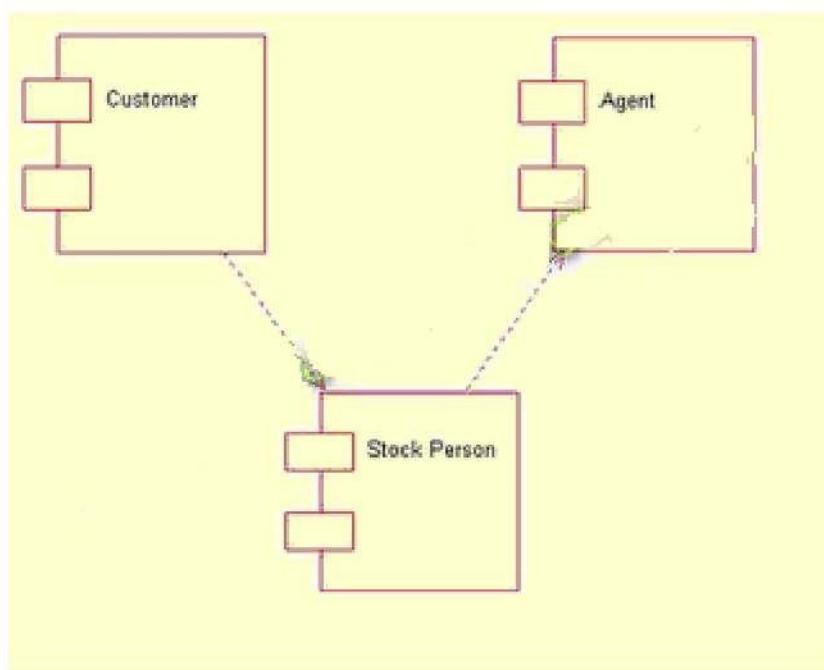
DEPLOYMENT DIAGRAM:-

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



UML COMPONENT DIAGRAM:-

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



UML TECHNICAL SERVICE LAYER:-

S.NO	USERNAME	PASSWORD
1	Santhosh	54792
2	Vignesh	12345
3	Sharfu	67890

S.no	Amount	Name of the Bank	Branch of the bank	Contact
1	10,000	City Union	Vellore	9765432198
2	5,000	ICICI	Chennai	9743278367
3	15,000	SBI	Ranipet	8764352789

SAMPLE CODE:-

```
import java.util.Vector; import java.util.Vector;
public class login { public class Stock Clarence {
    public Integer password; public string old_stocks;
    public string Login_id; public integer amount;
    public String Name; public float Amount_loss;
    public varchar Aces; public Vector mylogin;
    public Integer Newark; public Vector my Stock;
    public Vector my Stock Clarence; public Vector my Manager;
    public Vector my Manager;
    public void update() { public void update() {
    } } }
    public void save() { }
    public void login() { }
```

}

USER INTERFACE LAYER:-

The figure consists of three horizontal screenshots of a web-based application titled "Stock Maintenance System".

- Screenshot 1:** Shows the "Login" page. It has input fields for "User Name" and "Password", and a "Submit" button.
- Screenshot 2:** Shows the "Registration" page. It has input fields for "Name", "Email", "Address", and "Contact", and a "Submit" button.
- Screenshot 3:** Shows the "Registration" page after a successful submission. A message "Your Stock Is Registered Successfully" is displayed prominently in the center.

The browser address bar shows the URL "localhost:8080/stockManagement". The status bar at the bottom of each screenshot lists various menu items such as "Home", "Logout", "Registration", "About Us", "Help", "Feedback", "Suggestion Box", "Contact Us", "Privacy Policy", and "Help & Support".

Result:-

Thus the Stock Maintenance System has been done successfully by using Argo-UML.

EX.NO: 5	ONLINE COURSE REGISTRATION
----------	----------------------------

AIM:

To design Online Course Registration System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN :-

To simplify the process of applying Online Course Registration, software has been created by designing through ARGO-UM tool.

The exam registration is an application in which applicant can register themselves for the exam. The details of the students who have registered for the examination will be stored in a database and will be maintained. The registered details can then be verified for any fraudulent or duplication and can be removed if found so.

PROBLEM STATEMENT:-

Exam registration system is used in the effective dispatch of registration from to all of the students this system adopts a comprehensive approach to minimize the manual work and schedule resources, time in cogent manner the core of the system is to get the online registration from (with details such as name, oregano , etc.,) filled by the student whose statement is verified for it is genuineness by the exam registration system with respect to the already existing information in the database.

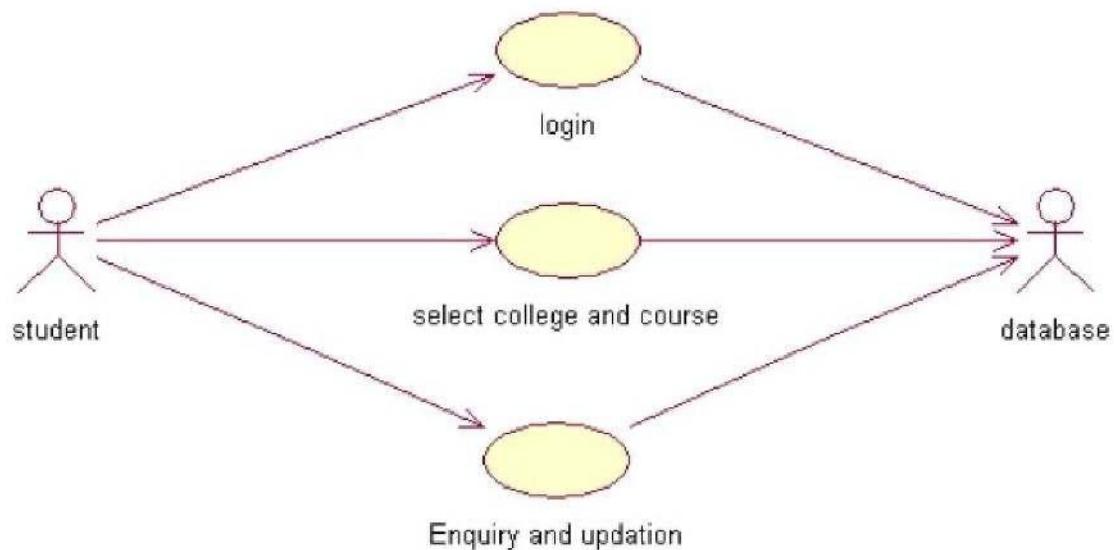
The process of students accessing the registration application and applying for the examination by filling out the form with proper details and then the authorities verify those details given for truth and correctness are sequenced through steps

- a. The students access exam registration application.
- b. They fill out the form with correct and eligible details.
- c. They complete the payment process.
- d. The authorities verify or check the details.
- e. After all verification the exam registration database is finalized

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that make use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

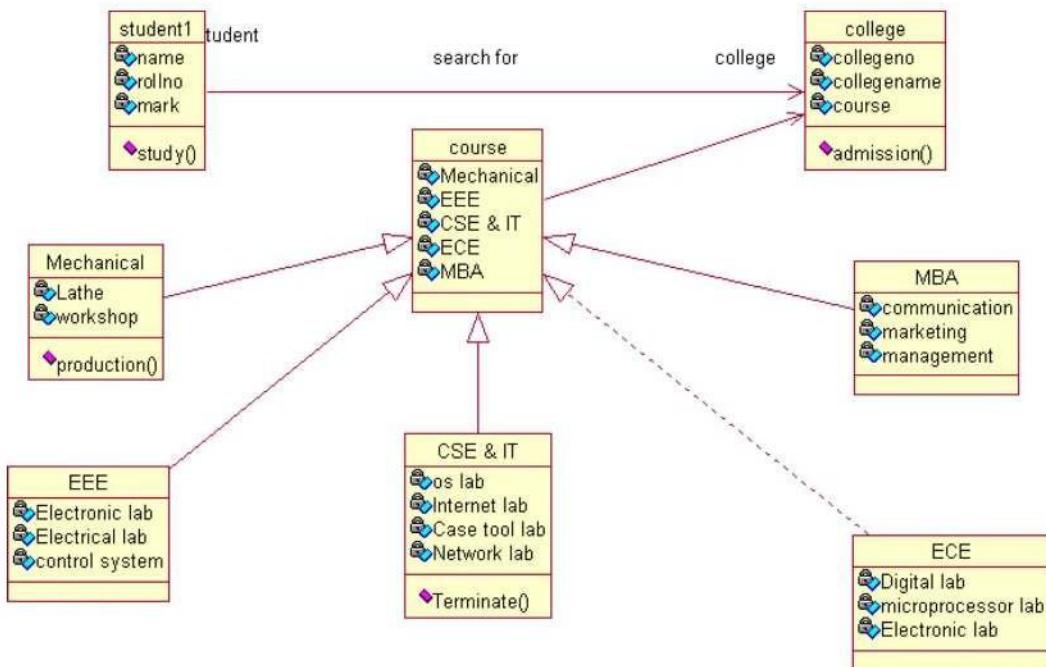
The actors in this use case diagram are Students, Interface and Database. The use cases are the activities performed by actors.

- a. Student fills outs the form in the form filling process.
- b. The interface checks and validates registered details.
- c. Then the database is searched for details and verified.
- d. Database stores the details and returns acknowledgment
- e. The interface checks and validates registered details.
- f. Then the database is searched for details and verified.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

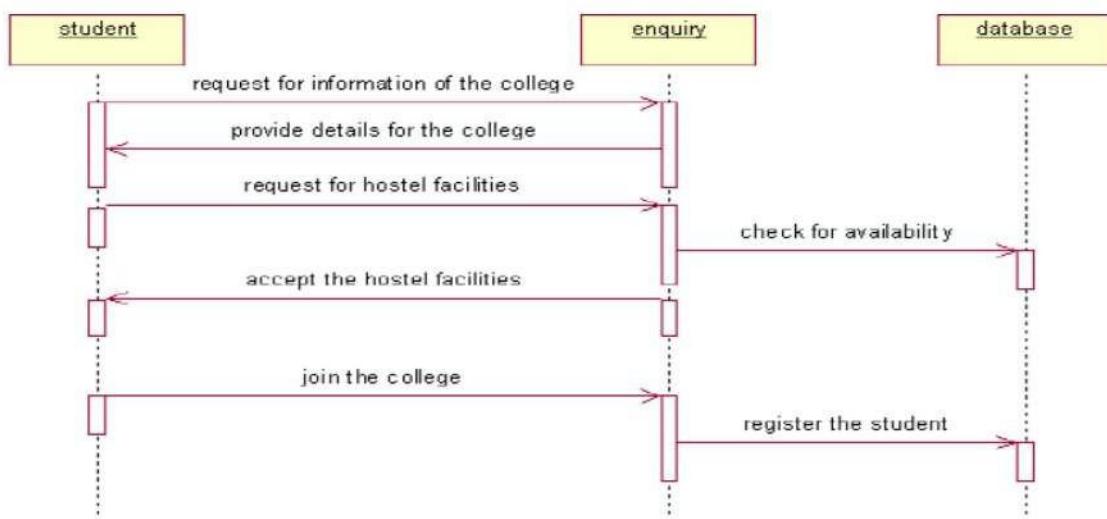
- Course Management
- Course Details
- Registration
- Payment
- Bank

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

The sequence diagram describes the sequence of steps to show

- The candidate selects the exam and enters their detail in the online registration form.
- Then the candidate selects the payment mode like Credit card, Debit card and Net banking.
- If the payment is not completed then it goes for repayment and after completion of the payment the application will be generated.
- The detail of exam is viewed by the candidate.

UML ACTIVITY DIAGRAM:-

Description:-

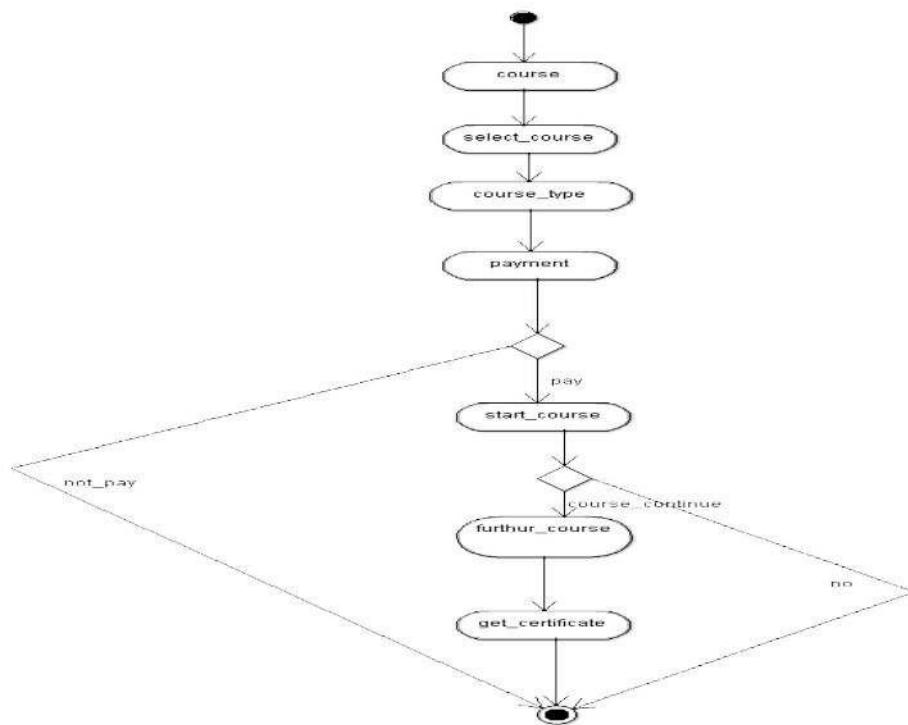
Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

How activities are coordinator to provide a service.

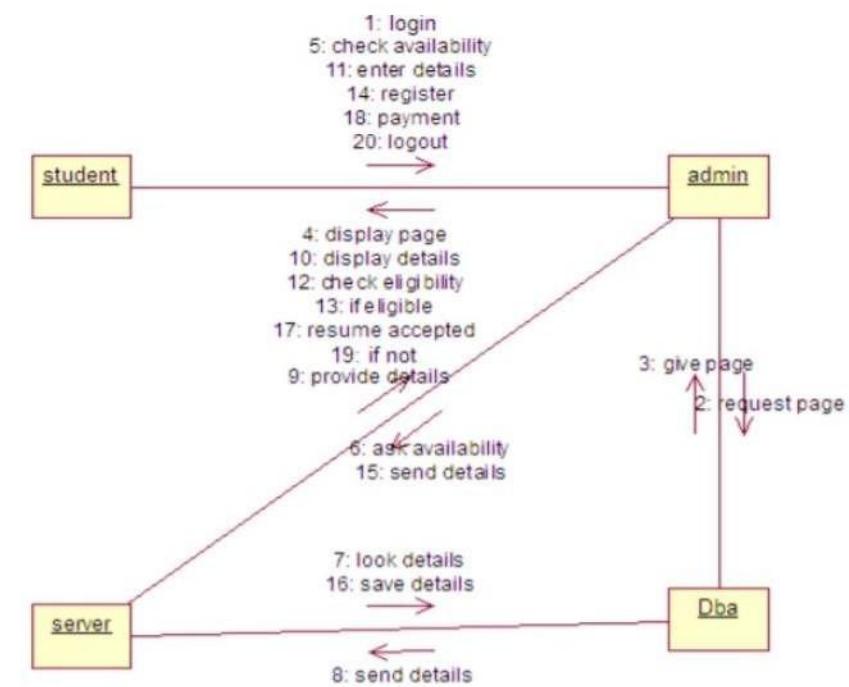
The events needed to achieve some operation.

How events in a single use case relate to one another.



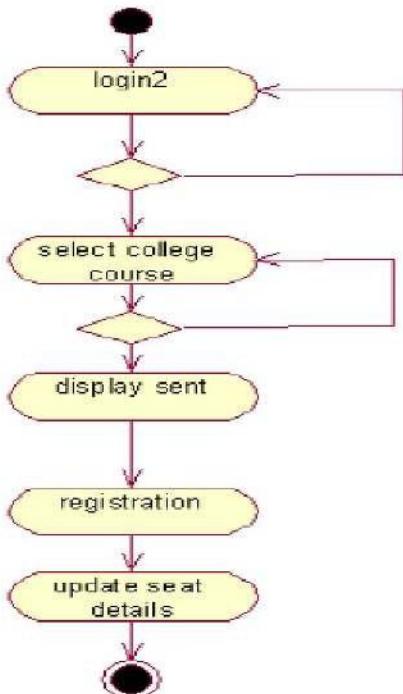
UML COLLABORATION DIAGRAM:-

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.



STATE CHART DIAGRAM:-

It includes all the activities of particular project and various steps using join and for Ks.



DOCUMENTATION OF STATE CHART DIAGRAM:-

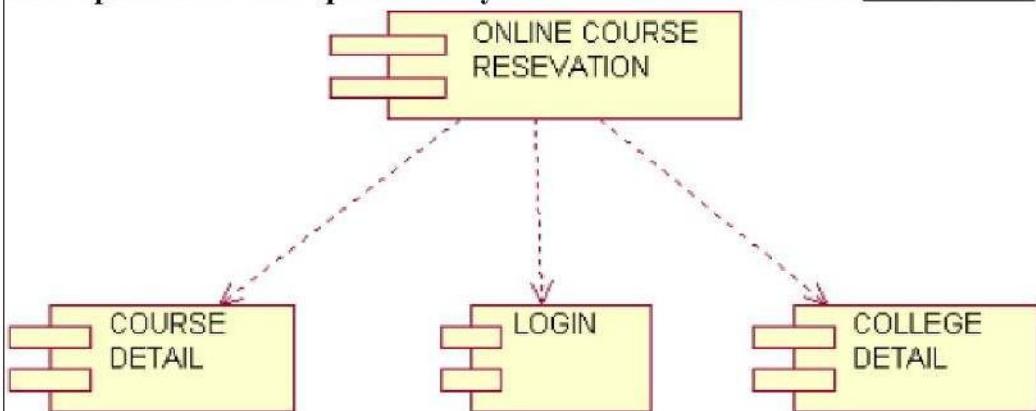
- The state chart diagram starts with the main screen the candidate will enter into the registration form he / she enters the necessary details.
- The candidate moves to the payment option he /she enters the amount and waits for the payment completion.
- The Admin performs the application generation and the hall ticket will be viewed by the candidate through the main screen.

If the payment doesn't complete it moves to the end state.

UML COMPONENT DIAGRAM:-

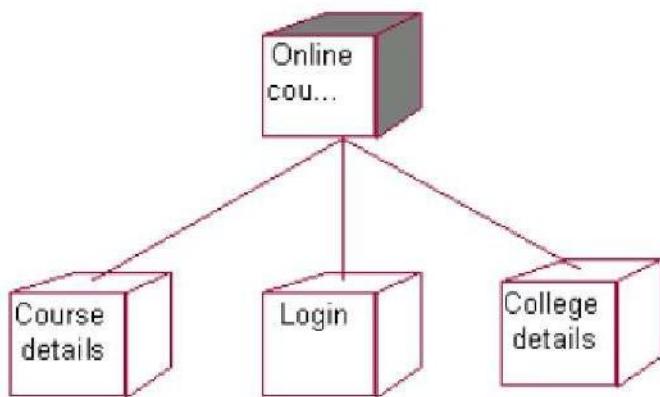
It includes all the activities of particular project and various steps using join and forks
The component diagram is represented by figure dependency and it is a graph of design of figure dependency. The component diagram's main purpose is to show the structural relationships between the components of a systems. It is represented by boxed figure.

and Dependencies are represented by communication association



UML DEPLOYMENT DIAGRAM:-

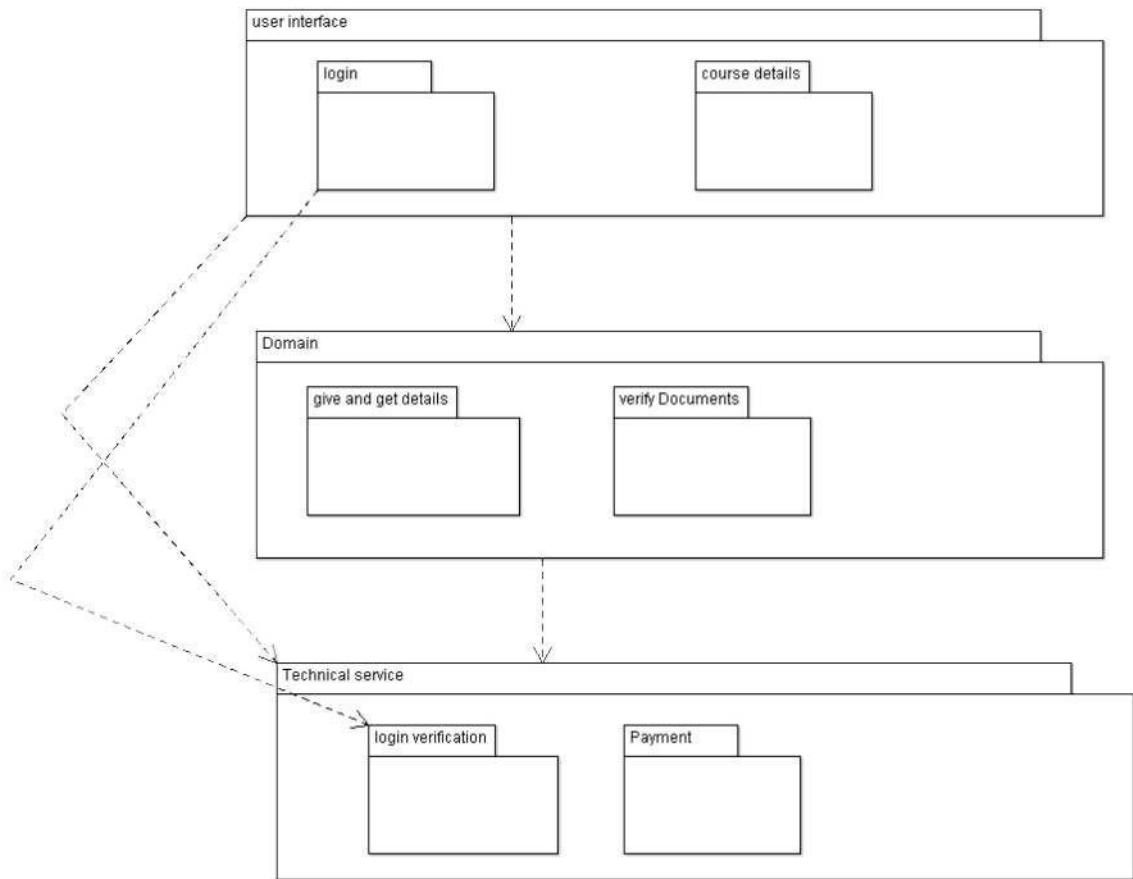
DEPLOYMENT Diagram It is a graph of nodes connected by communication association. It is represented by a three dimensional box. A deployment diagram in the unified modeling language serves to model the physical deployment of artifacts on deployment targets. Deployment diagrams show "the allocation of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



UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



DOCUMENTATION OF PACKAGE DIAGRAM:

The three layers in the exam registration system are

- **User interface layer** – consists of swing and web. This layer describes how the candidates login to the application form.
- **Domain layer** – shows the activities that are performed inside the exam registration system. All the activities of candidate and admin are performed in this layer.
- **Technical service layer** – The sourcing and updating the details are performed in this layer.

UML TECHNICAL SERVICE LAYER:-

Name	Date of Birth	Qualification	Gender	Select Course	Phone Number
Saravana	01-05-1997	BE	M	NIT	9087349490
Surya	06-07-1998	BE	M	II	8337849320
Vino	09-08-1998	BE	M	CIT	7334739390

SAMPLE CODE:-

PERSONAL INFO:-

```
import java.util.Vector;
public class personal Info {
    public String peeress_red;
    public string verify;
    public Vector my candidate;
    public void save0 {
    }
    public void modify0 {
    }
    public void new Operation0 {
    }
}
```

ADMIN:-

```
import java.util.Vector;
public class admin {
    public string candidate Info;
    public Vector my Fees Details;
    public Vector my Debit card;
    public void save0 {
    }
    public void Discard0 {
    }
}
```

USER INTERFACE LAYER:-

The screenshot shows a web browser window titled "Exam Registration" displaying an "ONLINE REGISTRATION FORM". The form contains the following fields:

- Select Exam: Bank Exam
- Name: Lakshmi.D
- Email_id: lakshmi@gmail.com
- Father name: A.dharmalingam
- Address: 123,abc street, chennai-28
- DOB: 12-03-1989
- Gender: Male Female
- Religion: Hindu
- Community: MBC
- Nativity: Tamil Nadu
- Qualification: ME
- Others: Type Writing
- Verification: 1324253 54758876

A "Submit" button is located at the bottom right of the form area.

Result:-

Thus the Online Course Reservation System has been done successfully by using Argo-UML.

EX.NO:6	E-TICKETING
---------	-------------

AIM:

To design E-Ticketing System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN:

To simplify the process of applying e-ticketing, software has been created by designing through ARGO-UM tool.

In the E-Ticketing system the main process is an 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 Availability of seats or else canceling the process.

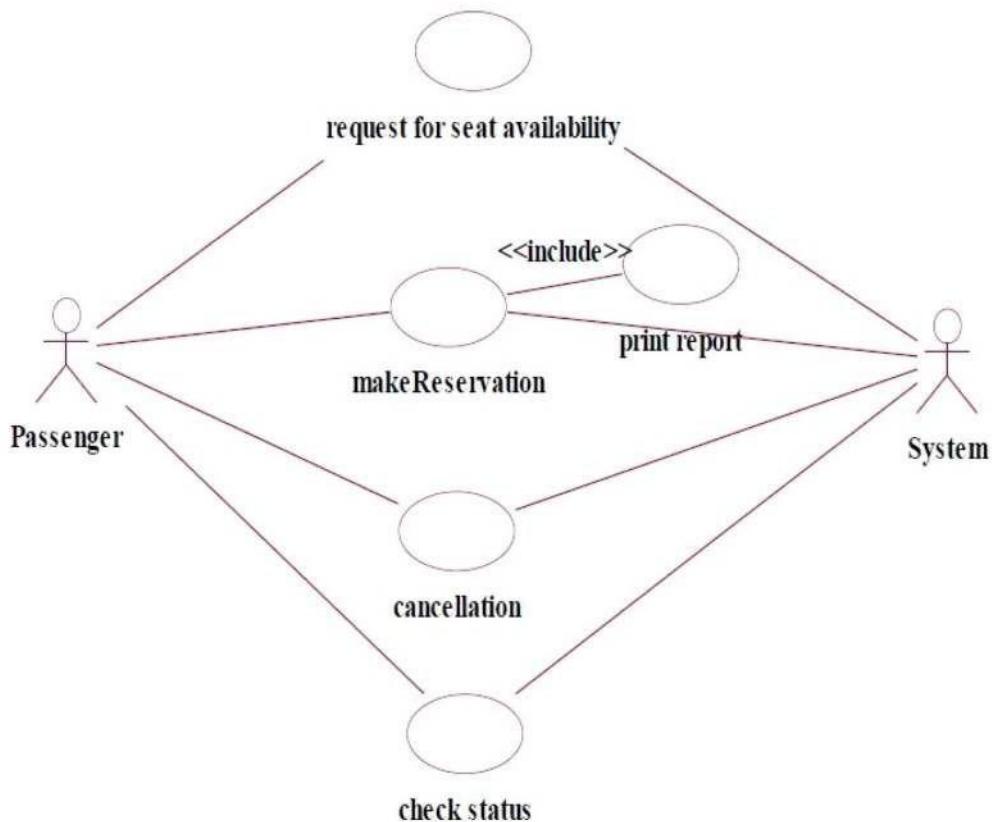
PROBLEM STATEMENT:-

- The E-Ticketing system is the initial requirement to develop the project about the mechanism of the E-ticketing system what the process do at all.
- The requirement are analyzed and refined which enables the end users to efficiently use the E-ticketing system.
- The complete project is developed after the whole project analysis explaining about scope and project statement is prepared.
- The main scope for this project is the applicant should reserve for the flight ticket.
- First the applicant wants to login to the database after that the person wants to fill their details.
- Then the database will search for ticket or else the person will cancel the ticket if he/she is in not need.
- The complete project is developed after the whole project analysis explaining about scope and project statement is prepared.
- The scope for this project is the applicant should reserve for the flight ticket.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that make use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

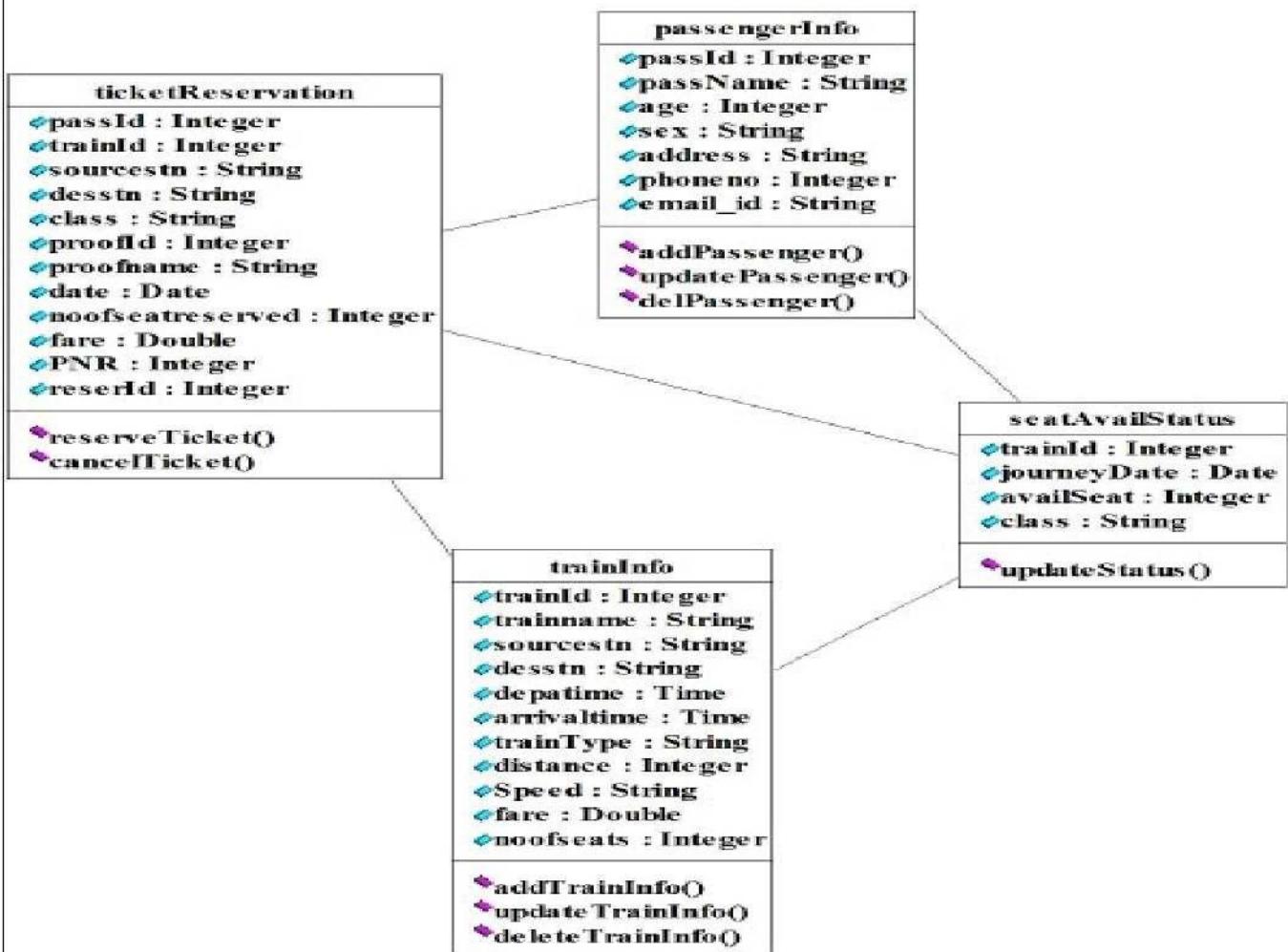
The actors in this use case diagram are applicant, and E-ticketing Data Base. The use cases are the activities performed by actors.

The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

This class diagram has two classes applicant, E-Ticketing Data Base.

Applicant -logins the E-Ticketing and filling the required data fields.

E-Ticketing Data Base-verify the login and filling the details and selected applicant details are stored in it.

This diagram shows the classes associated with this system and the way how they are linked with each other. This diagram also shows the attributes and methods of the class. The

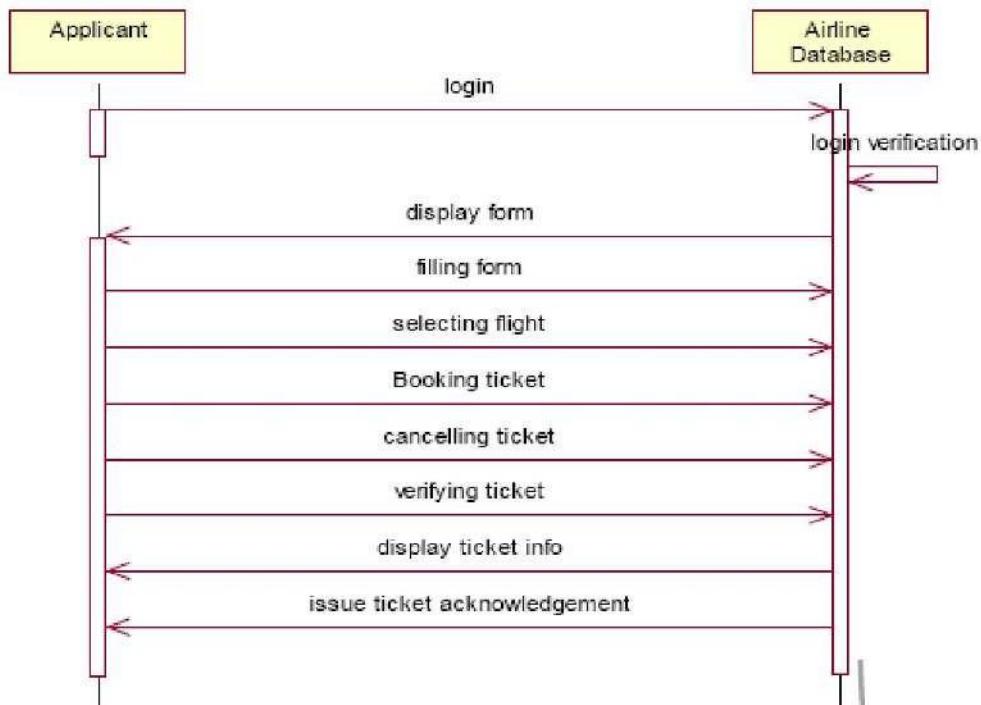
first partition shows the name of the class and second shows the attributes and third shows the methods.

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

This sequence diagram describes the sequence of steps to show

- Applicants are used to login the form and then it's verify the username and password.
- If the password and username are correct then applicants are used to login the filling details.
- Applicants are used to selecting the Movies and book the tickets.
- Now the E-Ticketing Data Base verify the filling Details.
- And then the E-Ticketing Data Base displays the ticket information.
- In case of any sudden change of the plan, The applicant can cancel the ticket

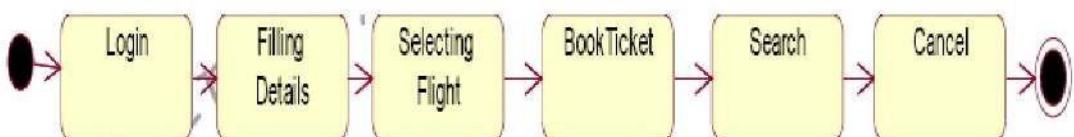
UML STATE CHART DIAGRAM:-

Description:-

Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

- How activities are coordinator to provide a service.
- The events needed to achieve some operation.
- How events in a single use case relate to one another.

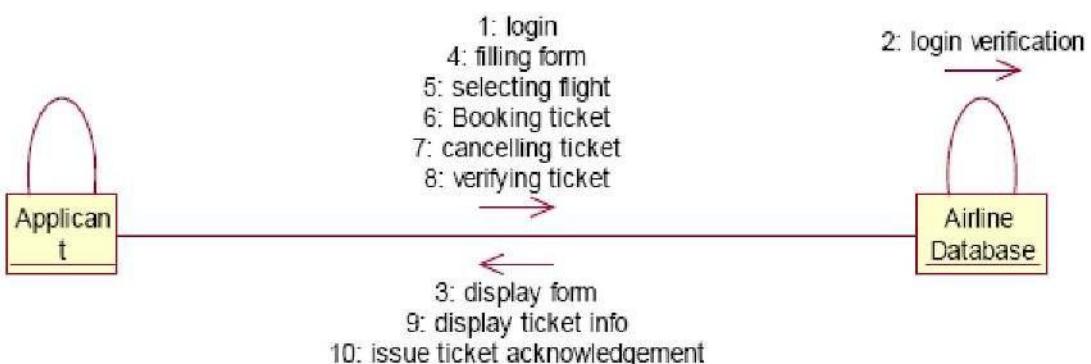


DOCUMENTATION OF STATE CHART DIAGRAM:-

- This activity diagram describes the behavior of the system.
- First state is login where the applicant login to the E-Ticketing system.
- The next state is filling details the applicant are used to fill the form.
- Then applicant used to selecting the flight.
- The applicant appears for book ticket and search details from E-Ticketing Data Base.

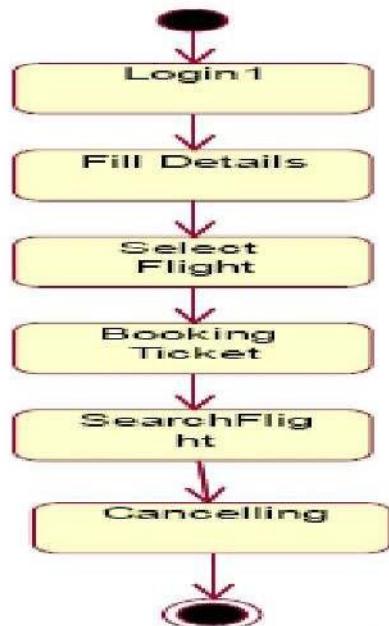
UML COLLABORATION DIAGRAM:-

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.



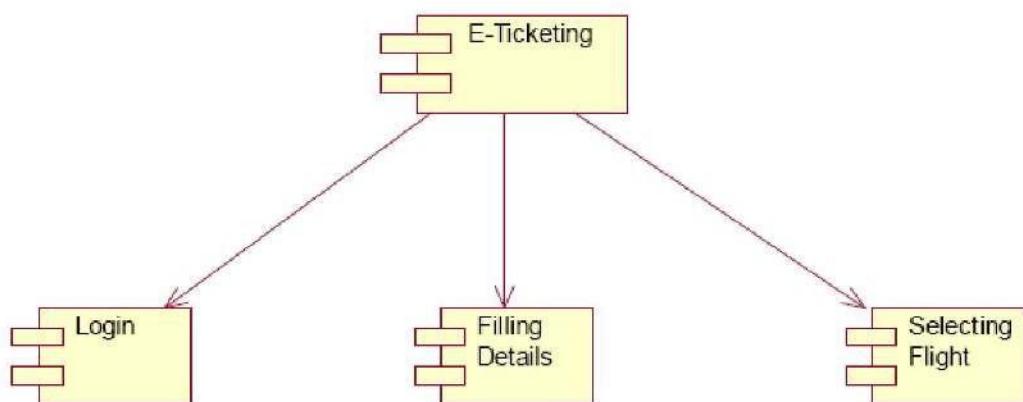
UML ACTIVITY DIAGRAM:-

Activity diagrams are graphical representations of workflows of step wise 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.



UML COMPONENT DIAGRAM:-

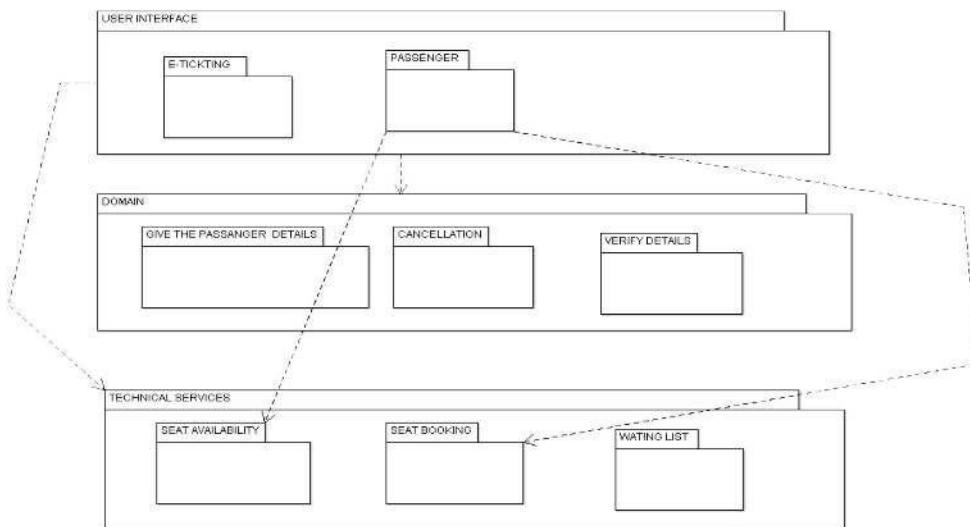
The component diagrams main purpose is to show the structural relationships between the components of a system. It is represented by boxed figure. Dependencies are represented by communication association.



UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



UML TECHNICAL SERVICE LAYER:-

S. No	Passenger	Train_No	From	Destination	Spare
1.	Naveen	123782	Vellore	Sooty	300
2.	Sathish HariPrakash	347832	Amur	Bangalore	400
3.	Ramesh Chandru	434792	Katydid	Delhi	1200

S. No	Passenger	Train_No	From	Destination	Spare
1.	Hari Prasad	123782	Vellore	Sooty	300
2.	Sandy	347832	Amur	Bangalore	400
3.	Siva	434792	Katydid	Delhi	1200

SAMPLE CODE:-

Booking:-

```
import java.util.Vector;
public class Booking {
    public float Payment;
    public varchar timing;
    public string name;
    public integer quantity;
    public Vector my Airlines;
    public Vector my Railway;
    public void print() {
    }
    public void save() {
    }
}
```

```
import java.util.Vector;
public class Booking {
    public float Payment;
    public varchar timing;
    public string name;
    public integer quantity;
    public Vector my Airlines;
    public Vector my Railway;
    public void print() {
    }
    public void save() {
    }
}
```

USER INTERFACE LAYER:-

The screenshot shows the homepage of the Airline Reservations website. At the top, there's a navigation bar with links for HOME, REGISTRATION, TRAVEL NEWS, CONTACT US, HELP, and ABOUT US. Below the navigation bar, the main title "AIRLINE RESERVATIONS" is displayed in large, bold letters. Underneath the title, there's a section titled "FLIGHT SCHEDULES". On the left side, there's a sidebar with links for ADMINISTRATORS, Scheduling, Passengers List, Feedback view, Flight, and Report. The main content area shows a table of flight schedules:

	Flightname	Time	Source	Destination	Duration	Seats	Price
Edit Delete	BRITISH	27/06/2010 05:55:00 AM	HYD	UK	116	1200	
Edit Delete	INDIAN	27/06/2010 04:45:00 AM	HYD	US	116	1200	
Edit Delete	INDIAN AIRLINES	27/06/2010 24:50:00 AM	HYD	US	120	2000	
Edit Delete	KINGFISHER	27/06/2010 04:55:00 AM	HYD	NW	120	4000	

At the bottom of the page, there's a footer with the text "AIRLINE RESERVATIONS Copyright © 2010".

The screenshot shows a "FLIGHT_DETAILS" window. The window has several input fields and buttons. On the left, there's a sidebar with labels: FLIGHT_ID, FLIGHT_NAME, SOURCE, DEPARTURE, FLIGHT_CLASS, FLIGHT_CHARGES, and SEATS. To the right of these labels are corresponding input fields. There are also "SEARCH", "DESTINATION", "ARRIVAL_TIME", and "Economy" buttons. At the bottom of the window, there are three buttons: "ADD FLIGHT", "UPDATE", and "DELETE".

Result:-

Thus the E-Ticketing has been done successfully by using Argo-UML.

EX.NO:7	SOFTWARE PERSONAL MANAGEMENT SYSTEM
---------	-------------------------------------

AIM:

To design Software Personal Management System by using Argo-UML tool.

PROCEDURE:

- The software to be designed will control a simulated software personnel management system.
- This software is designed for the process of knowing the details of a person works in a software company. The details are being stored in the central management system for the crosschecking the person's details.

PROBLEM ANALYSIS AND PROJECT PLAN

To simplify the process of applying Software Personal Management, software has been created by designing through ARGONAUT tool.

The employee management system is used to manage our personnel things such as maintaining databases in offices etc. this project is easy for the CEO to handle the details. This is personally used for CEO.

PROBLEM STATEMENT:-

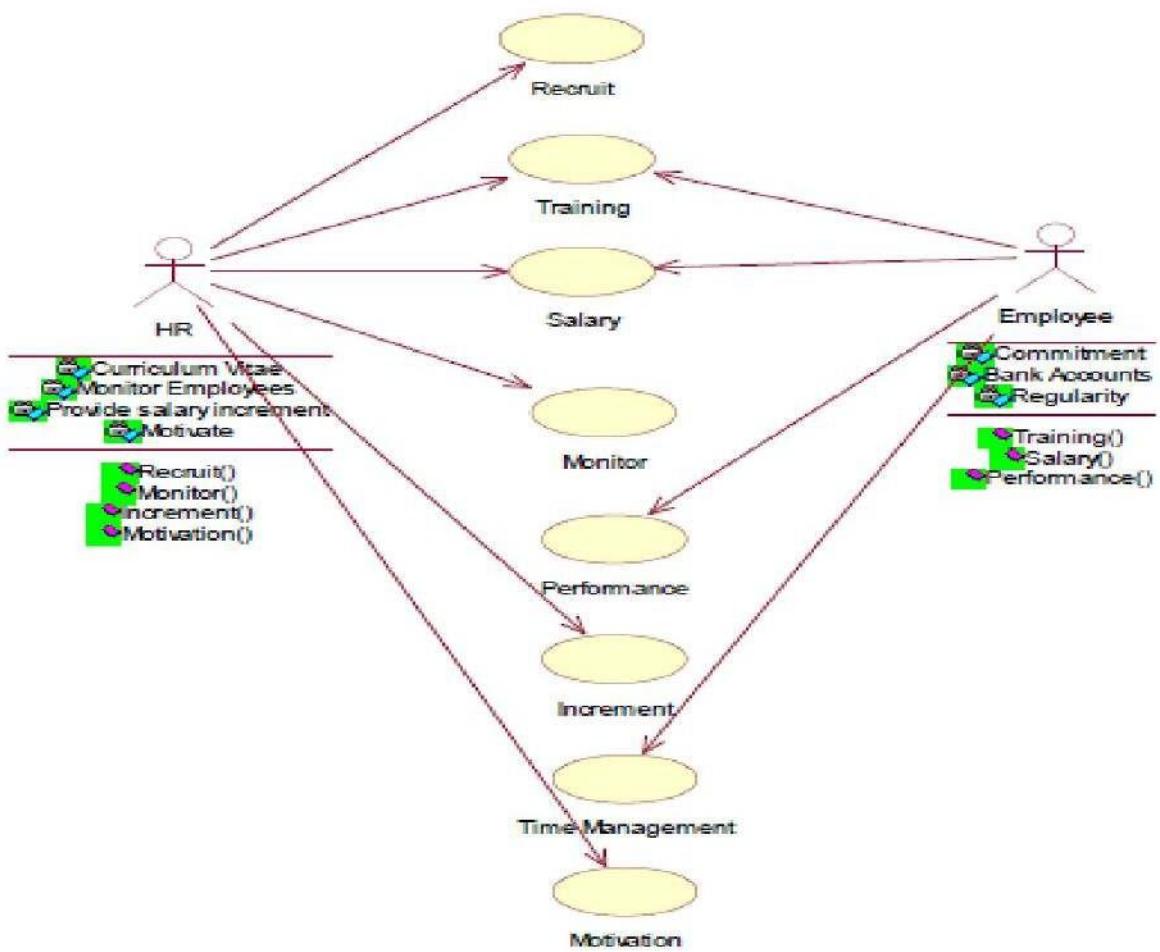
The CEO must enter the name and password to login the form and select the particular employee to view the details about that employee and maintaining the employee details personally. This process of employee management system are described sequentially through following steps,

- The CEO login to the employee management system.
- He/she search for the list of employees.
- Then select the particular employee.
- Then view the details of that employee.
- After displaying the employee details then logout.
- After that select the particular employee.
- Then again view the details of that employee.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

The actors in this use case diagram are Student, Staffs and Library Database. The use cases are the activities performed by actors.

The use case diagram in the employee management system illustrates the sequence of sequencing and describing an interaction between a CEO and a system.

Login:

This use case gives as entry to the CEO and the database.

List of employee:

This will create the situation for the CEO to select particular employee from the available list.

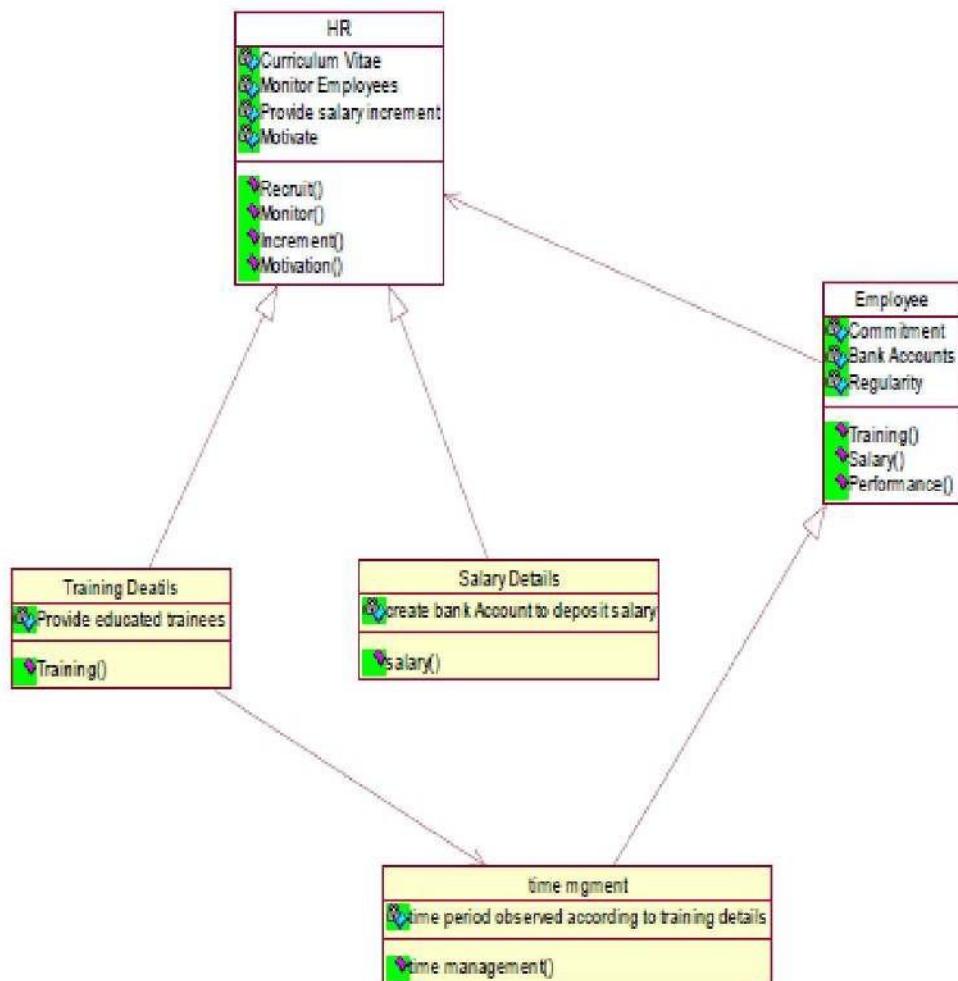
Employee details:

The CEO can able to view the details of the employee using this use case.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

The Classes used in this project are

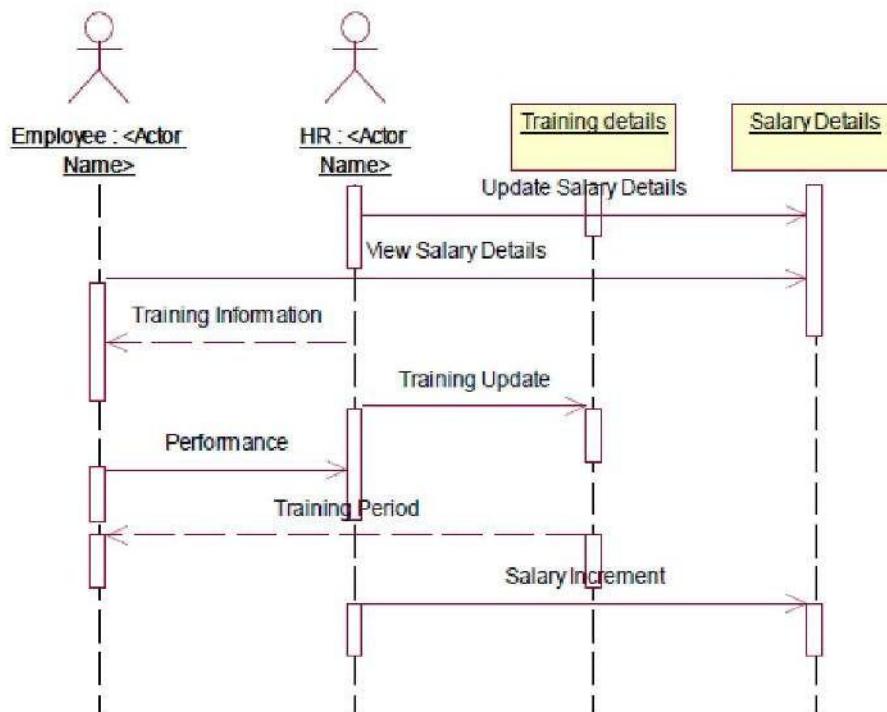
- **CEO:** The CEO has to login the form by specifying the name and password of him.
- **Database:** The database checks whether the CEO has given the name and password accordingly if not the error message will be displayed.
- **Available employees:** The database is connects to the list of available employees and the CEO if wants then select the employee from it.

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

The CEO must enter his name and password to login the employee management system. The verification process is undergone by the database.

If the details are correct he can enter to the system otherwise error is displayed. After login the details of the particular employee is viewed by the CEO.

Finally he is logged out from the system.

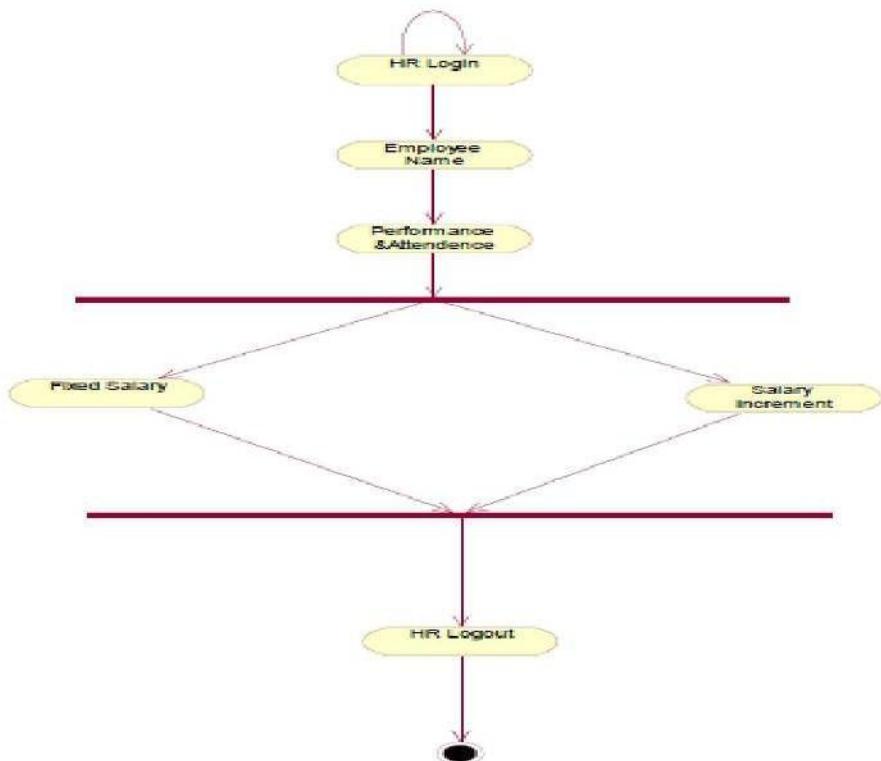
UML ACTIVITY DIAGRAM:-

Description:-

Activity diagrams are graphical representations of workflows of step wise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

- How activities are coordinator to provide a service.
- The events needed to achieve some operation.
- How events in a single use case relate to one another.



DOCUMENTATION OF ACTIVITY DIAGRAM:-

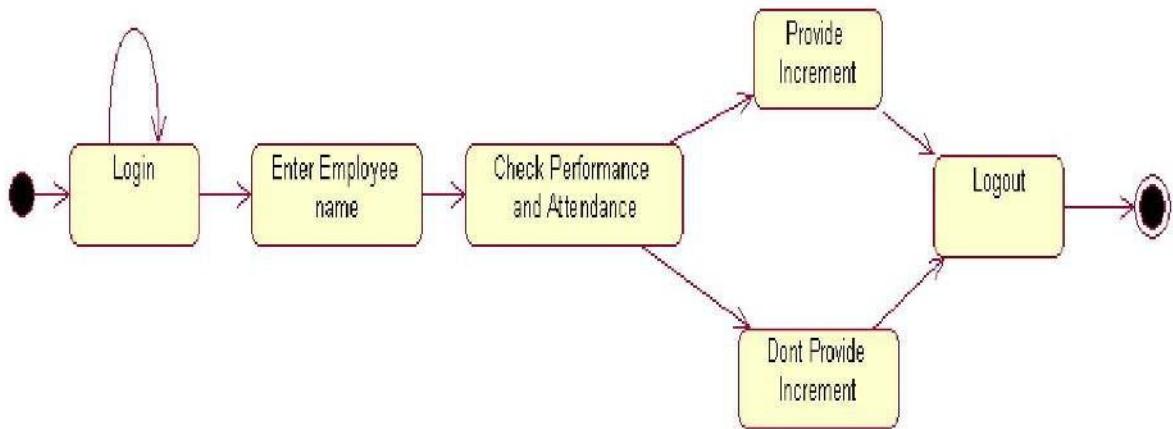
- The purpose of state chart diagram is to understand the algorithm involved in performing a method. It is also called as state diagram.
- A state is represented as a Round box, which may contain one or more compartments. An initial state is represented as small dot. A final state is represented as circle surround Ding a small dot.

UML COLLABORATION DIAGRAM:-



UML STATE TRANSITION DIAGRAM:-

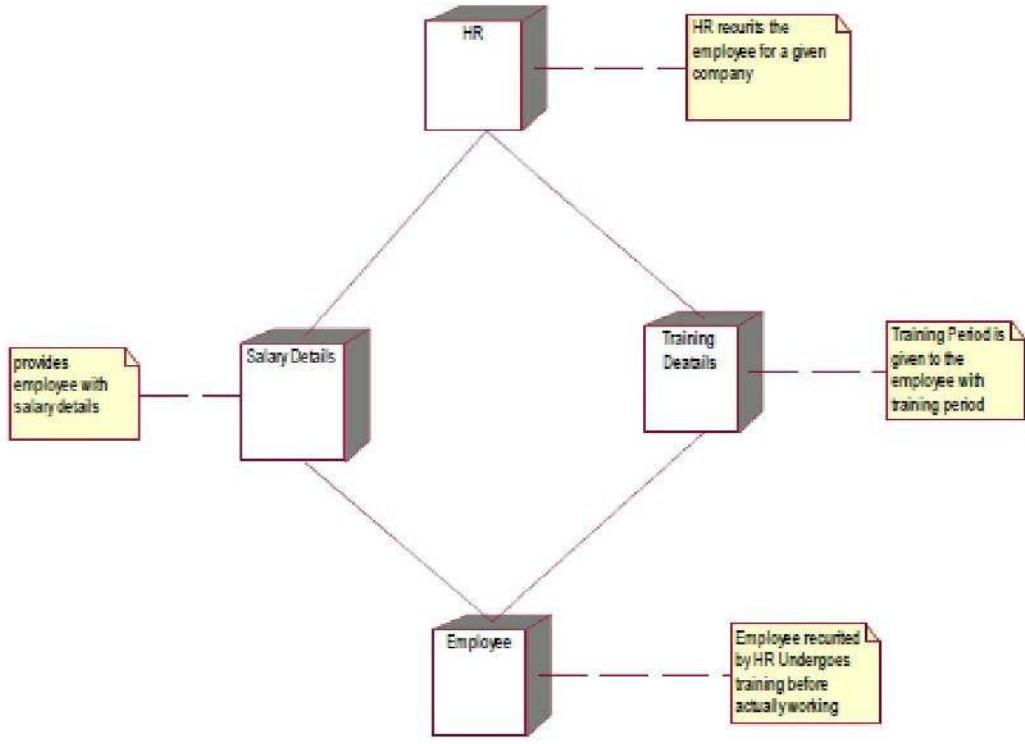
States of object are represented as rectangle with round corner, the transaction between the different states. A transition is a relationship between two state that indicates that when an event occur the object moves from the prior state to the subsequent.



UML DEPLOYMENT 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.

Component diagram are used in conjunction with deployment diagram to show how physical modules code are distributed on various hardware platform. The processor node in the system is student information system and the execution environment nodes or device nodes are student, staff and DBA.

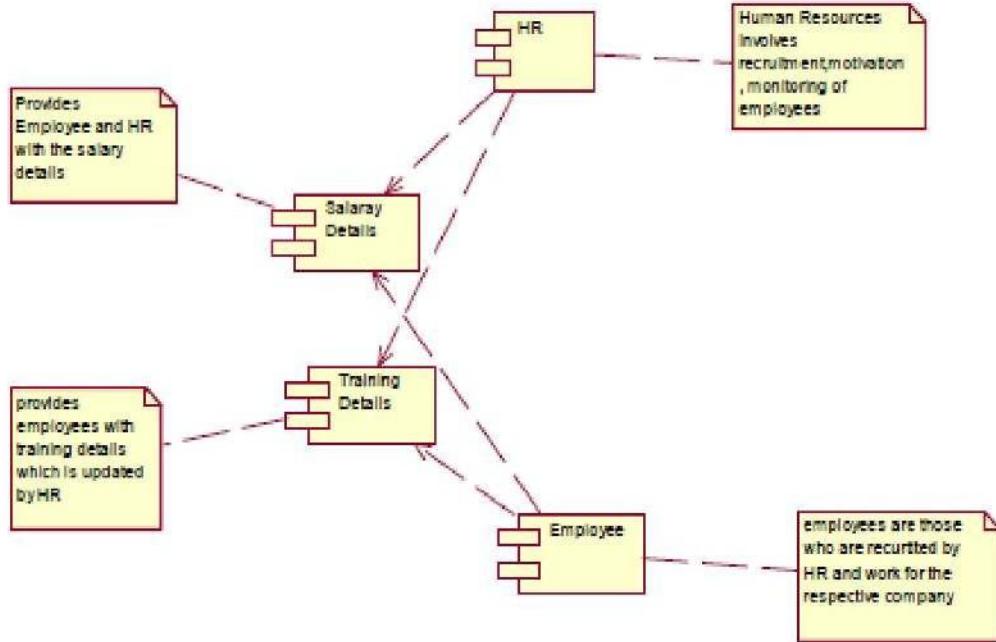


UML COMPONENT DIAGRAM:-

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
Component diagram carries the major living actors of the system.

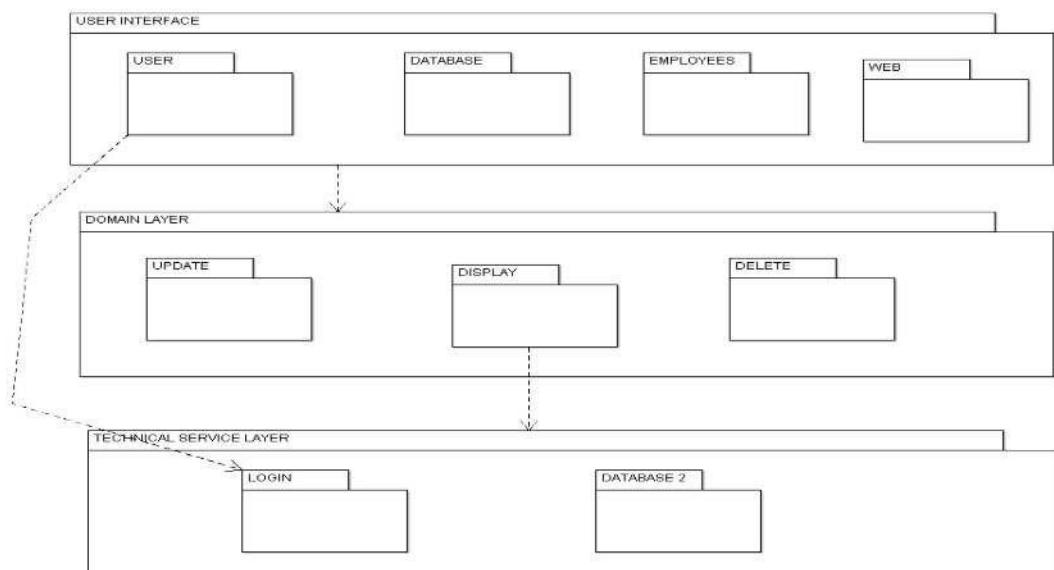
The component diagram main purpose is to show the structural relationship between components of the system. The main component of the system is student information system and the other components of the system are student, staff and DBA.



UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network



UML TECHNICAL SERVICE LAYER:-

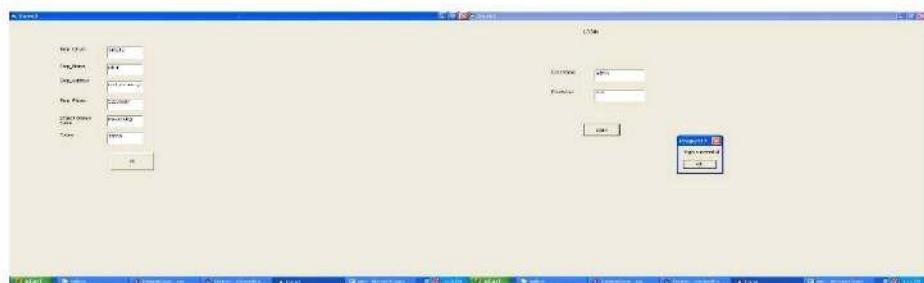
EMPLOYEE NAME	PASSWORD
Surya	54792
Saravana	12345
Seenu	67890

SAMPLE CODE:-

```
public class Central Management System {  
    public string employ name;  
    public int number;  
    public varchar Details;  
    public void leave taken() {  
    }  
    public void tax() {  
    }  
    public void loan() {  
    }  
    public void salary() {  
    }  
    public void centeralmanagementsystem() {  
    }  
}
```

```
public class HR1 extends CentralManagement  
System, CentralManagement {  
    public varchar checkdetails;  
    public Integer newattr;  
    public Vector myEmployee1;  
    public void lossofpay() {  
    }  
    public void tax() {  
    }  
    public void project() {  
    }  
}
```

USER INTERFACE LAYER:-



Result:-

Thus the Software Personal Management System has been done successfully by using Argo-UML.

EX.NO: 8	CREDIT CARD PROCESSING SYSTEM
----------	-------------------------------

AIM:

To design Credit Card Processing System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN:-

To simplify the process of applying Credit Card Processing, software has been created by designing through ARGO-UML tool.

Online payment system can left people who purchase something and make payments using their credit card their bank account through internet. The problem here is to build up a reliable, affordable, secure and scalable online transaction processing systems do that consumers and business merchant bank can allow the business merchant to accept credit cards over the internet.

PROBLEM STATEMENT:-

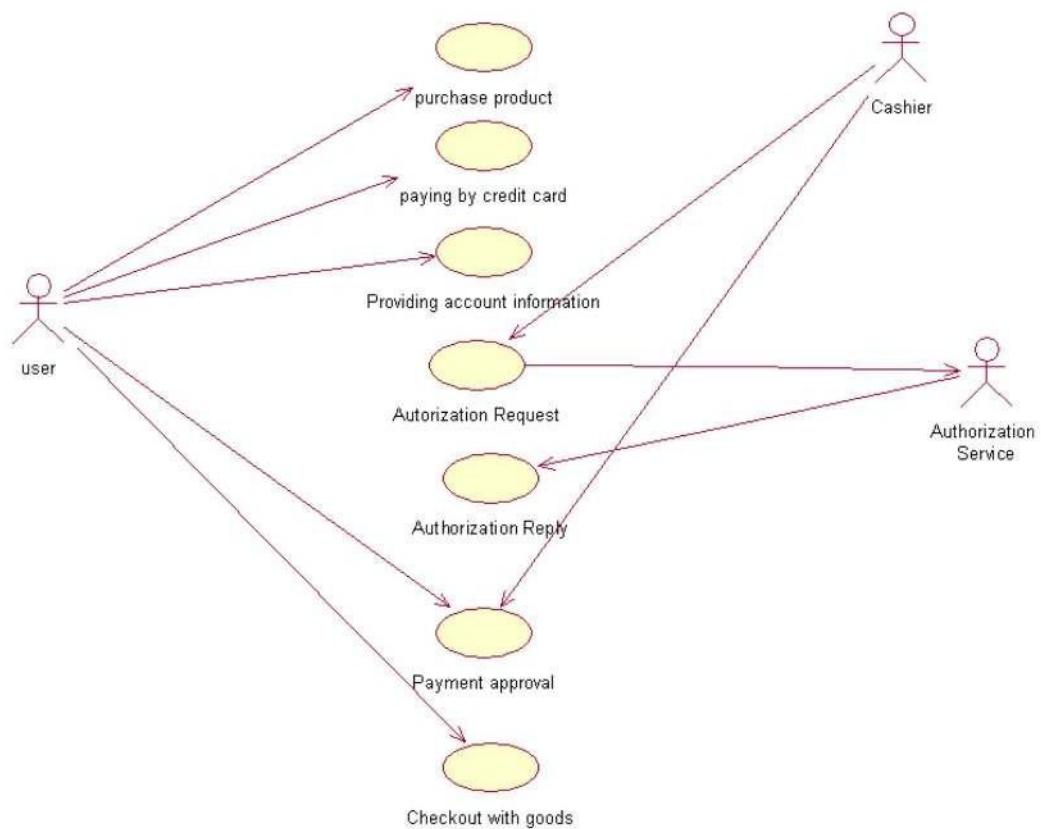
- When the consumer decides to buy something the merchant's commerce application prompts the consumer for credit card information usually along with other information such as a shipping address.
- The consumer enters payment information either into a form secured by secure socket layer (SSL) protocol or into an application such as internet explorer that is complaint with the secure electronic transaction specification. With the secure form the payment information is protected by SSL.
- Using the payment software incorporated in the web server the merchant sends the encrypted transaction to the acquiring
- Processor for authorization.
- The acquiring processor either authorizes a certain amount of money. An authorization reduces the available credit limit but does not actually put a charge on the customer's bill or move money to the merchant.
- If transaction is authorized a "capture" is next step.
- If "cancels" void is generated if consumer returns goods after the transaction has been captured credit given is cancelled.
- Final step is to "settle" the transaction between the merchant and acquiring processor.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of

interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.

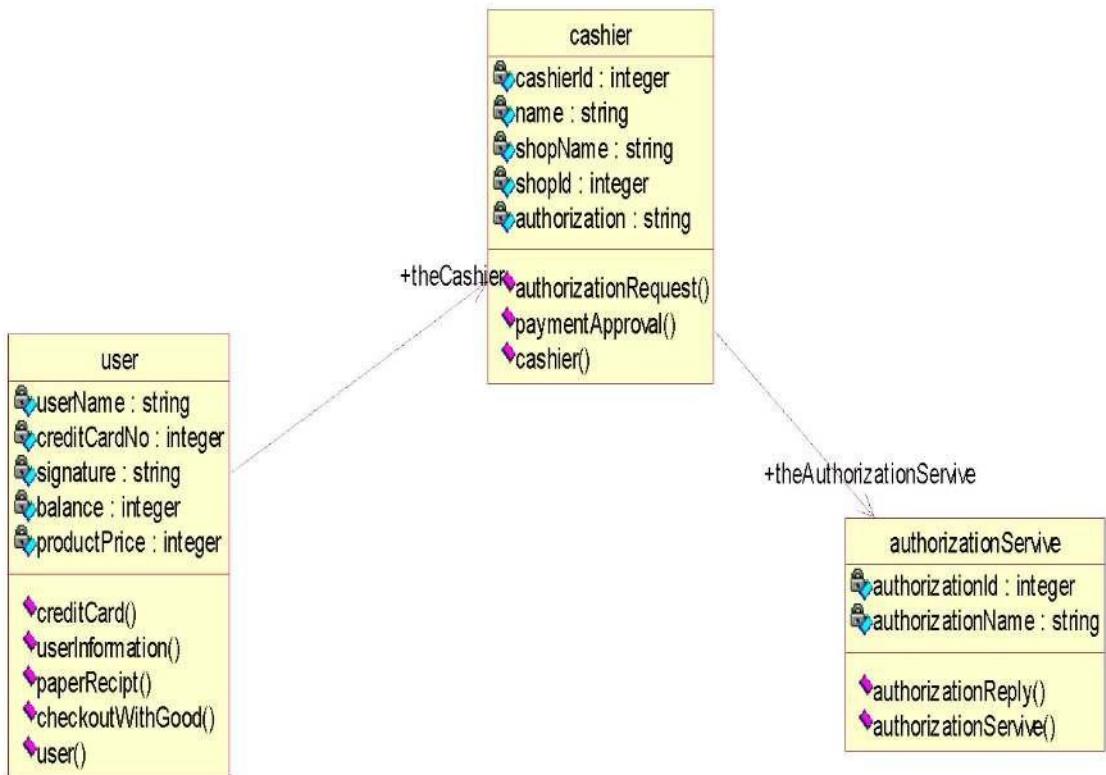


UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.

The class diagram is the graphical representation of all classes used in the system. The class diagram is drawn as rectangular box with three components or compartments like class name, attributes and operations. The student information system makes use of the following classes like student, staff, system, DBA and server.



DOCUMENTATION OF CLASS DIAGRAM:-

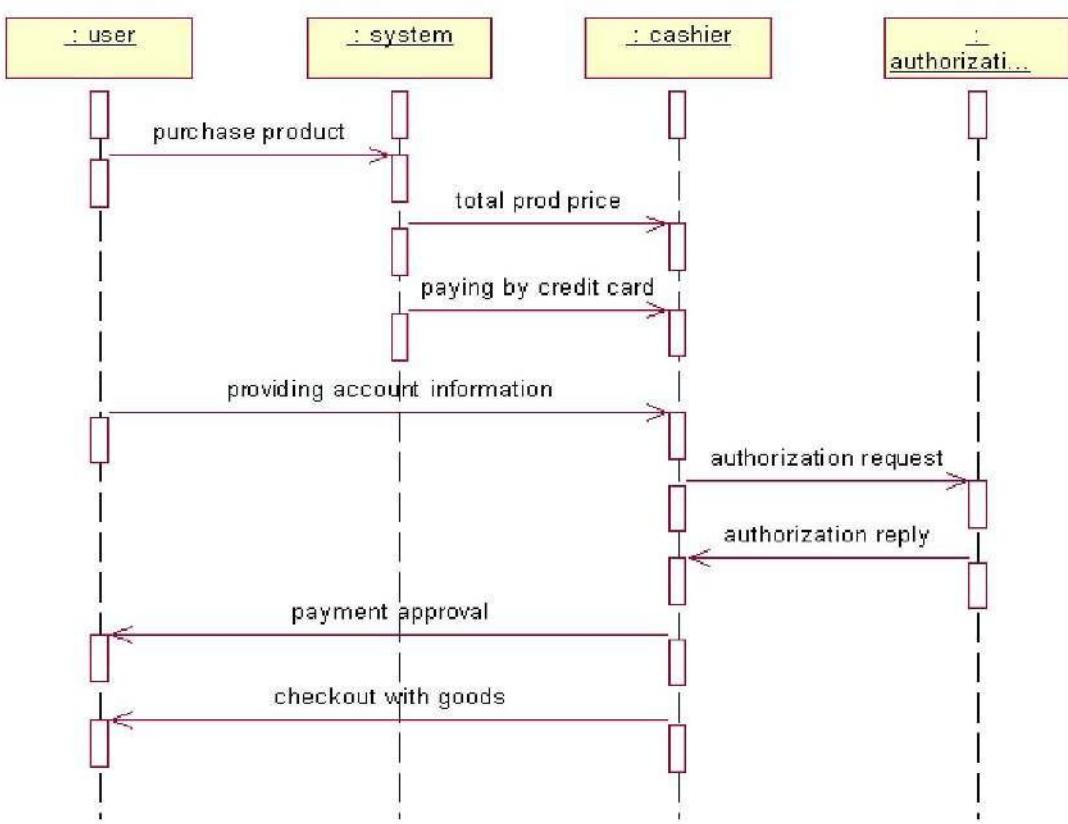
- Purchase item
- Bill issue
- Card verification
- transaction
- Bank database

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

- Purchase item
- Vendor
- Card reader
- Bank database
- Transaction

UML ACTIVITY DIAGRAM:-

Description:-

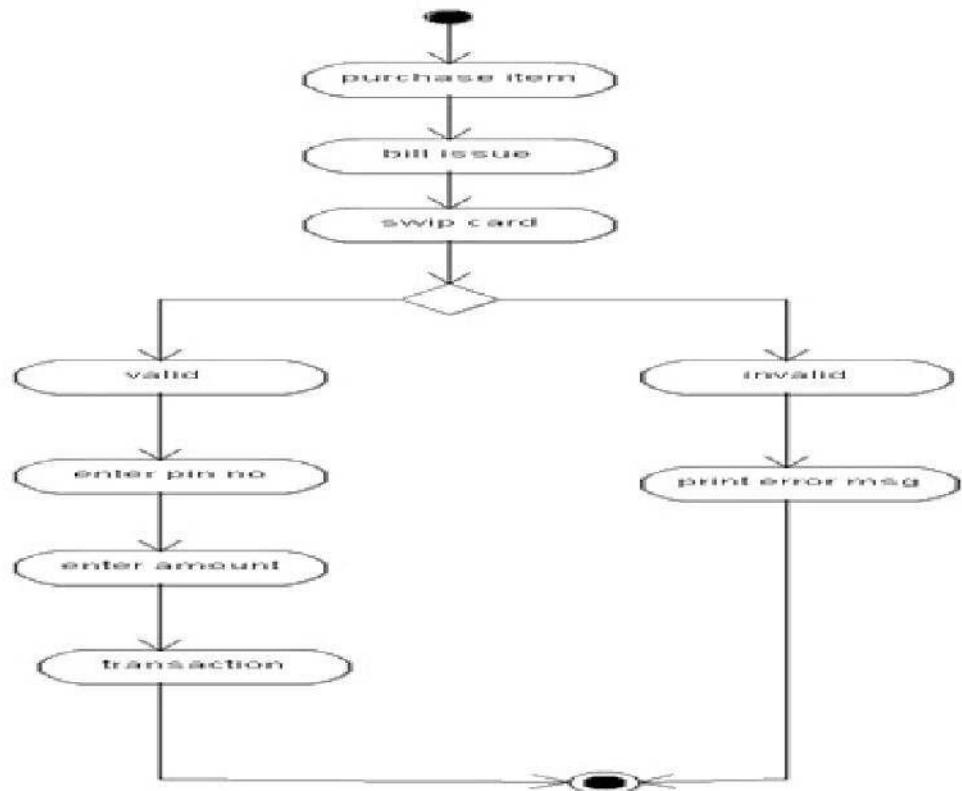
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

How activities are coordinator to provide a service.

The events needed to achieve some operation.

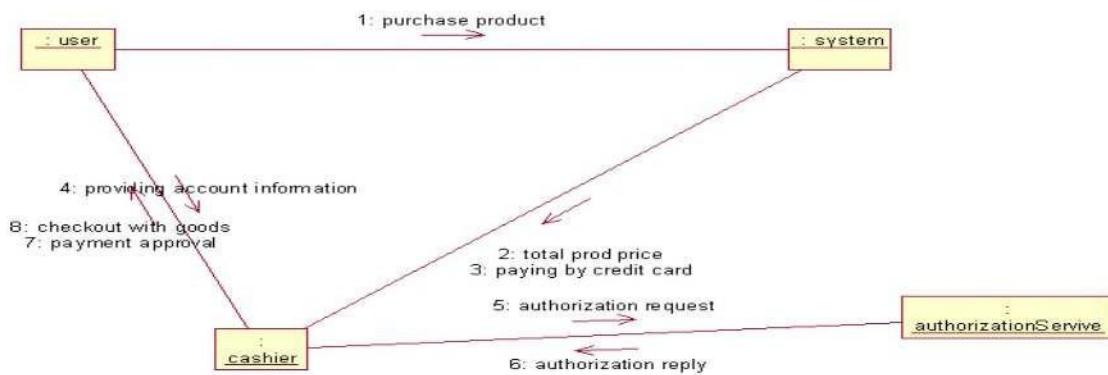
How events in a single use case relate to one another.



DOCUMENTATION OF ACTIVITY DIAGRAM:-

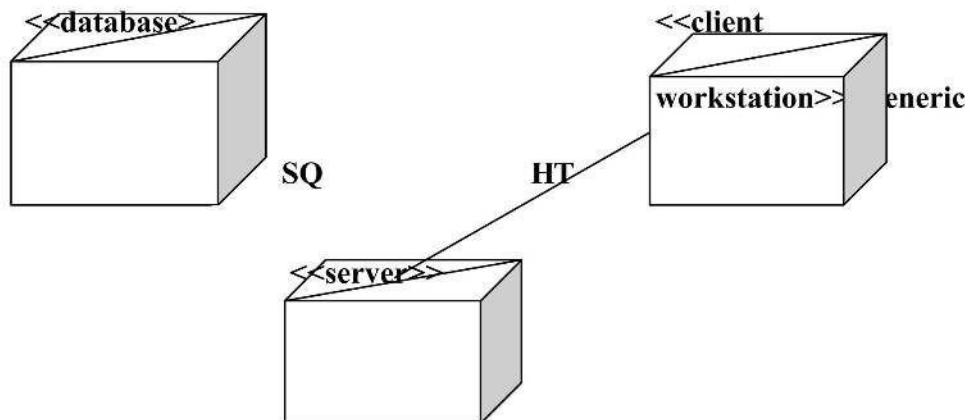
- Purchase item
- Bill issue
- Swipe the card
- Verification
- Transaction

UML COLLABORATION DIAGRAM:-



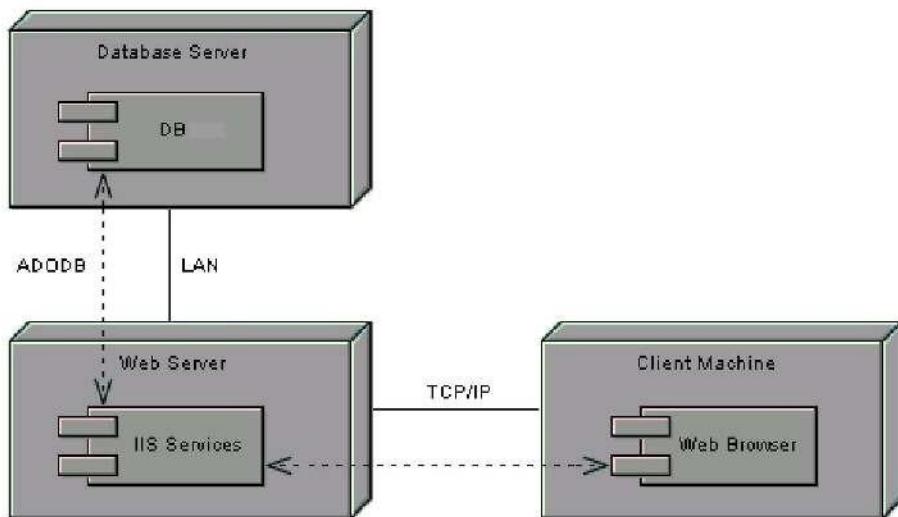
UML DEPLOYMENT DIAGRAM:-

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



UML COMPONENT DIAGRAM:-

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

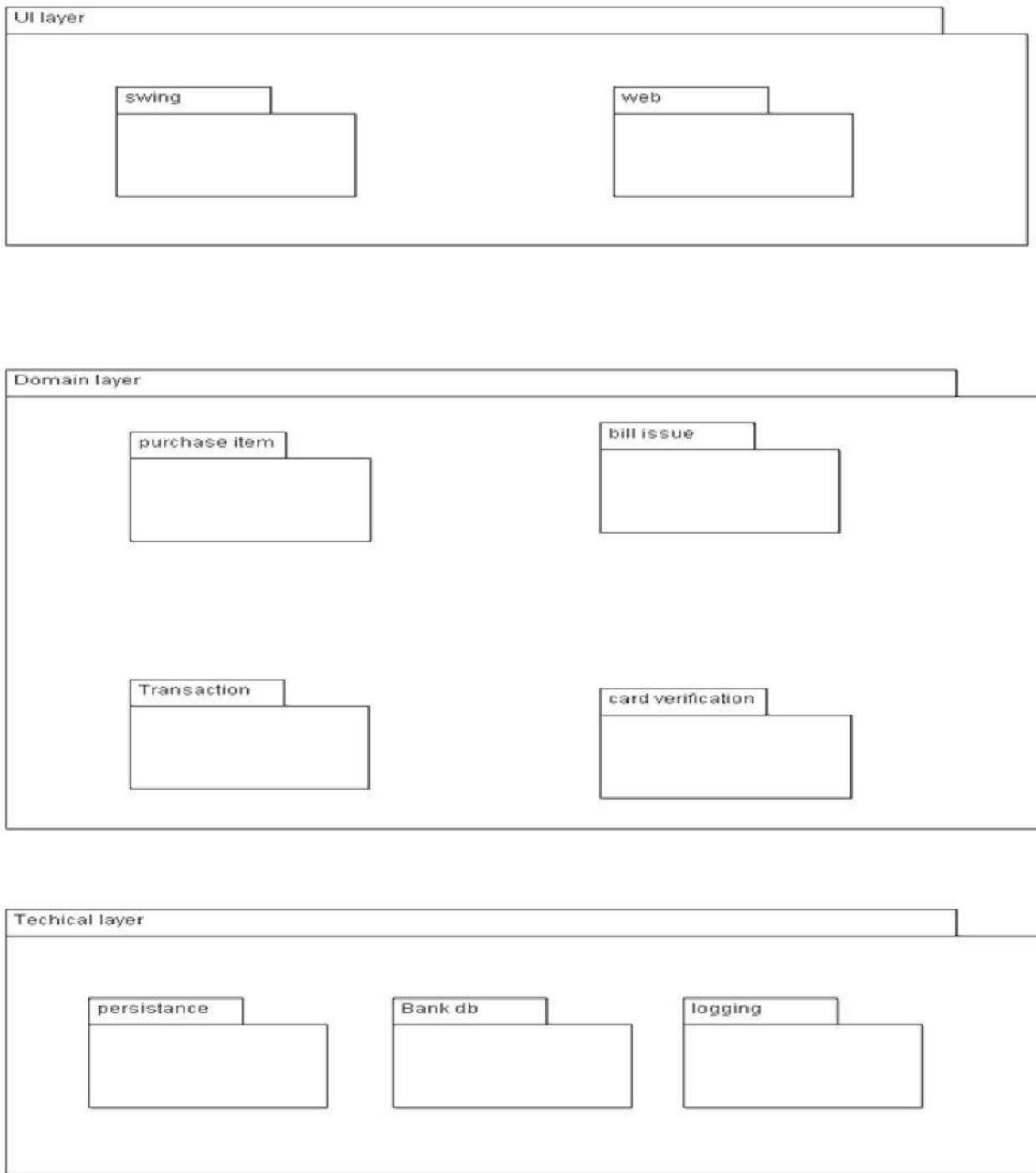


UML PACKAGE DIAGRAM:-

Description:

The **Logical architecture** is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these

elements are developed across different operating systems processes are across physical computers in a network.



UML TECHNICAL SERVICE LAYER:-

S.No	Acc_Holder	Acc_No	Card_No	Card_Name	Validity
1.	Yogesh	2075731634	4383462074632073	VISA	03/33
2.	Charumathi	4385349539	4593498734984959	RUPAY	05/40

3.	Karthikeyan	495734799	5893484474853988	MASTERCARD	03/50
----	-------------	-----------	------------------	------------	-------

SAMPLE CODE:-

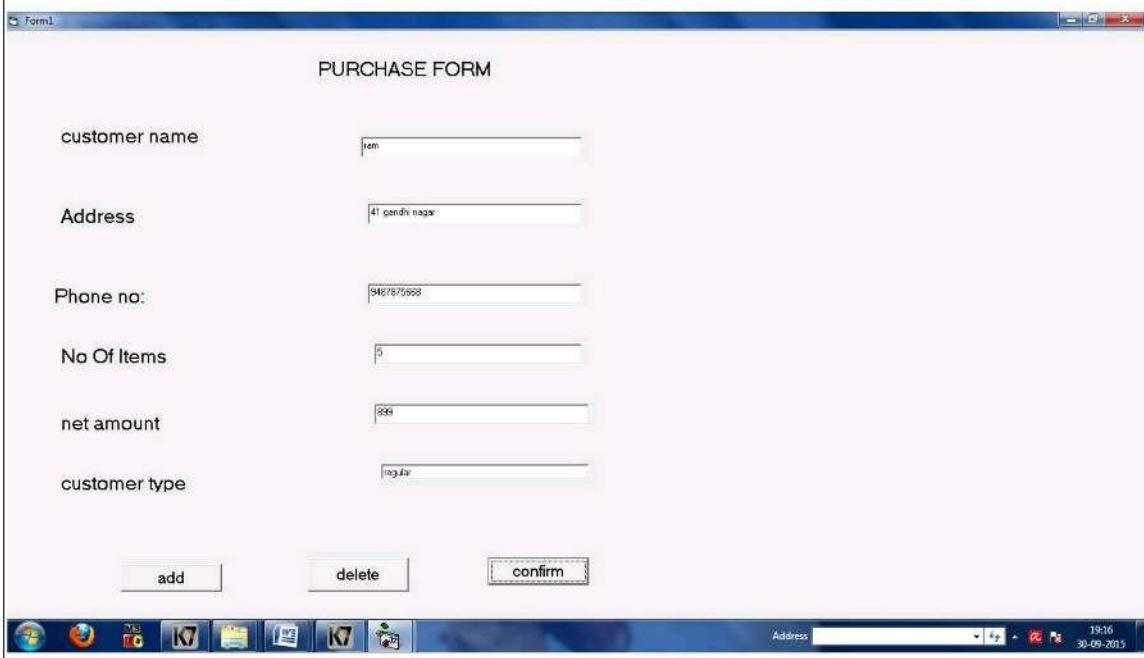
PURCHASE ITEM

```
import java.util.Vector;
public class Purchase_item {
/* {src_lang=Java}*/
public String cusname;
public String addr;
public Integer phno;
public Integer no_item;
public Integer net_total;
public String custype;
public Vector myBill_issue;
public void add() {
}
public void delete() {
}
public void confirm() {
}
}
```

BILL ISSUE

```
import java.util.Vector;
public class Bill_issue {
/* {src_lang=Java}*/
public Integer no_item;
public Integer price;
public Integer quantity;
public String shop_name;
public Vector myPurchase_item;
public Vector mycard_verification;
public void save() {
}
public void add() {
}
public void delete() {
}
public void discount() {
}
public void print() {
} }
```

USER INTERFACE LAYER:-



Form1

ACQUIRING BANK FORM

account number

vendor name

bank name

amount

Address 1943 30-09-2015

Result:-

Thus the Credit Card Processing System has been done successfully by using Argo-UML.

EX.NO:9	E-book Management System
---------	--------------------------

AIM:

To design E-book Management System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN

To simplify the process of applying E-Book Management, software has been created by designing through ARGO-UML tool.

PROBLEM STATEMENT:-

The software to be designed will control an eBook managing website which has eBooks in its database, which is available for registered users to download it by paying the necessary cost; the registered users can login and search for the required books and the users can update their details.

This can serve more than one customer at the same time, initially the customer enters his User ID and password and opens his account the User ID and password combination is checked in the database and only if correct, the account opens. After finishing the user can logout.

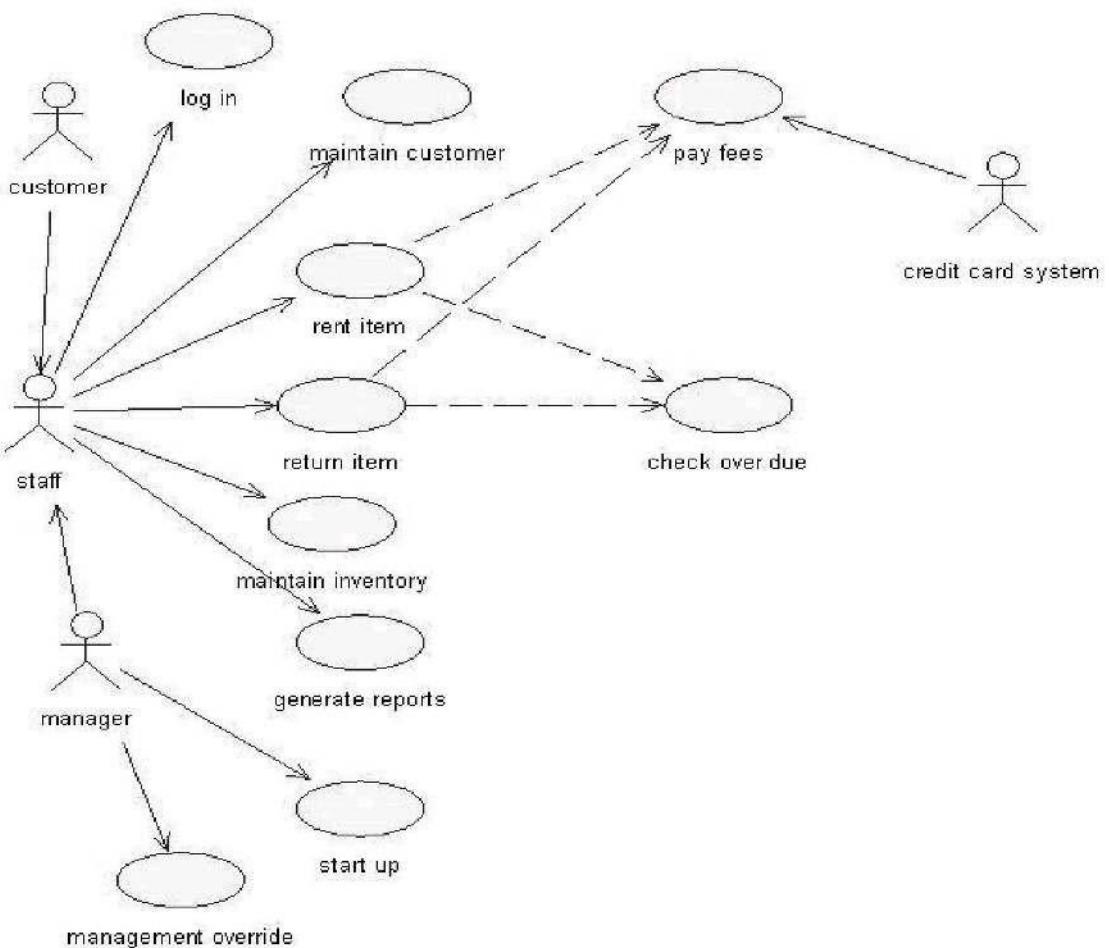
This software must provide the following:

1. A new customer must be able to register himself for using it using his email ID.
2. A registered user must be able to log-in into his account.
3. A user must be able to search for the book which he requires and must be able to get the book if required after paying the required cost.
4. He must be able to pay cost through credit card, debit card and internet banking.
5. He must be able to update his personal details.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

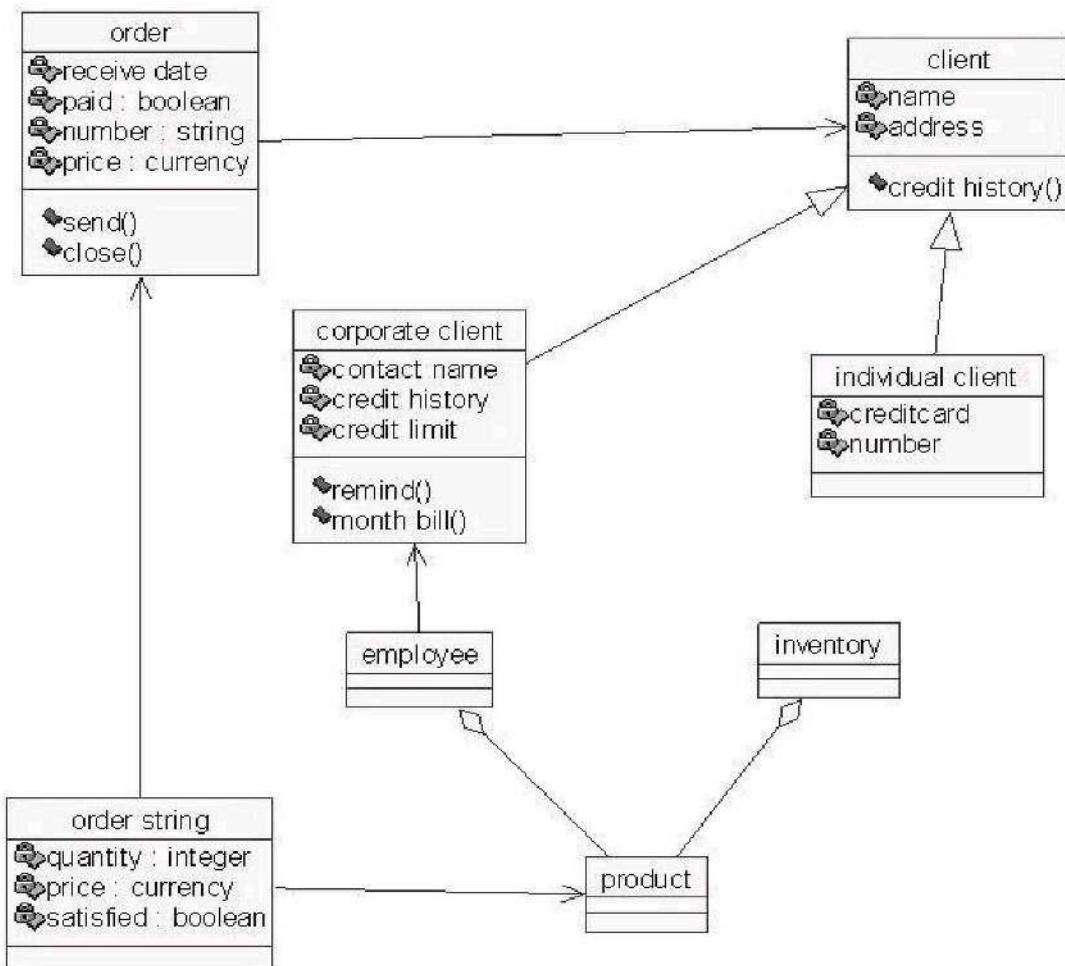
1. User access his account by providing the correct userid and password.
2. The user searches for the required book.
3. The user selects the book which he wants from the listed books.
4. The user confirms which redirects to payment page.
5. In payment page, the user enters the details of netbanking or his card.
6. The details are validated in the Bank database, if correct,
7. An OTP is generated and sent to the user's mobile.
8. If that is entered correctly, transaction is done and updated in the bank database.
9. The book is downloaded to the user and the database is updated.

The user can either log out or he can continue to download more books.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

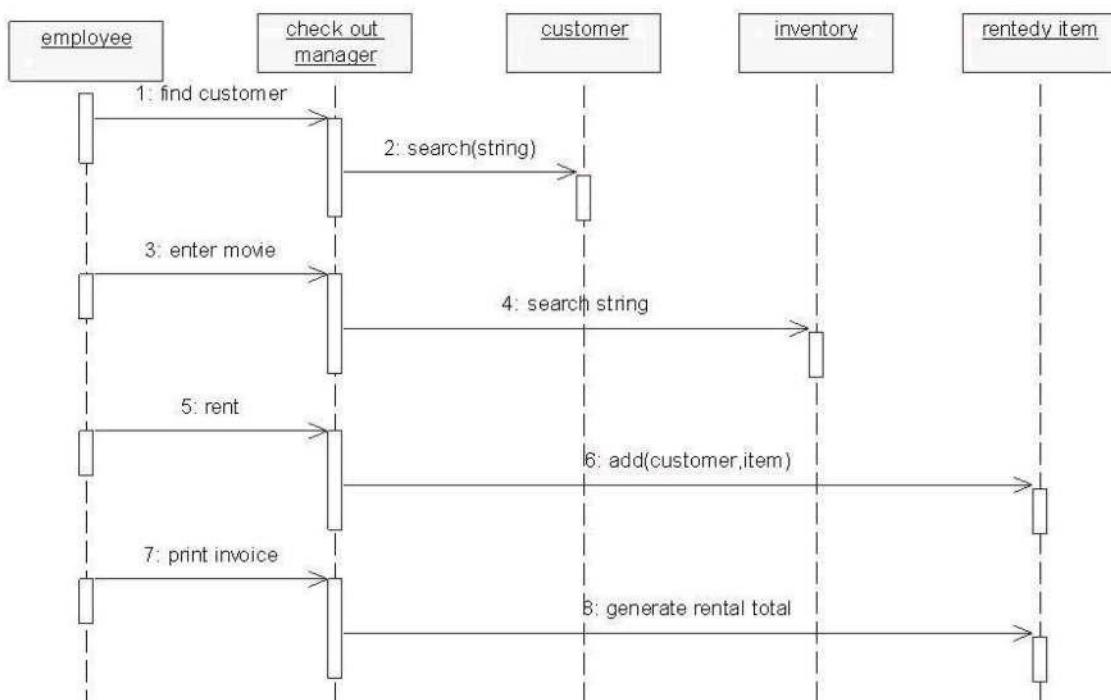
This diagram shows the classes associated with this system and the way how they are linked with each other. This diagram also shows the attributes and methods of the class. The first partition shows the name of the class, second shows the attributes and third shows the methods.

UML INTERACTION DIAGRAM:-

Description:-

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.**



DOCUMENTATION OF SEQUENCE DIAGRAM:-

This sequence diagram shows the sequence of operations for downloading a book. First the user opens his account searches and selects the required book, pays the required amount and then the user gets the book.

UML ACTIVITY DIAGRAM:-

Description:-

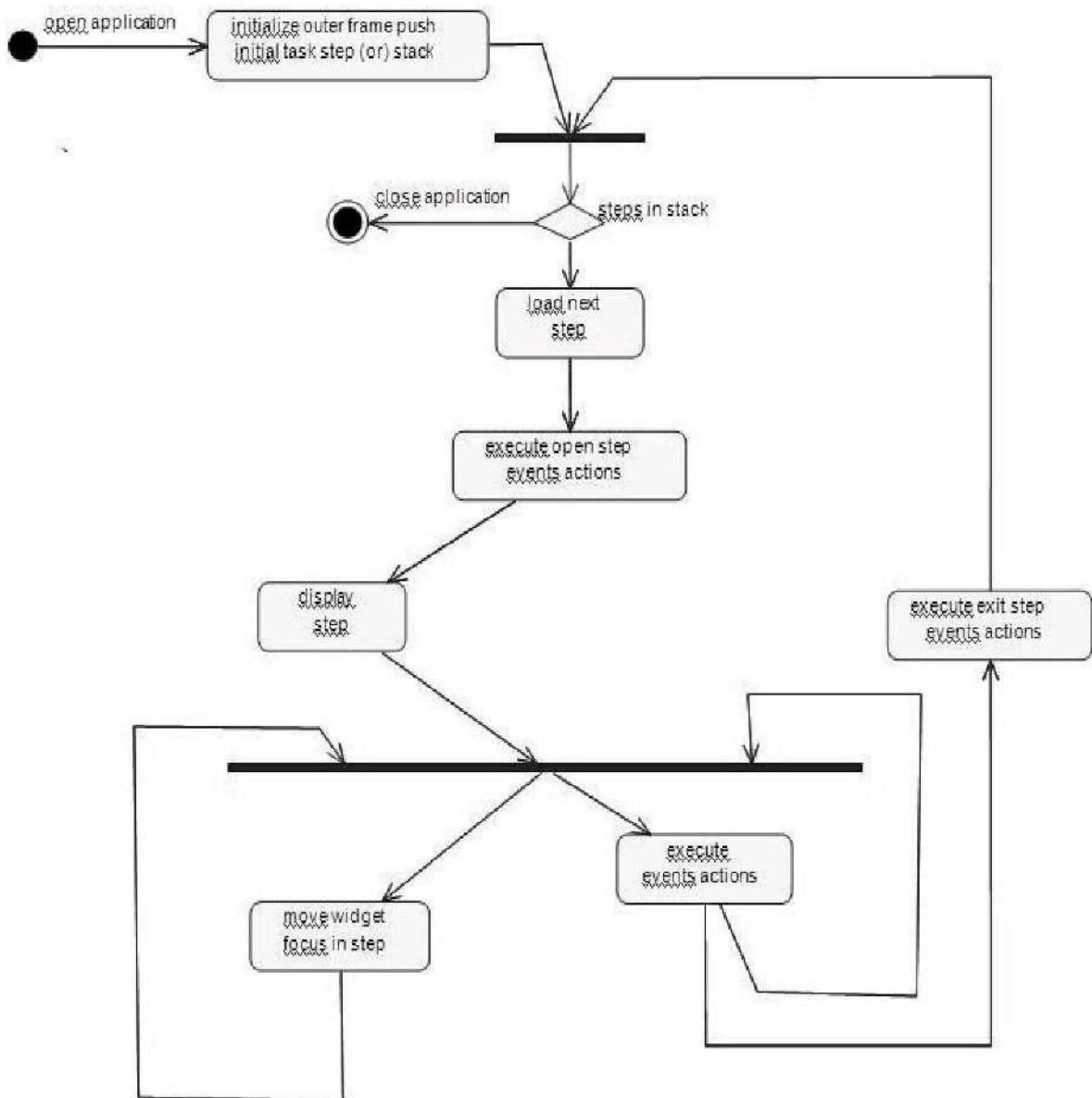
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

How activities are coordinator to provide a service.

The events needed to achieve some operation.

How events in a single use case relate to one another.

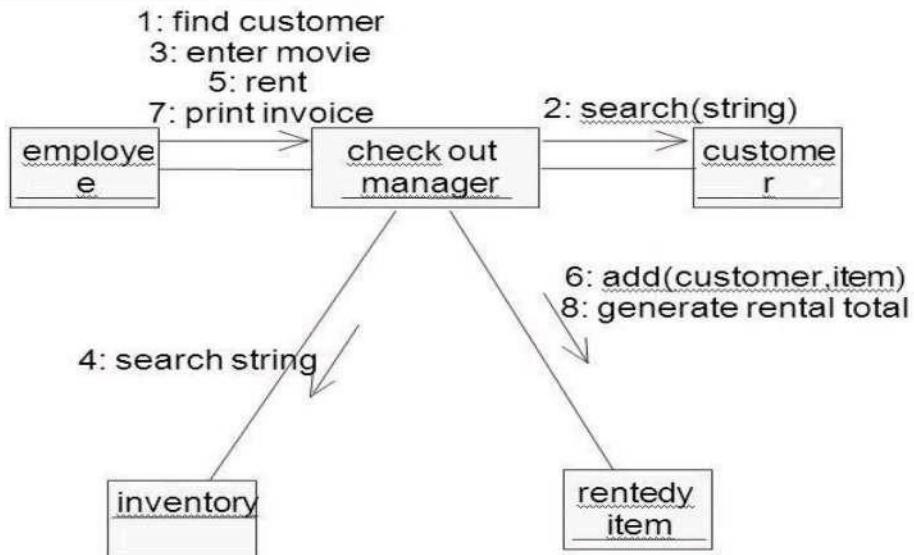


DOCUMENTATION OF ACTIVITY DIAGRAM:-

The above shown activity diagram shows the generalized view of the working of the system. This shows the various activities performed by the system.

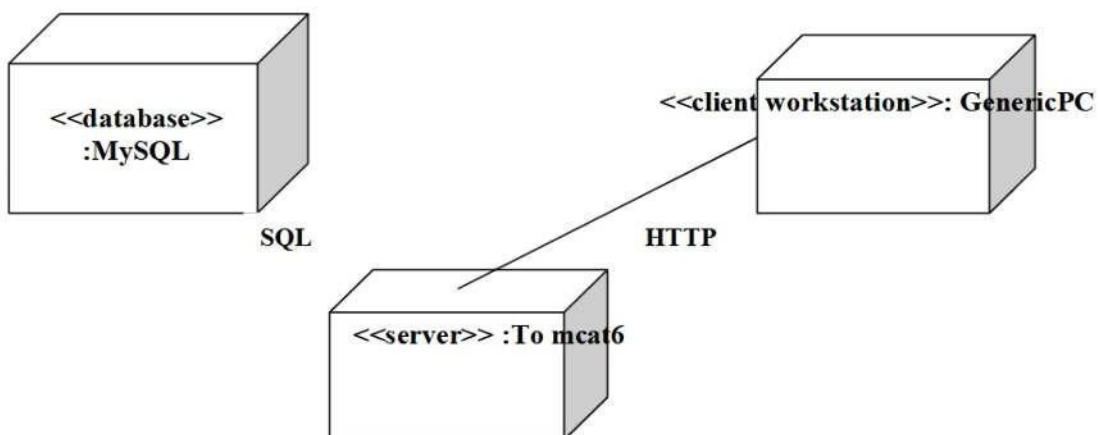
Activity diagram are graphical representation of workflows of stepwise activities and actions with support for choice, iteration and concurrency. Here in the activity diagram the user login to the system and perform some main activity which is the main key element to the system.

UML COLLABORATION DIAGRAM:-



UML DEPLOYMENT DIAGRAM:-

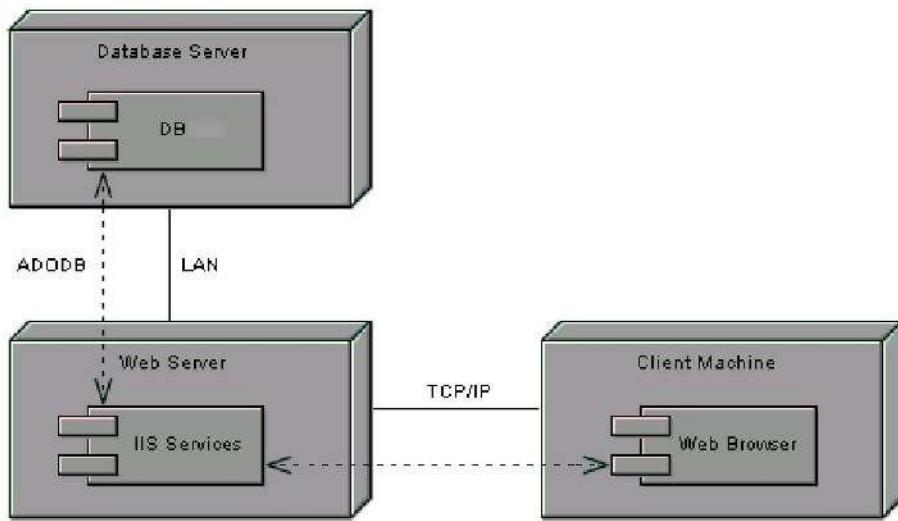
Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed.



UML COMPONENT DIAGRAM:-

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

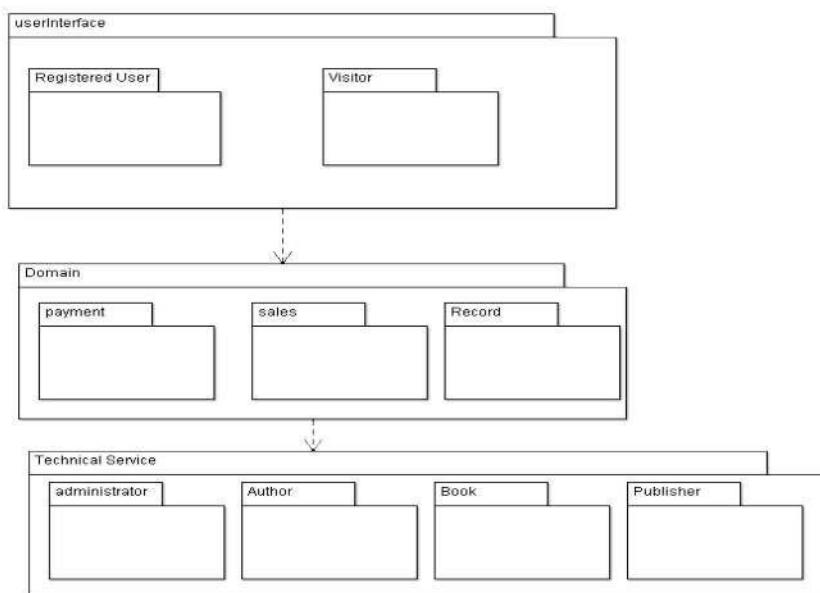
Component diagram shows the dependencies and interactions between software components. Component diagram carries the most important living actors of the system i.e, user, librarian and DBA



UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



UML TECHNICAL SERVICE LAYER:-

S. No	Author Id	Name	Address	Contact Number	Total Books
1.	300001	Mories Mano	Punjab	9832872738	10
2.	300002	AK-Ray	Delhi	9374389392	15
3.	300003	Singaravelu	Chennai	9374239982	30

SAMPLE CODE:-

Administrator:

```
public class Administrator {
    /* {src_lang=Java}*/
    public String name;
    public void add() {
    }
    public void delete() {
    }
    public void modify() {
    }
}
```

Author:

```
importjava.util.Vector;
public class Author {
    /* {src_lang=Java}*/
    public Integer id;
    public String name;
    public String address;
    public Integer contact number;
    public Integer list of books;
    public Vector myAdministrator;
    public void add() {
    }
    public void remove() {
    }
    public void update() {
    }
}
```

USER INTERFACE LAYER:-

Name: <input type="text"/>	Card Type: <input type="text"/>
email id: <input type="text"/> @ <input type="text"/>	Card Number: <input type="text"/>
user id: <input type="text"/>	Pin: <input type="text"/>
password: <input type="text"/>	<input type="button" value="confirm"/> <input type="button" value="cancel"/>
<input type="button" value="submit"/> <input type="button" value="cancel"/>	

Registration Form

Payment

Result:-

Thus the E-Book Management System has been done successfully by using Argo-UML.

EX.NO:10	RECRUITMENT SYSTEM
----------	--------------------

AIM:

To design Recruitment System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN :-

To simplify the process of applying, software has been created by designing through ARGO-UML tool.

The Online Recruitment System is an online website in which applicant can register themselves and then attend the exam. Examination will be conducted at some venue. The details of the examination, venue & Date of the examination will be made available to them through the website. Based on the outcome of the exam the applicant will be short listed and the best applicant is selected for the job.

PROBLEM STATEMENT:-

The process of applicant is login to the recruitment system and register for the job through online. The resume is processed by the company and the required applicant is called for the test. On the basis of the test marks, they are called for next level of interview. Finally the best applicant is selected for the job.

This process of online recruitment system are described sequentially through following steps,

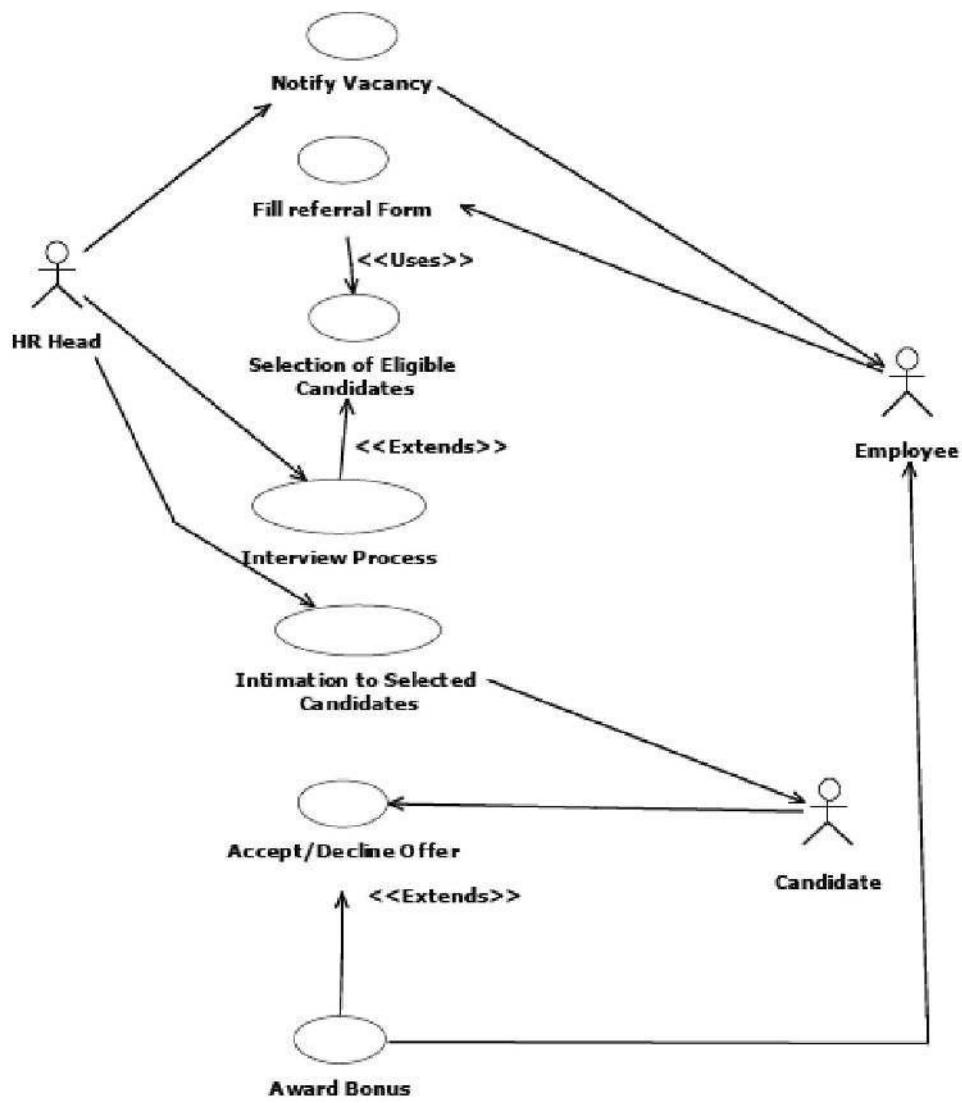
- The applicant login to the online recruitment system.
- They register to the company for the job.
- They appear for examination.
- Based on the outcome of the exam, the best applicant is selected.
- The recruiter informs the applicant about their selection

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.

Use case is a list of actions or events. Steps typically defining the interactions between a role and a system to achieve a goal. The use case diagram consists of various functionality performed by actors like user, librarian, system and DBA



DOCUMENTATION OF USECASE DIAGRAM:-

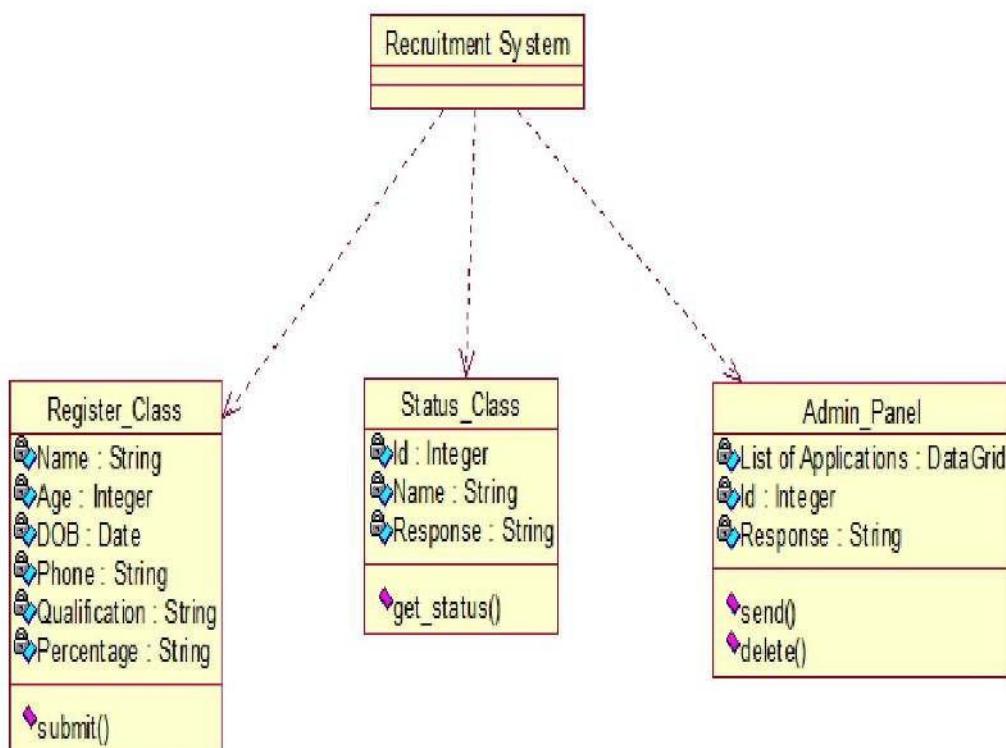
- [1] Candidate should sign up by giving username and password.
- [2] Admin sends the reply to the candidate with register numbers.
- [3] Enter into the software to attend the aptitude test. It will show the home page.
- [4] Candidate should attend the test which is conducted online
- [5] Admin should correct the answer sheet which has been sent by candidate.
- [6] Admin should select the candidate and update the admin tools and company details to respective candidates.
- [7] Organization should send the conformation letter to the selected candidate.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.

A class diagram in the unified modeling language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, and the relationships between the classes. It is represented using a rectangle with three compartments. Top compartment have the class name, middle compartment the attributes and the bottom compartment with operations.



DOCUMENTATION OF CLASS DIAGRAM:-

This class diagram has three classes' applicant, recruiter and database.

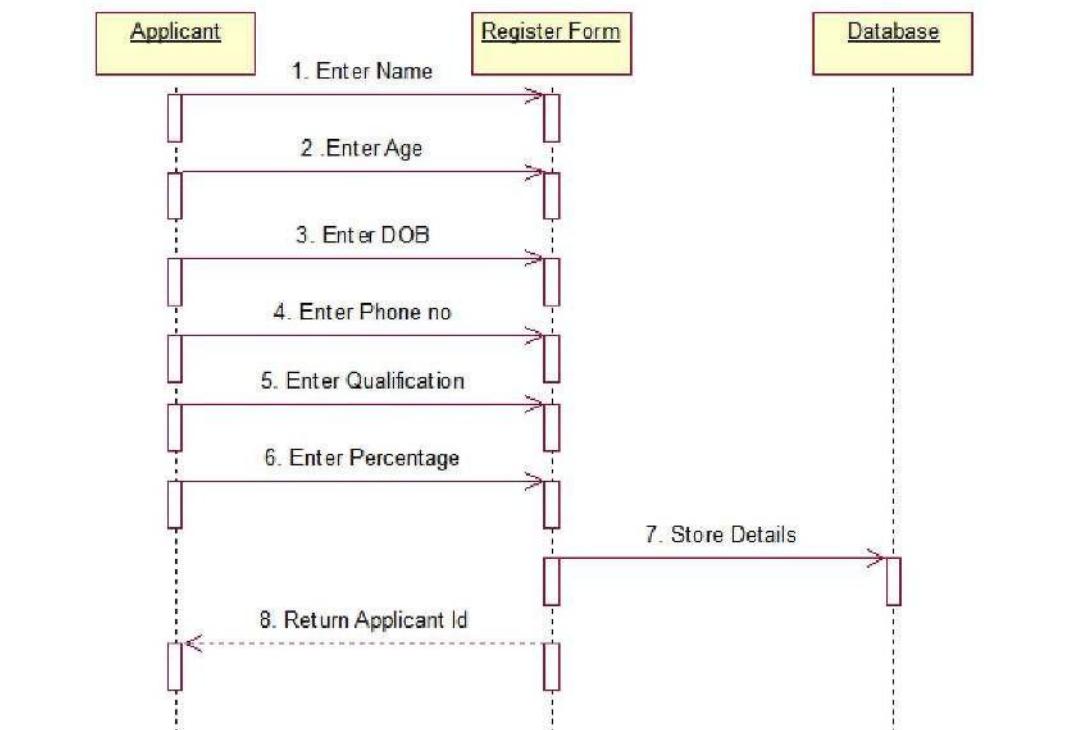
- **Applicant** – is the class name. Its attributes are username, password, name, phone no and address. The operations performed in the applicant class are login, register and giving applicant details.
- **Recruiter** – is the class name. Its attributes are name, designation, phone no, marks in apps and marks in technical. The operations performed are selecting applicants based on apps and technical.
- **Database** – is the class name. The operations performed are storing applicant details, verifying login and storing selected applicant details.

UML INTERACTION DIAGRAM:-

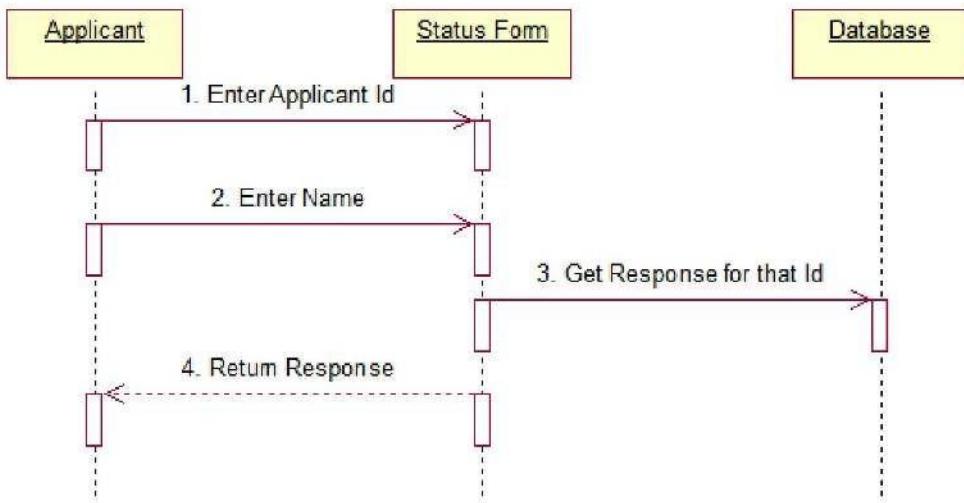
Description:-

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.**



SEQUENCE DIAGRAM FOR Register



SEQUENCE DIAGRAM FOR STATUS

DOCUMENTATION OF SEQUENCE DIAGRAM:-

REGISTER

This sequence diagram describes the sequence of steps to show

- The applicant login in to the recruitment system and register for job.
- The verification done in the database and recruiter
- The interview details are send to the applicant by recruiter.

SELECTING APPLICANT

This sequence diagram shows steps to show

- The applicant attend aptitude test and they are short listed based on evaluation
- The applicant appear for technical round

UML ACTIVITY DIAGRAM:-

Description:-

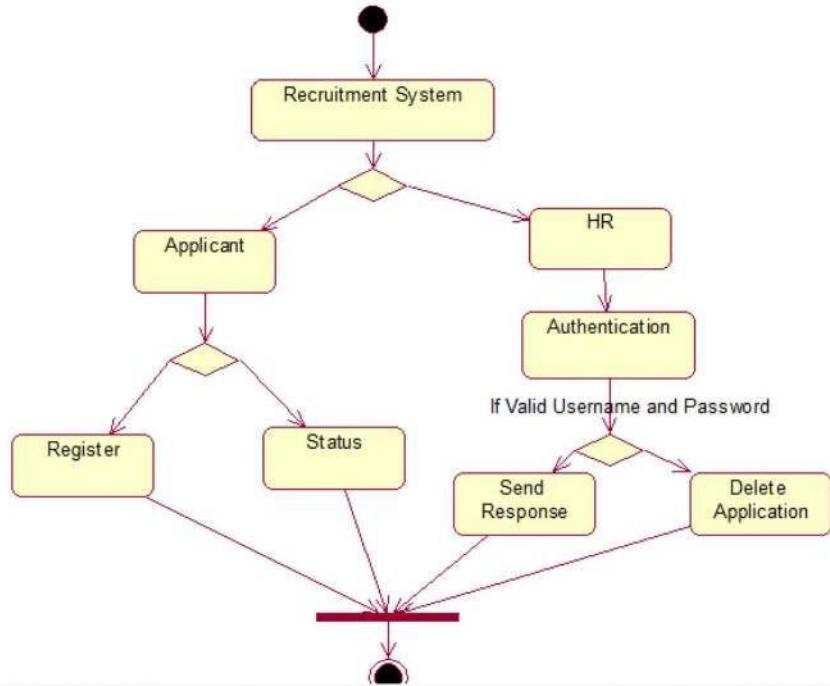
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

How activities are coordinator to provide a service.

The events needed to achieve some operation.

How events in a single use case relate to one another.



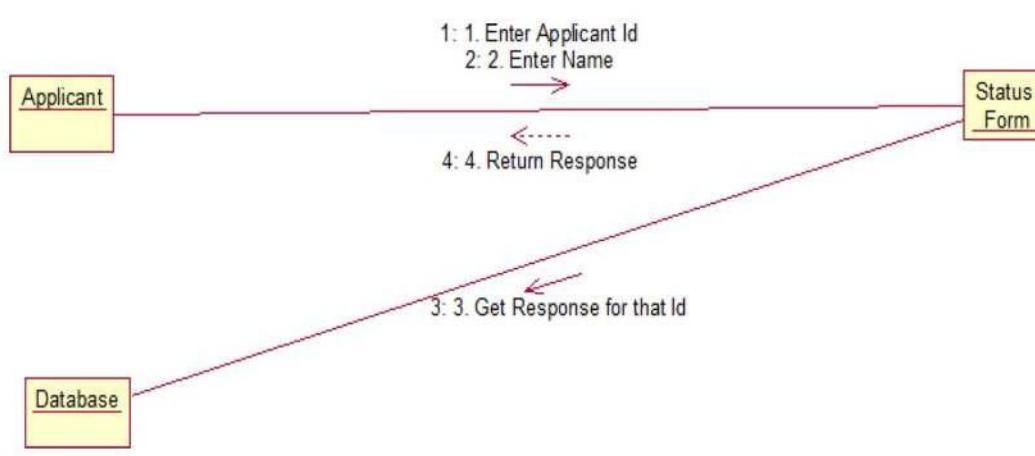
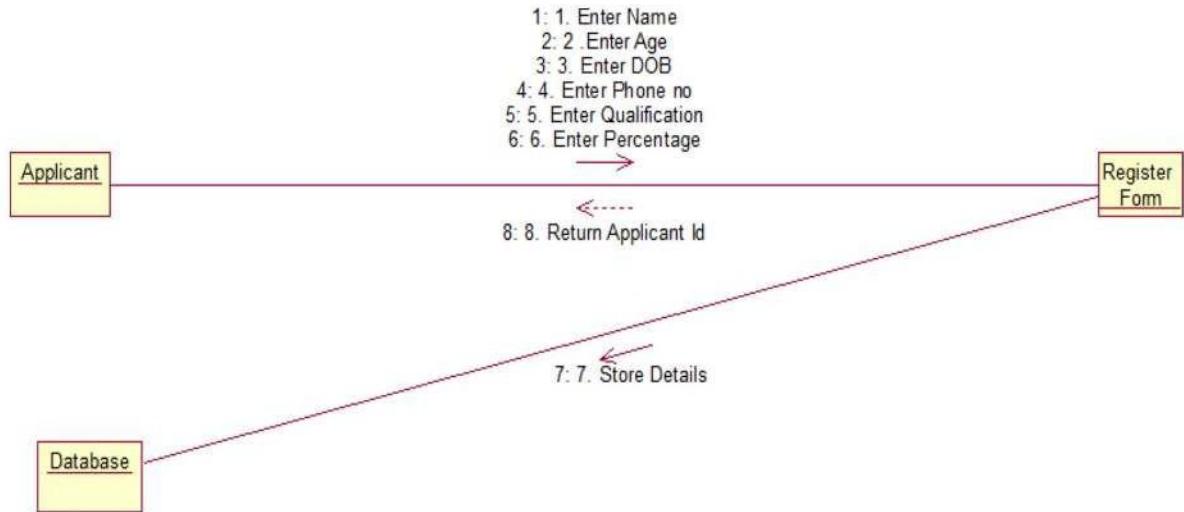
DOCUMENTATION OF ACTIVITY DIAGRAM:-

This state diagram describes the behavior of the system.

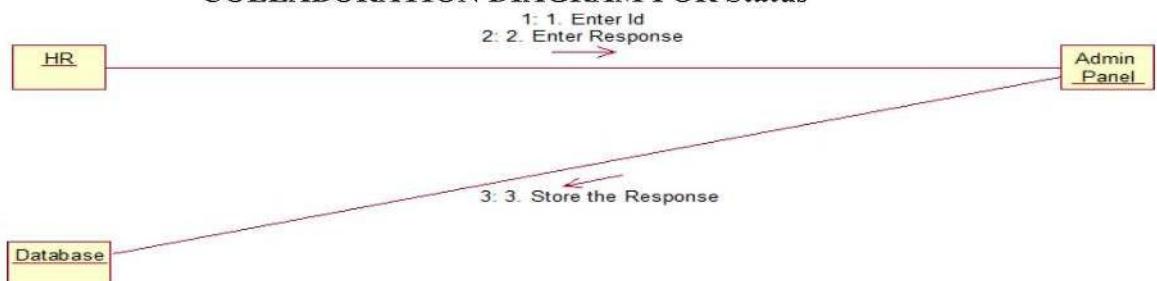
- First state is login where the applicant login to the recruitment system.
- The next state is register where the applicant register for job.
- Then verify the applicant details and sent interview details.
- The applicant appears for test.
- Update database with details of selected applicant.

UML COLLABORATION 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.



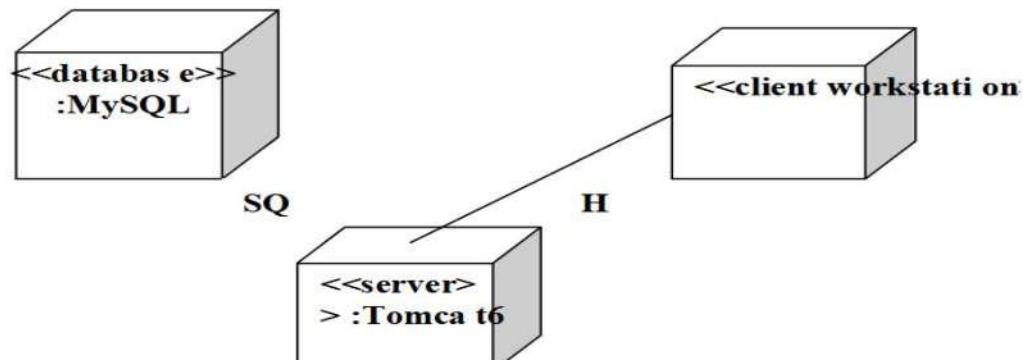
COLLABORATION DIAGRAM FOR Status



COLLABORATION DIAGRAM FOR Admin

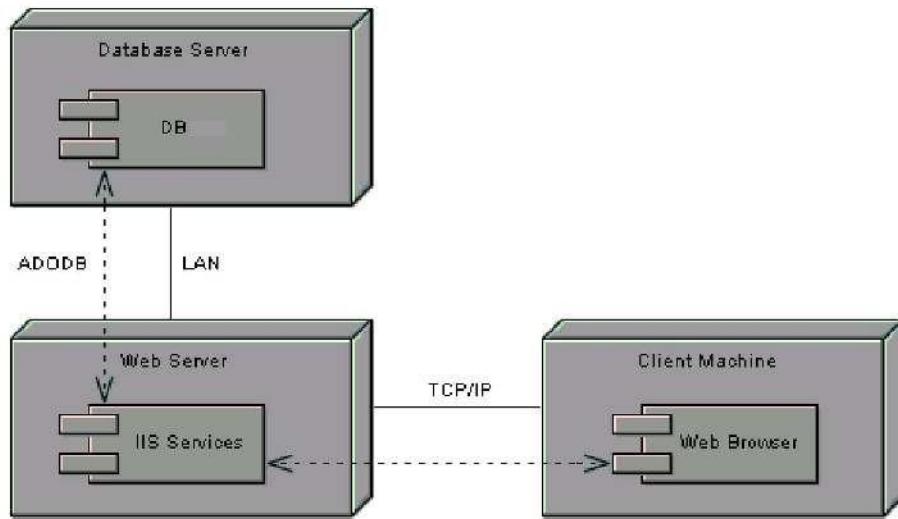
UML DEPLOYMENT DIAGRAM:-

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



UML COMPONENT DIAGRAM:-

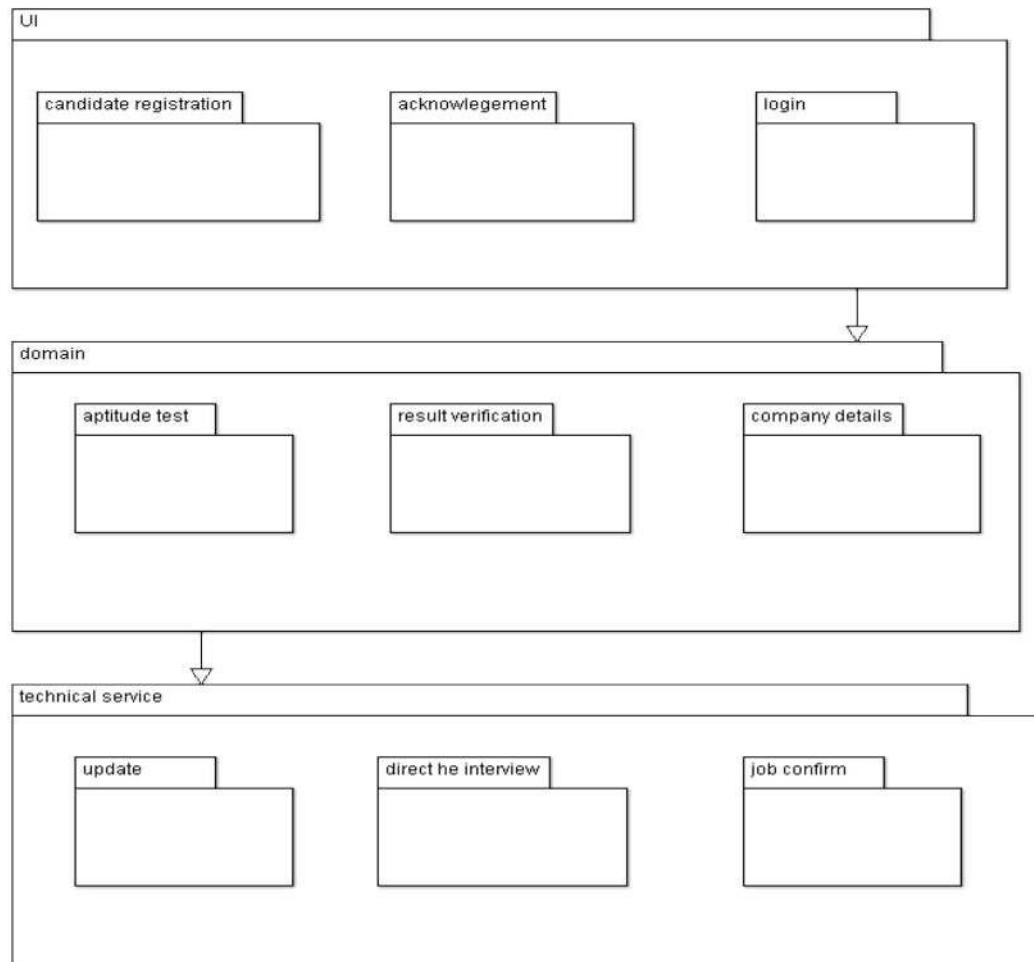
Component diagrams are used to visualize the organization and relationships among components in a system.



UML PACKAGE DIAGRAM:-

Description:

The **Logical architecture** is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



UML TECHNICAL SERVICE LAYER:-

S.No	Name	Tech_Marks
1.	Maha	90
2.	Sowmi	92
3.	Sharmi	88
4.	Divya	70
5.	Tejesh	70
6.	Barath	70
7.	Harish	70

SAMPLE CODE:-

Register Class

```

import java.util.Vector;
public class Register class {
    public String name;
    public Int age;
}

```

```

public varchar DOB;
public int Phone;
public string Qualification;
public varchar Percentage;
public Vector myRecruitment System;
public void submit() {
    }
}

```

USER INTERFACE LAYER:-



Result:-

Thus the Recruitment System has been done successfully by using Argo-UML.

EX.NO:11	FOREIGN TRADING SYSTEM
----------	------------------------

AIM:

To design Foreign Trading System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN

To simplify the process of applying, software has been created by designing through ARGO-UML tool.

The main activity of Foreign Trading System (FTS) is import and export procedure.

The online FTS is almost entirely a “spot” market. It means that trading is made immediately.

It also involves documentation, procedures, rules and regulations importing and exporting countries. Foreign trading spot transaction is made within two working days.

PROBLEM STATEMENT:-

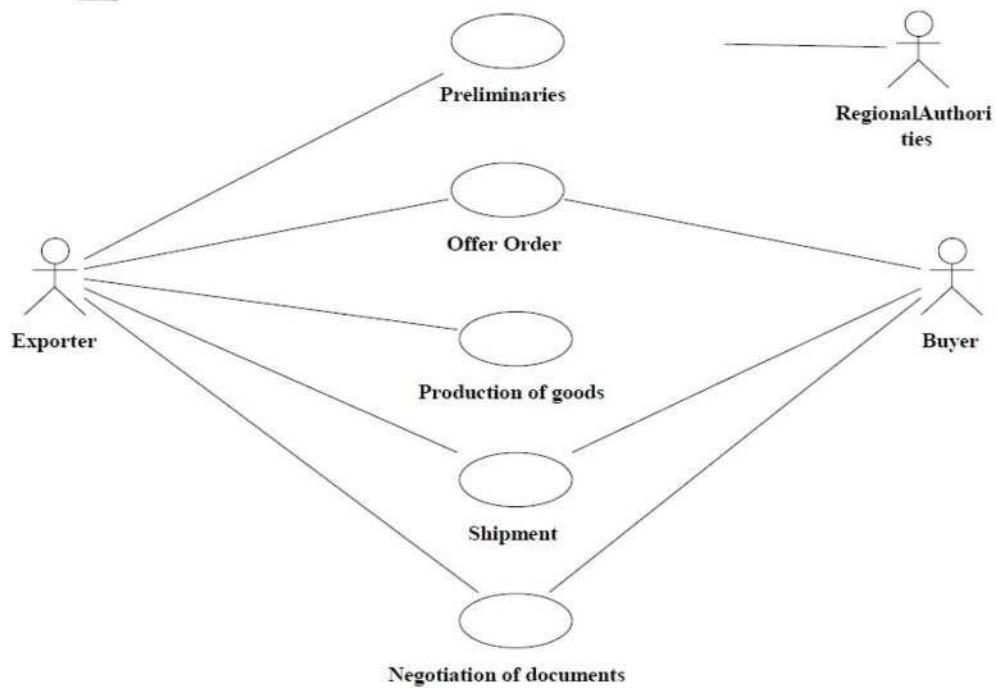
The steps involved in FTS are:

- The process of FTS begins with getting the username and password from the trader by the administrator.
- The administrator gives the authority to the trader to check the details of commodities.
- If the trader is satisfied with the commodities places his order to the administrator.
- The administrator checks for the availability and update the details in the database.
- The traders pass the amount to the administrator.
- The administrator will provide the bill after receiving the amount.
- The shipment people deliver the commodities to the trader.
- The system provides an interface where the buyer can fill in their personnel details and they can also scan and upload necessary documents. By using this shipment of goods can be done easily. Also application can be processed in speedy manner.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

The actors in this use case diagram are

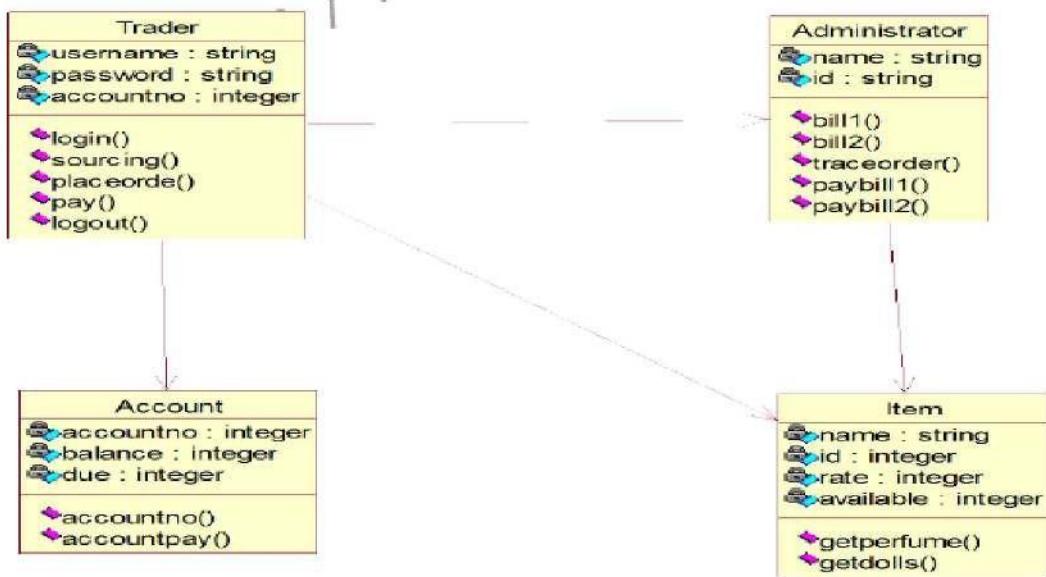
- Trader
- Buyer
- Regional authorities
- Bank
- Shipment people

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

This class diagram consists of nine class.

- Preliminaries
- Regional authorities
- Buyer
- Trader
- Offer order
- Invoice
- Production of goods
- Bank
- Shipment

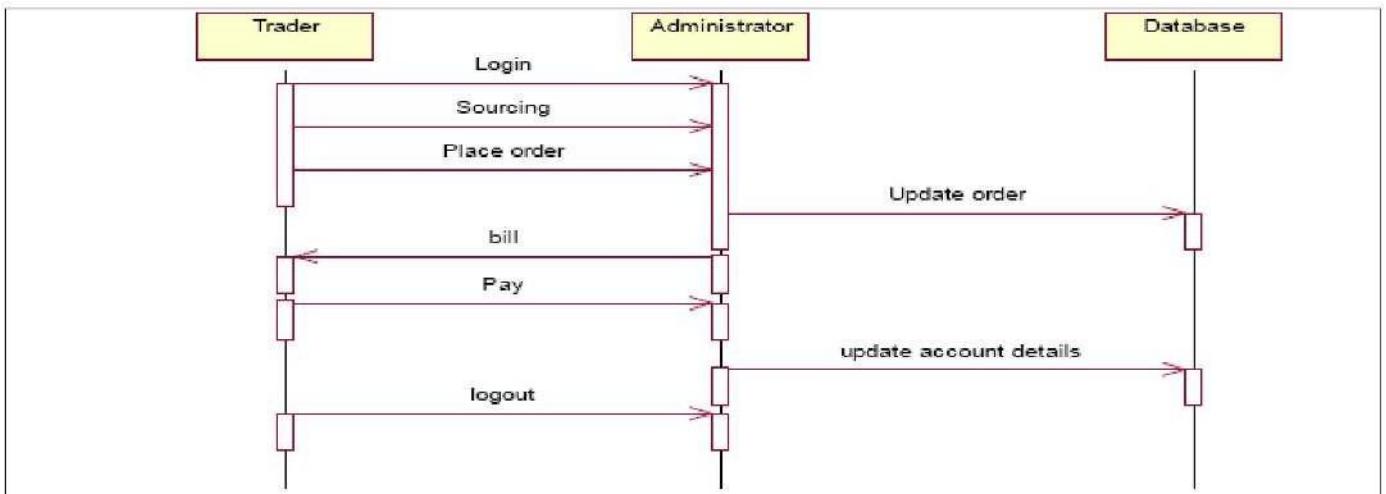
UML INTERACTION DIAGRAM:-

Description:-

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.

It is also represent in order by which they occur and have the object in the system send message to one another. Here the sequence starts with interaction between user and the system followed by database. Once the book have been selected the next half of sequence starts between librarian and user followed by database.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

The sequence diagram represents:

- ✓ The trader sends the details to the regional authorities and the regional authority gives approval .
- ✓ The trader offer order to the buyer and the buyer places his order.
- ✓ Then the trader generate the invoice and the buyer accepts it .
- ✓ Goods are produced and its send to the buyer through the shipment people .
- ✓ The payment is done via bank.
- ✓ The notification is send to the trader by bank about the update of payment.
- ✓ At the end the trader submits the document to the buyer.

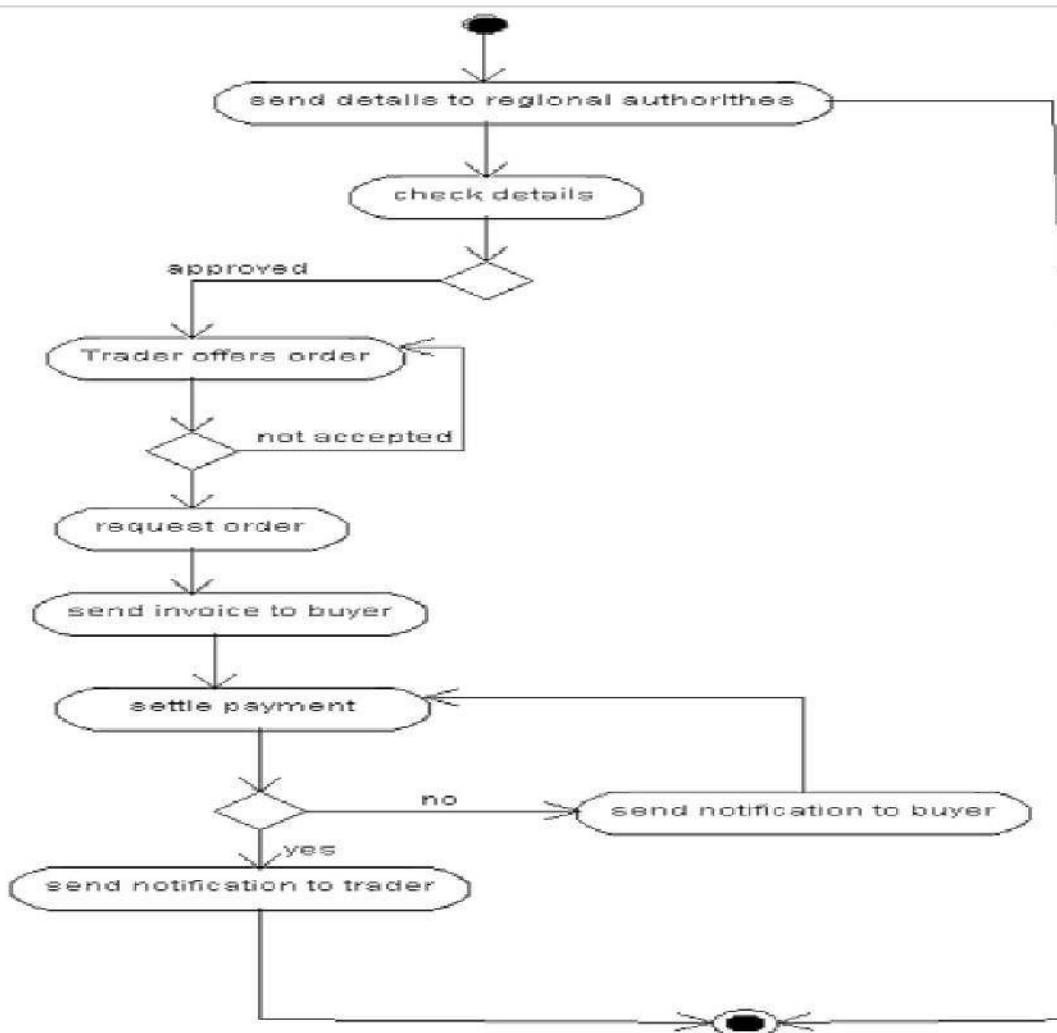
UML ACTIVITY DIAGRAM:-

Description:-

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

- How activities are coordinator to provide a service.
- The events needed to achieve some operation.
- How events in a single use case relate to one another.
- The events needed to achieve some operation.
- How activities are coordinator to provide a service.
- How events in a single use case relate to one another.

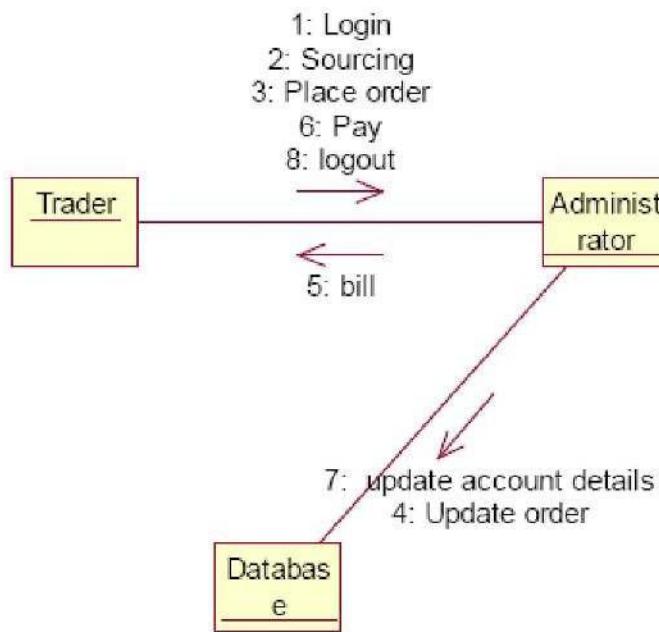


DOCUMENTATION OF ACTIVITY DIAGRAM:-

- Perform preliminaries activities i.e., getting IEC number from regional licensing authorities in the first action
- Submit a proposal order to the buyer in the send action
- After obtaining a confirmed order should produce the goods exactly as specifies in the invoice in third and fourth action.
- If the exporting house does not have production facilities, it has to procure the products from others.
- Transport the goods to the buyer in the sixth action.
- submit a order to the buyer in the send action
- After obtaining a confirmed order should produce the goods exactly as specifies in the invoice in third and fourth action.

UML COLLABORATION DIAGRAM:-

A collaboration diagram belongs to a group of UML diagrams called Interaction Diagrams. collaboration diagrams, like sequence diagrams, show how the objects interact over the course of time. collaboration diagrams show the sequence by numbering the messages on the diagram.

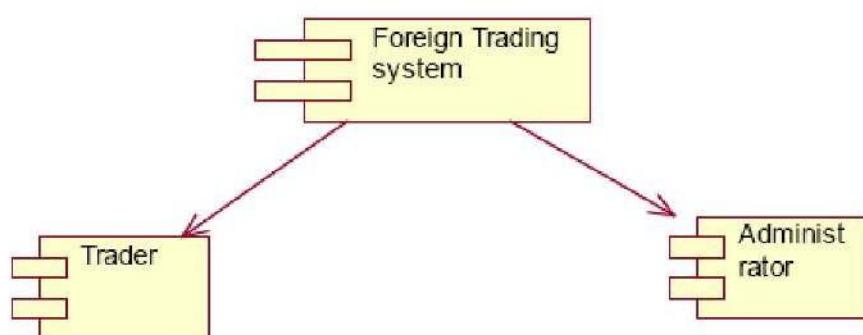


DOCUMENTATION OF COLLABORATION DIAGRAM:-

The collaboration diagram shows how the trader performs the sourcing and places order for which the administrator provides the bill and updates it in the database.

UML COMPONENT DIAGRAM:-

A component diagram depicts how the components are wired together to form larger components and or software systems.

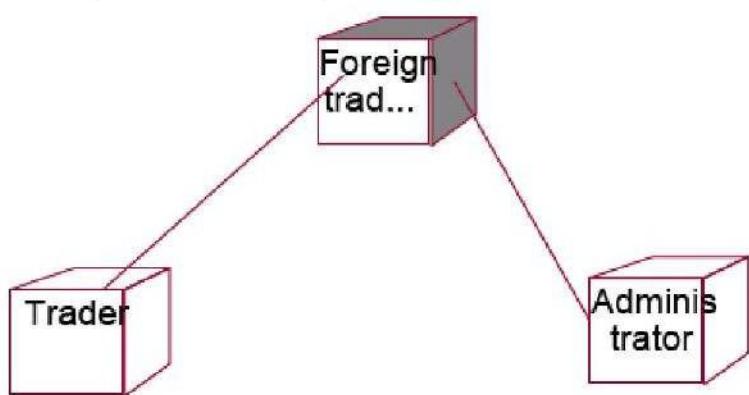


DOCUMENTATION OF COMPONENT DIAGRAM:-

The main component in the component diagram is foreign trading system. The trader who come to do the trading process and administrator who manages all the other processes is the sub components.

UML DEPLOYMENT DIAGRAM:-

A deployment diagram models the physical deployment of artifacts on nodes. The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub nodes, which appear as nested boxes.



DOCUMENTATION OF DEPLOYMENT DIAGRAM:-

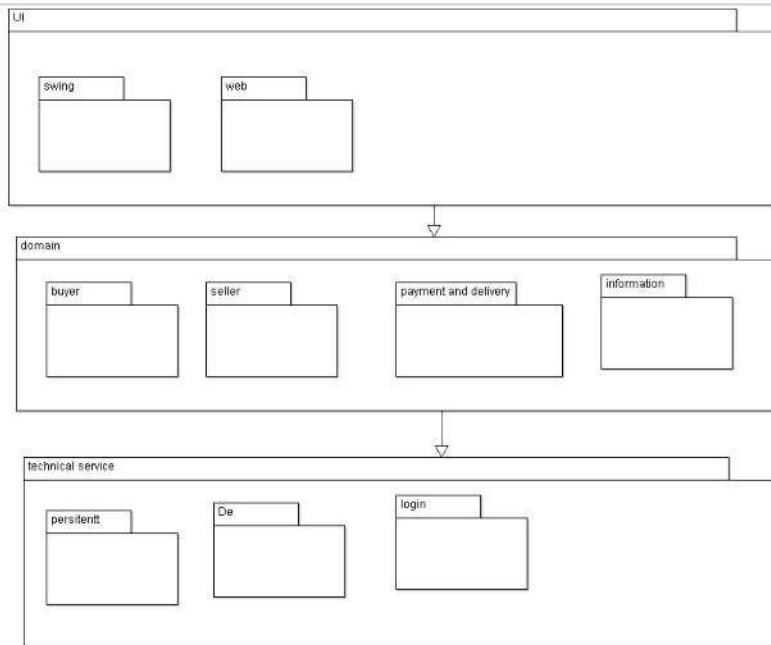
The processor in this diagram is the foreign trading system. The devices are the trader and administrator who perform the main activities in the system.

UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.

It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



UML TECHNICAL SERVICE LAYER

Trade Name	Trade No	Contact	Mail	Price
Siva Agencies	5390	2823992	Siva1@gmail.com	10,00,000
BMPSC	3487	2348930	bmpsc@gmail.com	2,00,000
Vicky Agencies	4589	2394030	Vky02@gmail.com	1,00,000
Buyer Name	Buyer No	Contact	Mail	Price
Priyanka	3348	2938490	priyank@gmail.com	10,00,000
Preetha	2093	2049034	prite@gmail.com	2,00,000
Lubna	2398	2349394	lubu@gmail.com	1,00,000

SAMPLE CODE:-

Code for preliminaries:

```

import java.util.Vector;
public class preliminaries
{
/* {src_lang=Java}*/
    private Integer indus name;
    public Integer reg no;
    public Integer intention of export;
    public Integer details of export goods;
}

```

Coding for regional authorities:

```

public class Regional Authorities
{
    /* {src_lang=Java}*/
    public Integer name of regional;
    public Integer emp.no;
    public Integer desigantion;
    public Integer mail id;
    public preliminaries mypreliminaries;
}

```

```

public Integer address;
public Integer certification on the quality
of export;
public Vector 2;
public void submit(){} 
public void cancel() {}
public void update() {}
public void verify() {}
public void accept() {}
public void rejects() {}
}
}

public void add()
{}
public void delete()
{}
public void update()
{}
public void generate authorities id()
{ }
}

```

USER INTERFACE LAYER:-

TRADER		BUYER	
TRADEER NAME	<input type="text" value="AMUL"/>	BUYER NAME	<input type="text" value="ARUN"/>
TRAEDER NO	<input type="text" value="562"/>	ADD	<input type="button" value="ADD"/>
ADDRESS	<input type="text" value="ANNA NAGAR"/>	SUBMIT	<input type="button" value="SUBMIT"/>
CONTACT	<input type="text" value="978561456"/>	CANCEL	<input type="button" value="CANCEL"/>
MAIL ID	<input type="text" value="AMUL123"/>	ADDRESS	<input type="text" value="AMMER NAGAR"/>
DES OF GOODS	<input type="text"/>	CONTACT	<input type="text" value="965832486"/>
PRICE	<input type="text"/>	MAIL ID	<input type="text" value="ARUN34"/>

Result:-

Thus the Foreign Trading System has been done successfully by using Argo-UML.

EX.NO:12	CONFERENCE MANAGEMENT SYSTEM
----------	------------------------------

AIM:

To design Conference Management System by using Argo-UML tool. .

PROBLEM ANALYSIS AND PROJECT PLAN :-

To simplify the process of applying, software has been created by designing through ARGO-UML tool.

The Conference Management System is an online website in which candidate can submit the paper and register themselves and then attend the conference. The paper will be reviewed. The details of the conference, date and time will be made available to them through the website. After getting the confirmation details the candidate should submit the revised and camera ready paper. Then the registration process will be done.

PROBLEM STATEMENT:-

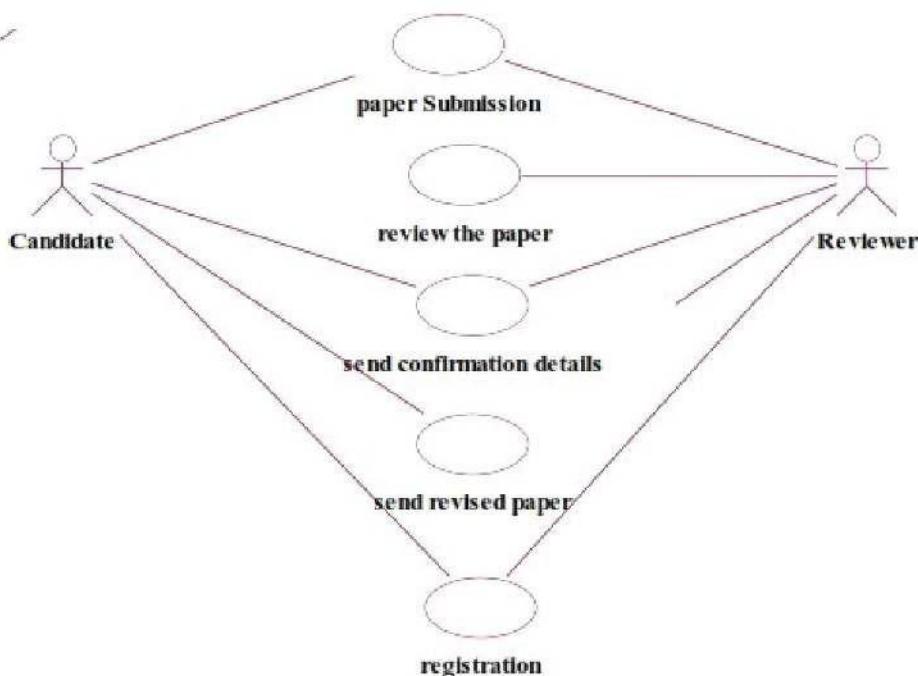
The process of the candidates is to login the conference system and submit the paper through online. Then the reviewer reviews the paper and sends the acknowledgement to the candidate either paper selected or rejected. This process of on conference management system are described sequentially through following steps,

- The candidate login to the conference management system.
- The paper title is submitted.
- The paper is been reviewed by the reviewer.
- The reviewer sends acknowledgement to the candidate.
- Based on the selection, the best candidate is selected.
- Finally the candidate registers all details.
- The paper title is submitted.
- The paper is been reviewed by the reviewer.
- The reviewer sends acknowledgement to the candidate.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.



DOCUMENTATION OF USECASE DIAGRAM:-

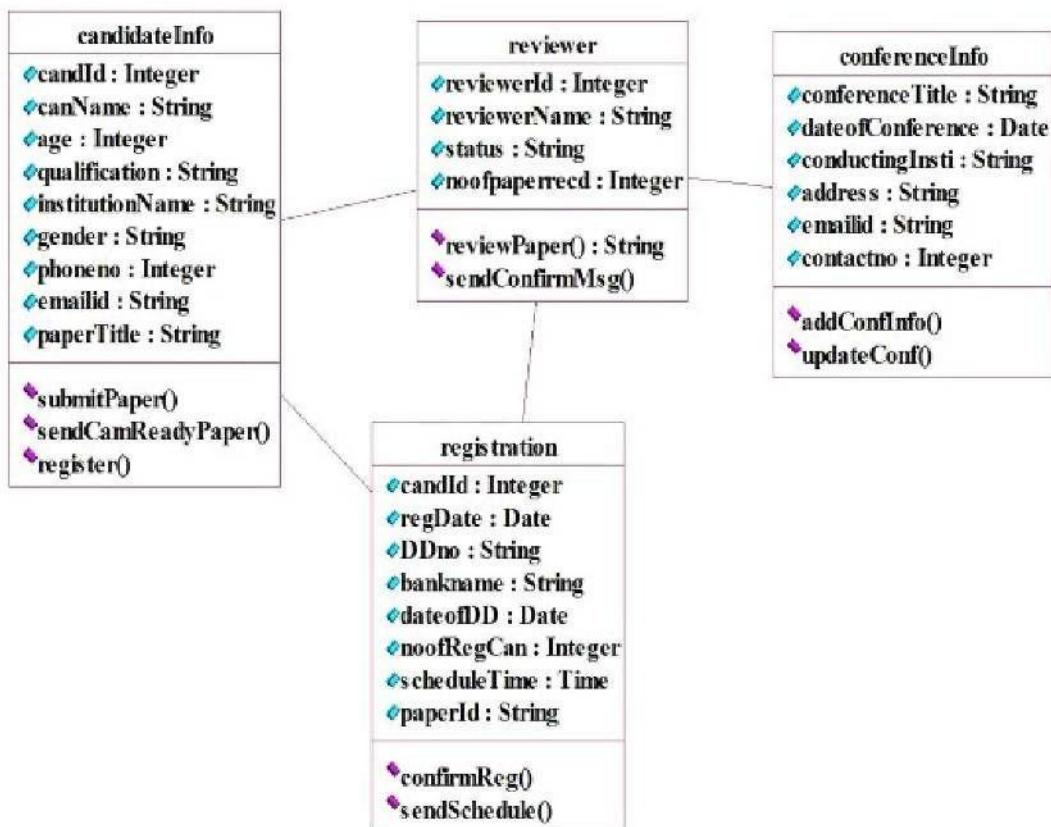
- | | |
|--|--|
| <ul style="list-style-type: none">• Candidate• Reviewer• Databases | <ul style="list-style-type: none">➤ Login➤ Paper submission➤ Review the paper➤ Paper confirmation details➤ Revised and camera ready paper➤ Registration |
|--|--|

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

This class diagram has three classes' candidate, reviewer and database.

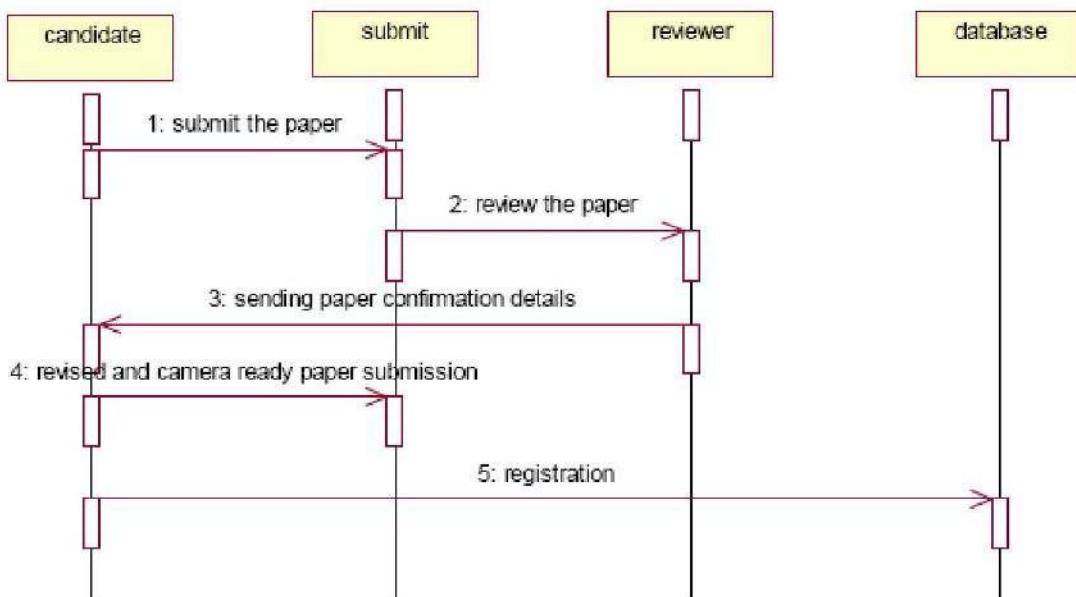
- Candidate – Its attributes are name, college name, department, paper title. The operations performed in the candidate class are login, submit the paper, submit revised and camera ready paper and registration.
 - Reviewer – Its attributes are name, department, reviewer ID and the operations performed are review the paper and send the paper confirmation details.
 - Database –The operations performed are storing candidate details and verifying login.
 - Reviewer – Its attributes are name, department, reviewer ID and the operations performed are review the paper and send the paper confirmation details.

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

LOGIN

PAPER SUBMISSION

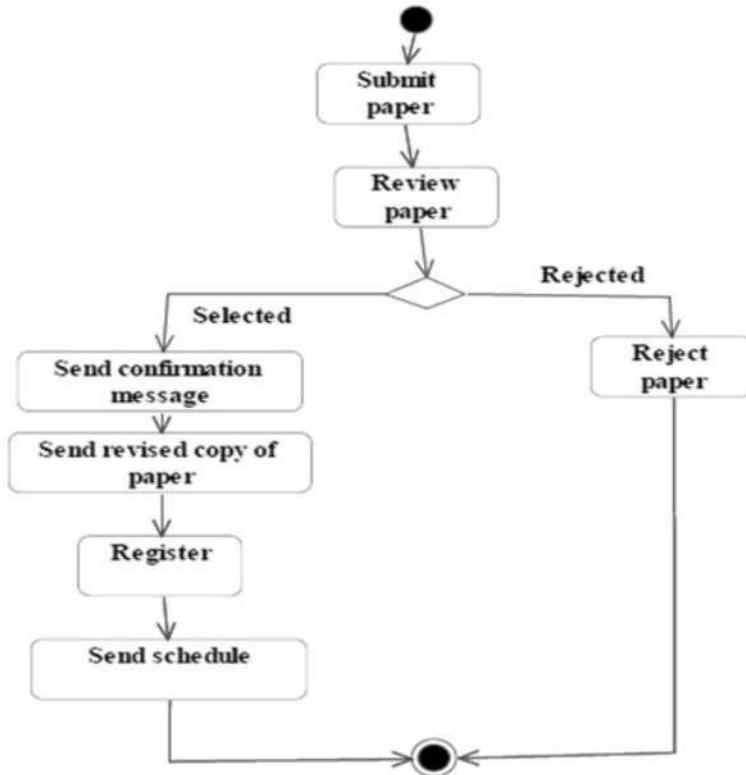
UML STATECHART DIAGRAM:-

Description:-

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

- How activities are coordinator to provide a service.
- The events needed to achieve some operation.
- How events in a single use case relate to one another.



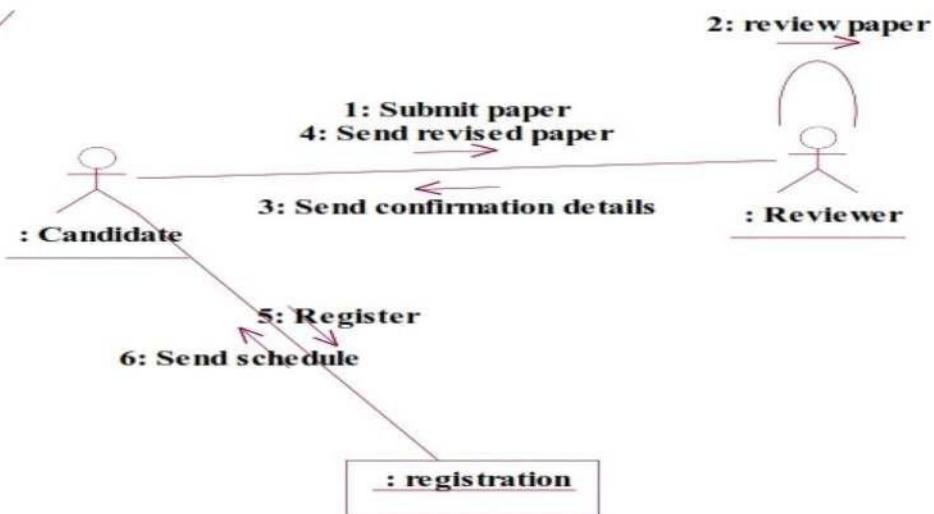
DOCUMENTATION OF ACTIVITY DIAGRAM:-

- First the candidate login to the database.
- Then the candidate should submit the paper.
- If it is selected the acknowledgement will send to the candidate.
- After submitting revised paper the registration process will be done.

UML COLLABORATION 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.

- First the candidate login to the database.
- Then the candidate should submit the paper.
- If it is selected the acknowledgement will send to the candidate.



DOCUMENTATION OF COLLABORATION DIAGRAM:-

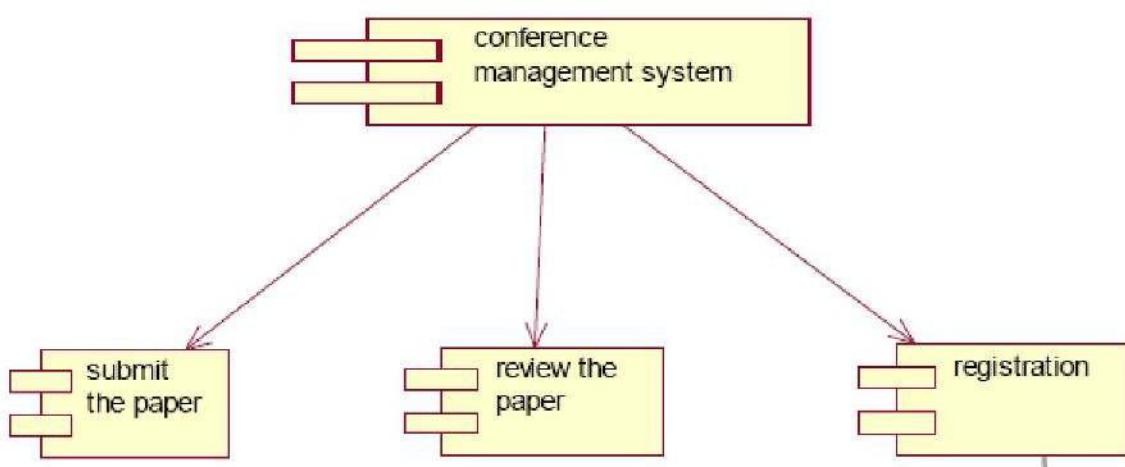
LOGIN

This collaboration diagram is to show how the applicant login in the conference system. Here the sequence is numbered according to the flow of execution.

This collaboration diagram is to show the submitting paper process of the candidate for the conference. The flow of execution of this selection process is represented using the numbers.

UML 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.

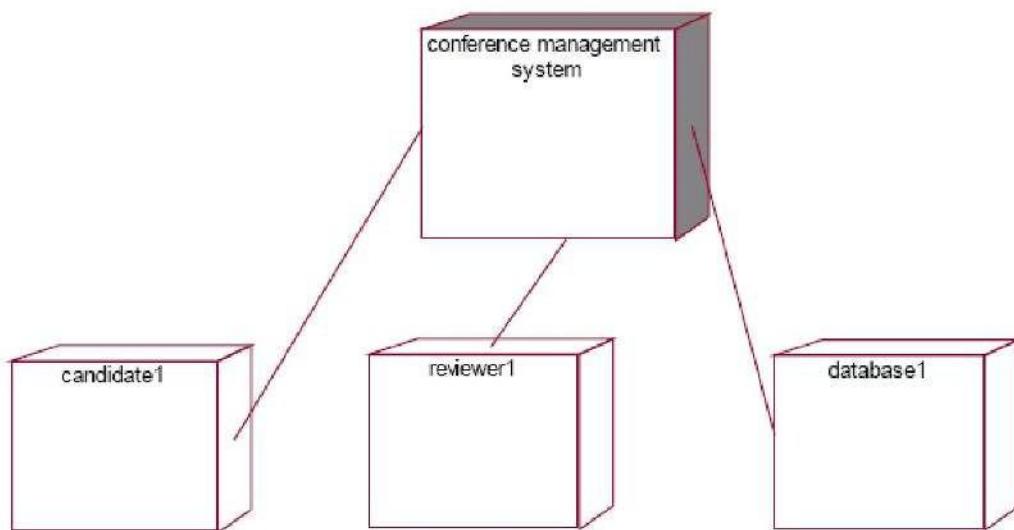


DOCUMENTATION OF COMPONENT DIAGRAM

The main component in this component diagram is conference management system. And submit the paper, review the paper and registration.

UML DEPLOYMENT DIAGRAM:-

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-dimensional box. Dependencies are represented by communication association.



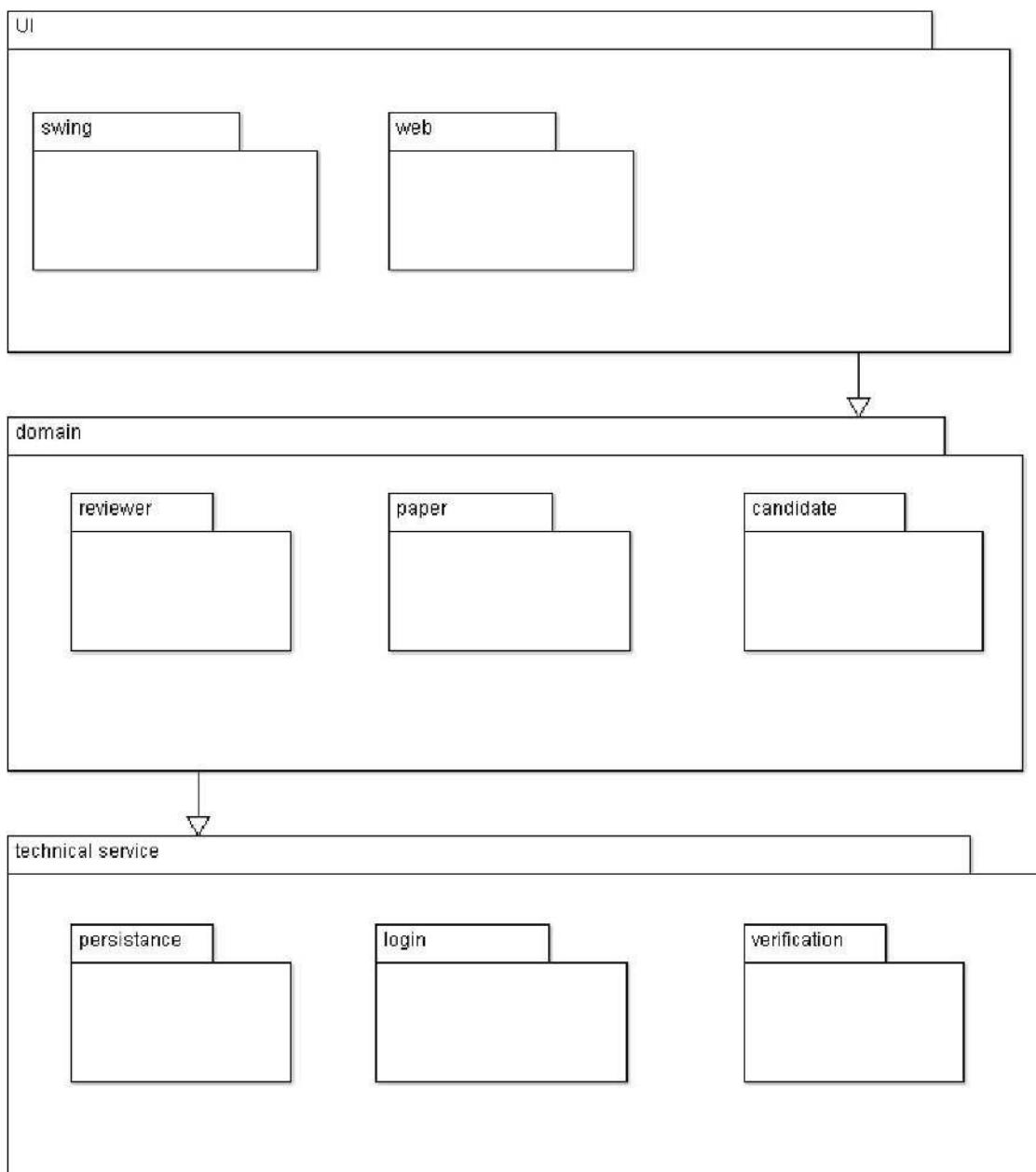
UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.

It is the graph of nodes connected by communication association. It is represented by three dimensional box. The device node is library management system and execution environment nodes are user, librarian, system and DBA.

The package diagram involves eight stages such as login, enter details, requesting for book, display book details, search book, issue book, return book and logout.



UML TECHNICAL SERVICE LAYER:-

S.N o	Name	Name of the Paper	Contact	E-mail
1.	Preethi	Cloud Computing	9324389383	preethi@gmail.com
2.	Nithya	Networking	9348739923	nadhisri@gmail.com
3.	Pavithra	Database	7354863948	pavi@gmail.com

S. No	Name of the Paper	Date	Time	Place
1.	Cloud Computing	02-09-2017	9.00 A.M	Chennai
2.	Networking	15-09-2017	10.00 A.M	Selam
3.	Database	03-10-2017	9.30 A.M	Bangalore

SAMPLE CODE:-

```

import java.util.Vector;
public class application {
    public char name;
    private char fathername;
    public int Dateofbirth;
    private varchar
permanentaddress;
    private varchar
Temporary_address;
    public varchar email;
    public int Phonenumber;
    public varchar panNo;
    public varchar ApplicationNO;
    public varchar Username;
    public varchar password;
    public Vector myDatabase;
    public void login() {
    }
    public void submitdetails() {
    }
    public void checkingstatus() {
    }
}

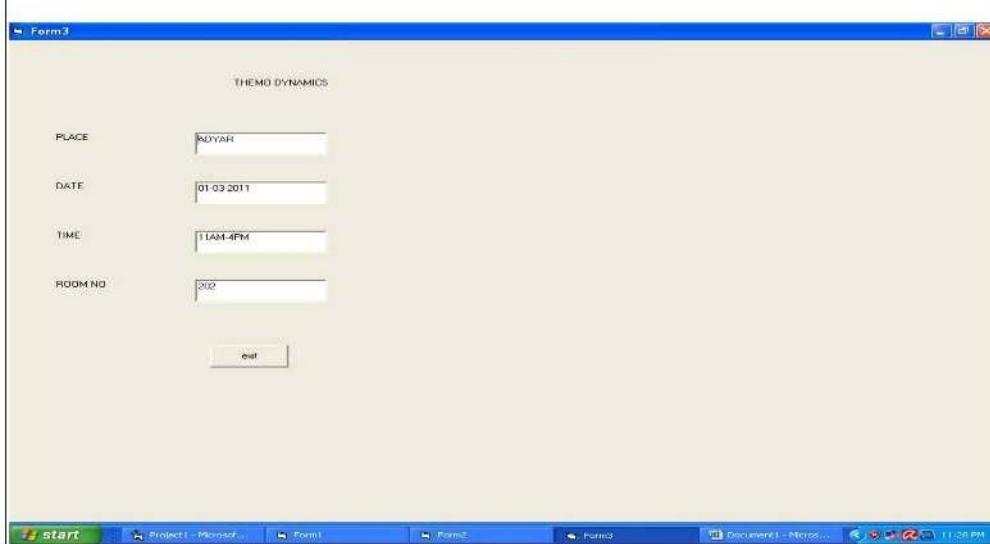
```

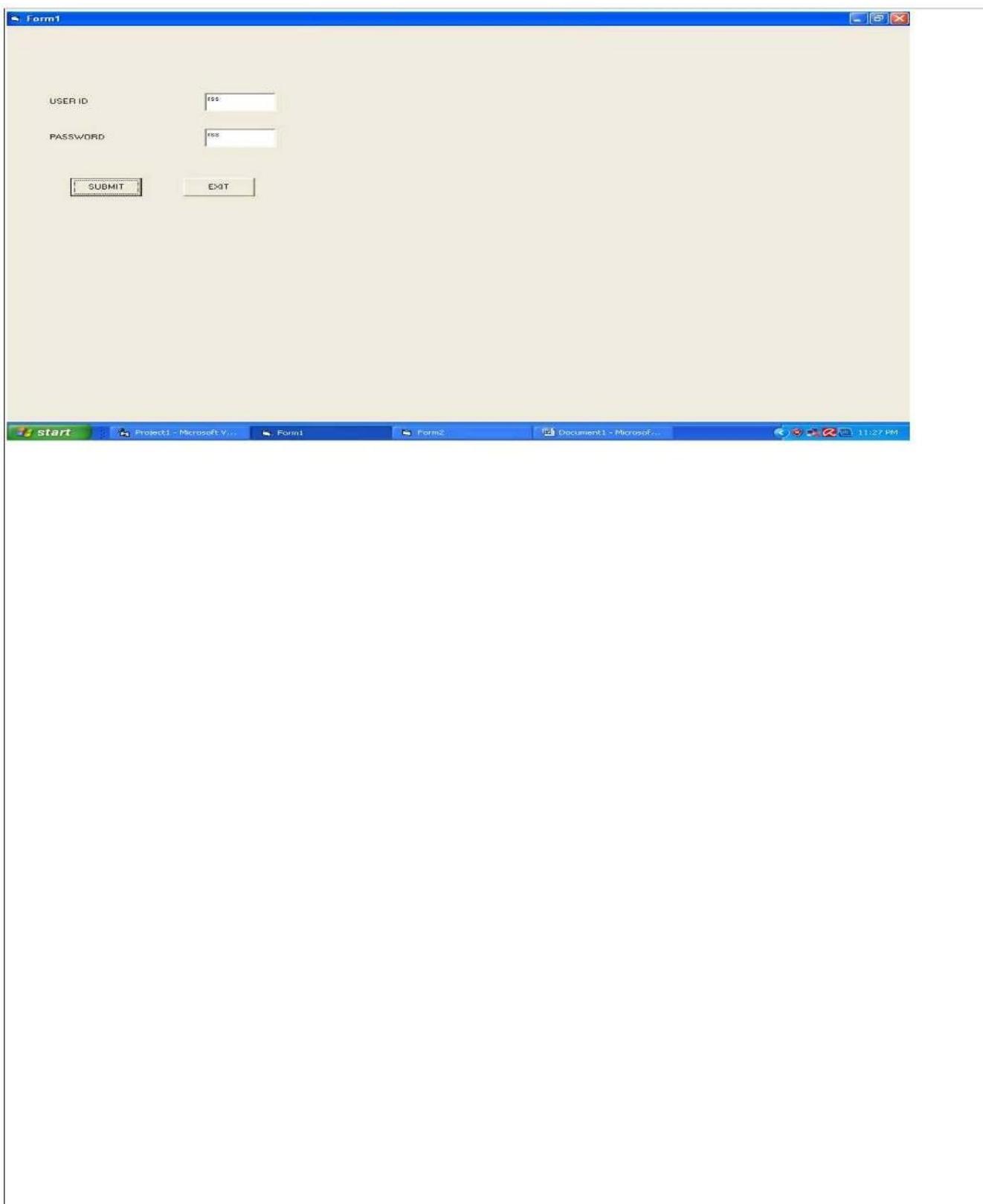
```

import java.util.Vector;
public class Database {
    public char name;
    public Vector myapplication;
    public Vector
mypassportAdministration;
    public Vector
myregionalAdminstrator;
    public Vector mypolice;
    public void store() {
    }
}

```

USER INTERFACE LAYER:-





Result:-

Thus the Conference Management System has been done successfully by using Argo-UML.

EX.NO:13

BPO MANAGEMENT SYSTEM

AIM:

To design BPO Management System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN :-

To simplify the process of applying, software has been created by designing through ARGO-UML tool.

Generally outsourcing can be defined as an organization entering into a contract with another organization to operate and managed one or more of its business process.

There are many problems faced by the BPO one among them is meeting their targets and leaving the concern very often and switch to another company.

In this project we deal with the inbound system of the BPO. In inbound system the agent calls the customer from his database to sell his product.

PROBLEM STATEMENT:

With the reduction in communication costs and improved bandwidths and associated infrastructure, BPO as a segment is witnessing a massive growth.

One of the key challenges that BPO companies that provide data entry/data validation services is an efficient and effective way of getting the source documents from different customers and accurately route the same to different operators for processing.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.

DOCUMENTATION OF USECASE DIAGRAM:-

ACTORS

- **BPO organization:**
- **Client**

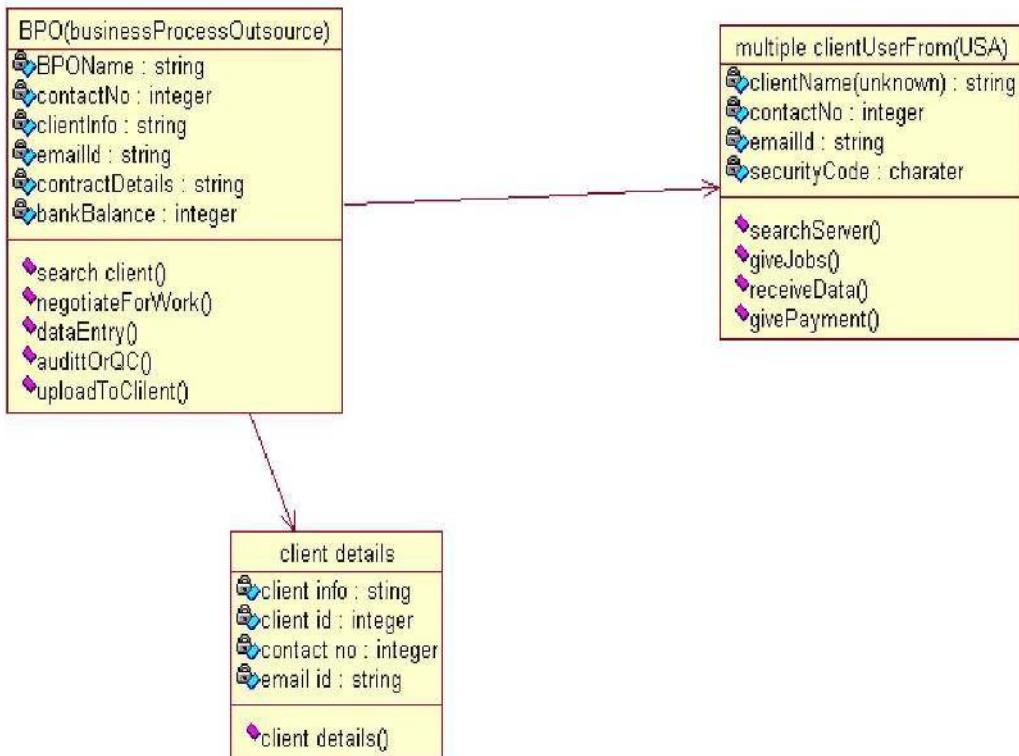
USE-CASE

- **Search for client/job**
- **Negotiate the project**
- **Upload input data**
- **Perform required conversion**
- **Quality Check**

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

This class diagram has three class process agent, customer and database.

- Agent – is the class name. Its attributes are username, password, name, phone no and address. The operations performed by the agent class are login, giving details to customer and selling the product.
- Customer – is the class name. Its attributes are name, phone no, address. The operations performed are attending the call, asks about the product.
- Database – is the class name. The operations performed are storing customer details, verifying login and updating the customer details.

The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.

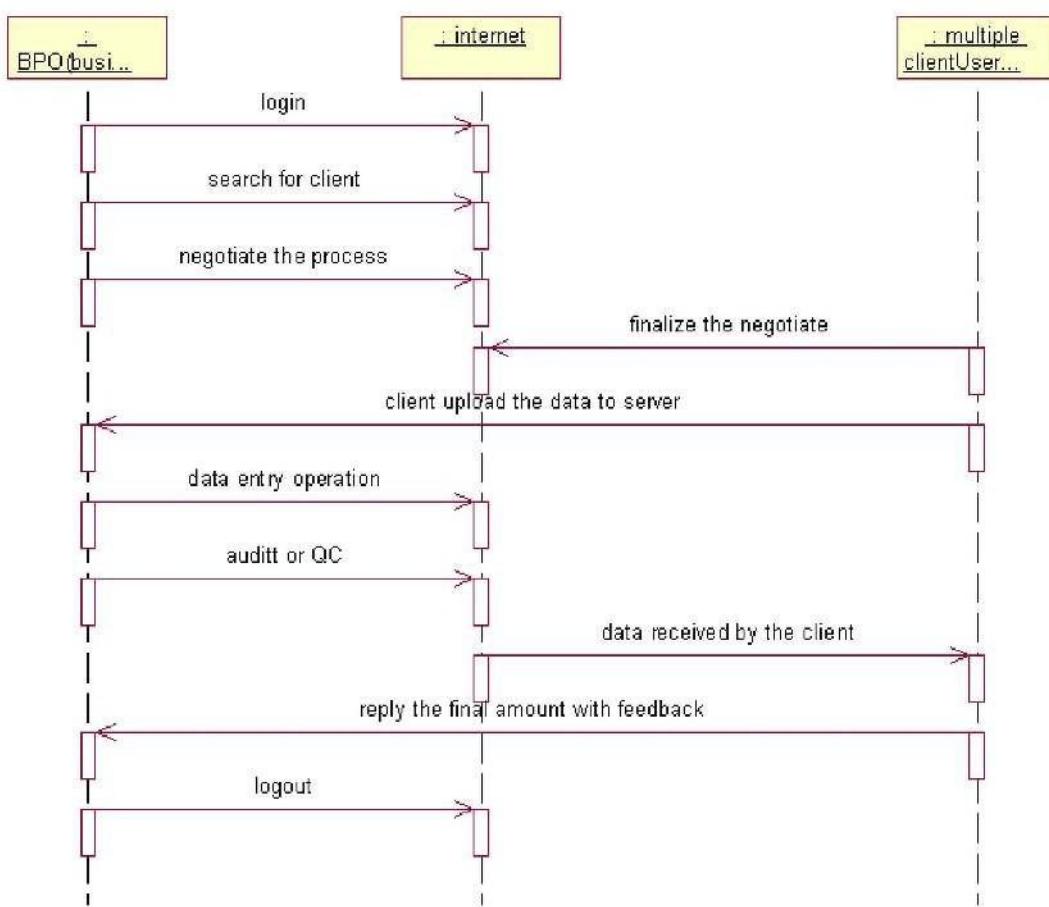
UML INTERACTION DIAGRAM:-

Description:-

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.

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



DOCUMENTATION OF SEQUENCE DIAGRAM:-

The single use case in BPO management system is taken and sequence of operations followed in the use case.

The BPO has the following sequence of process:

1. Agent fetches the data from the database.
2. Database provides the details of the customer to agent and agent dials to the customer.
3. Customer responds to the agent and agent pitches his/her product.
4. If necessary customer buys else discards.
5. Agent updates the call history.
6. Proceeds with another call.

UML STATECHART DIAGRAM:-

Description:-

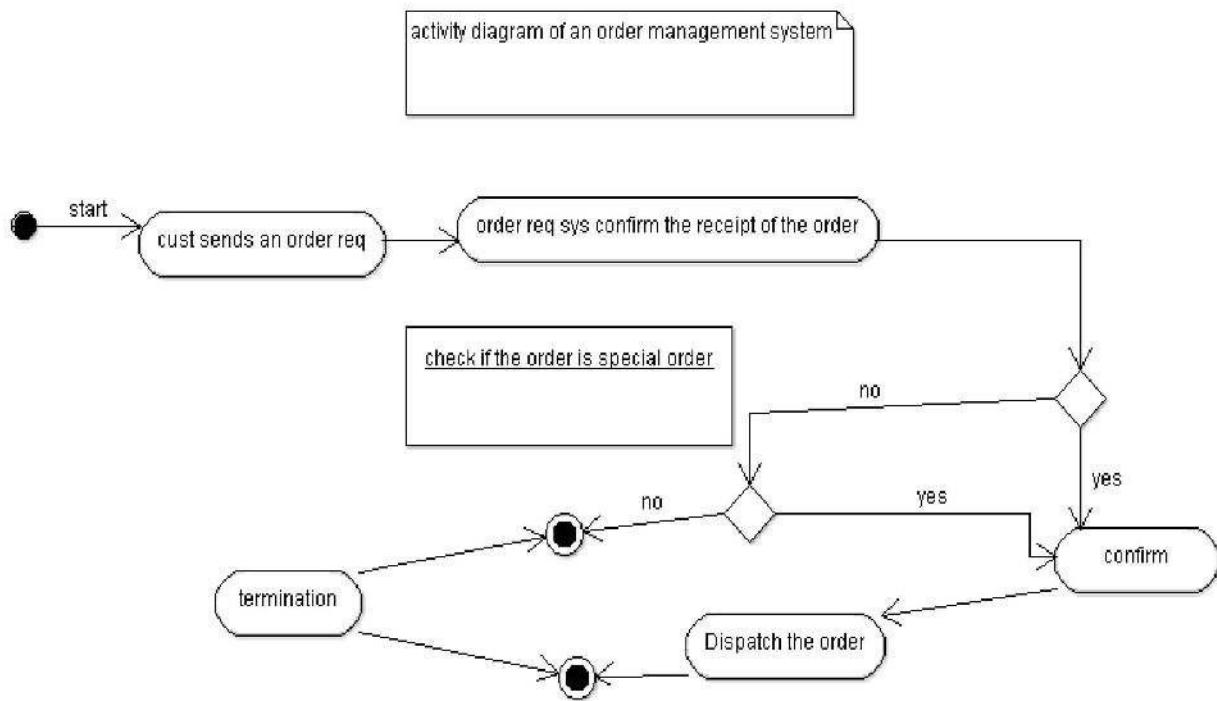
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

How activities are coordinator to provide a service.

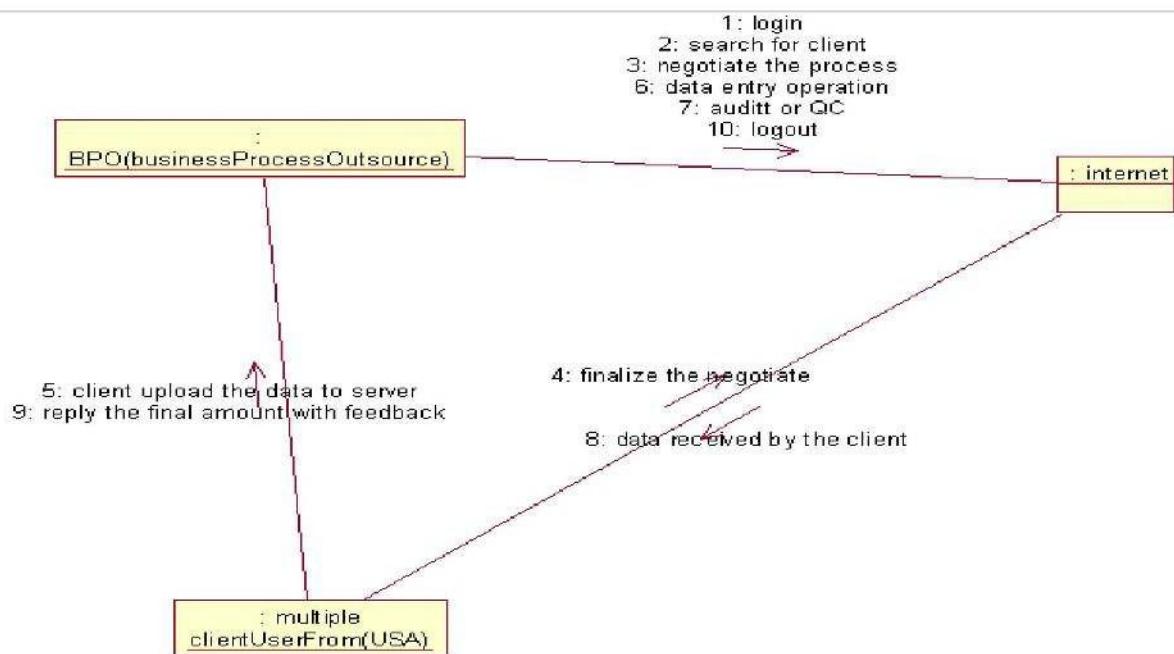
The events needed to achieve some operation.

How events in a single use case relate to one another.



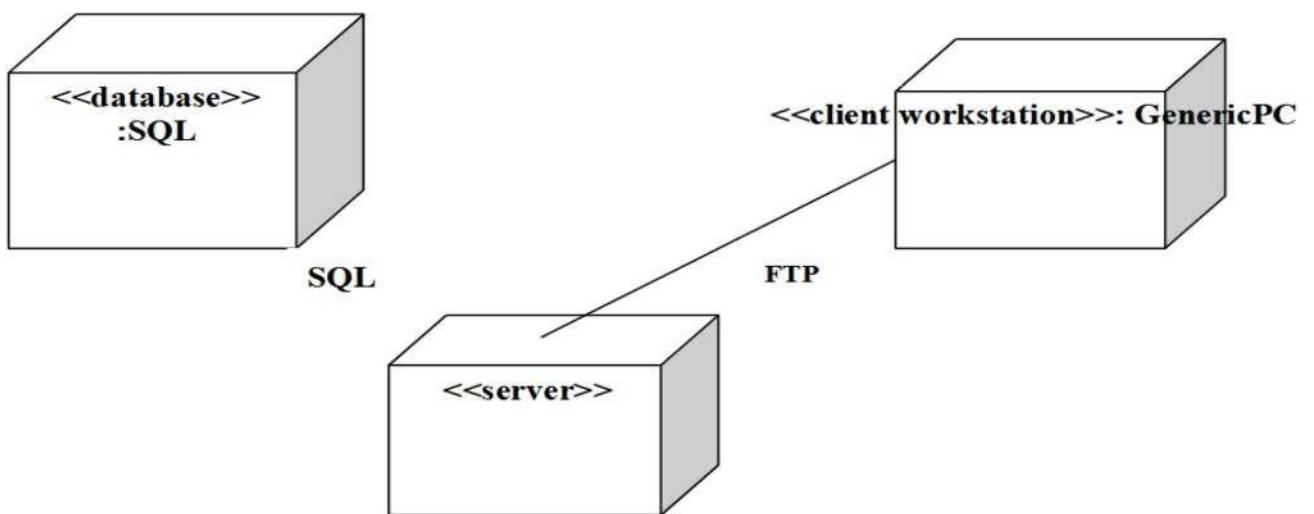
UML COLLABORATION DIAGRAM:-

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.



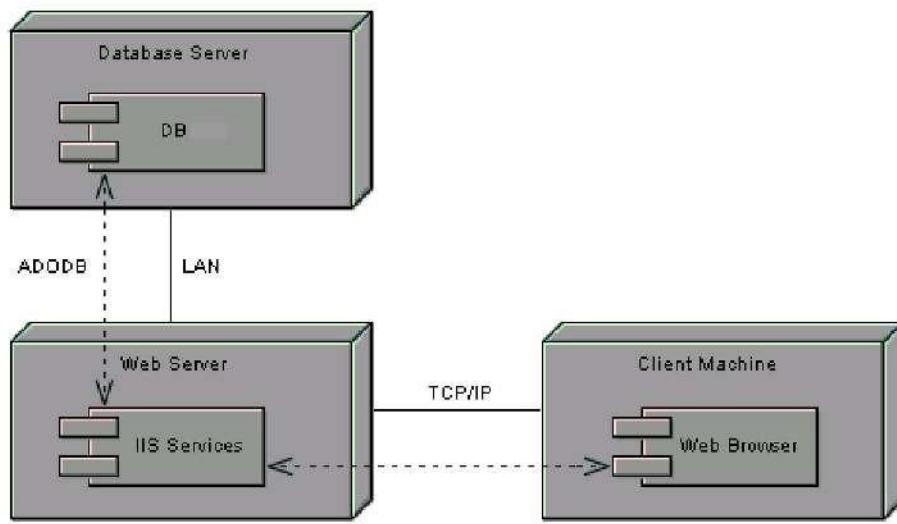
UML DEPLOYMENT DIAGRAM:-

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



UML COMPONENT DIAGRAM:-

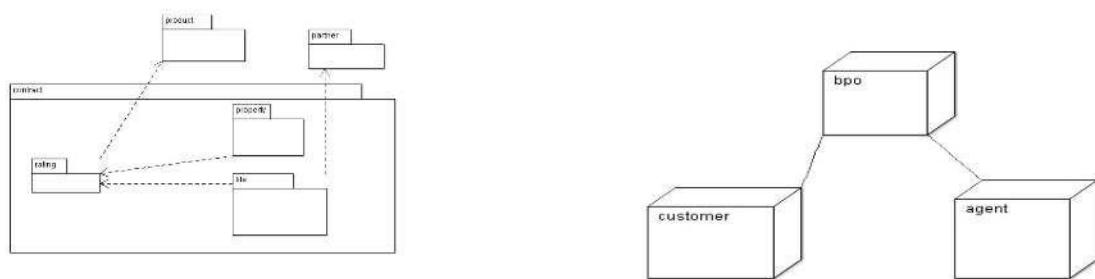
Component diagrams are used to visualize the organization and relationships among components in a system.



UML PACKAGE DIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.



DOCUMENTATION OF PACKAGE DIAGRAM

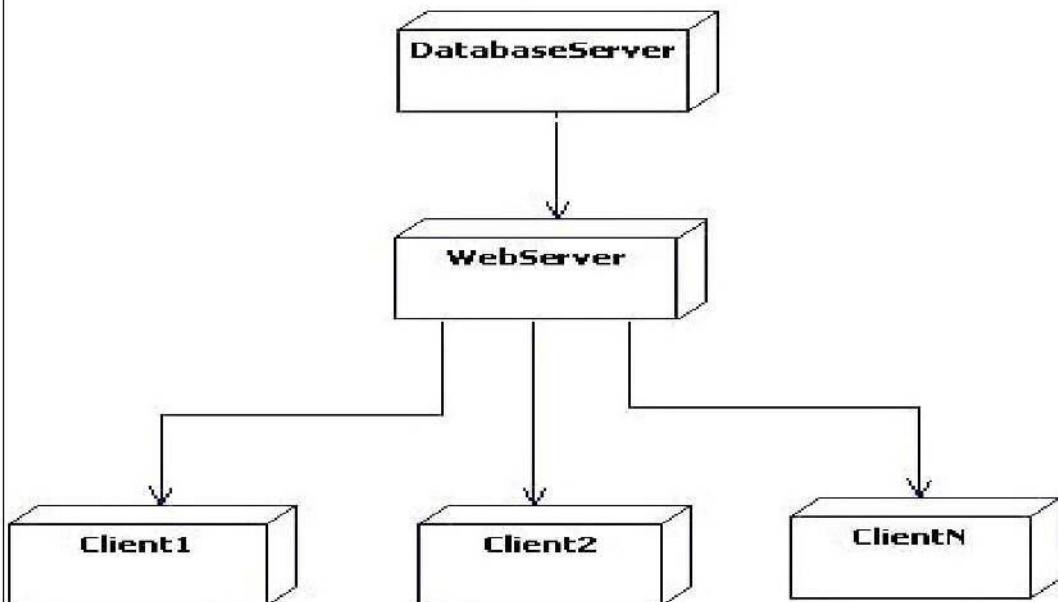
The three layers in BPO management systems are

- The User interface layer – consists of the web and login. This layer describes how

The agent logs on to the website and gets the customer details.

- The Domain layer – Shows the activities that are performed in the BPO management system. The agent makes the call and he pitches about the product to customer and makes sale. Finally agent aborts the call and proceeds with another call.
- The Technical service layer – The customer details are shown in the database. If the customer buys product it makes the sale entry.
- Activity final node – The filled circle with a border is the ending point. An activity diagram can have zero or more activity final nodes.
- Activity – The rounded rectangles represent activities that occur. An activity may be physical, such as Inspect Forms, or electronic, such as display the BPO details.
- Flow/Edge – The arrows on the diagram. Although there is a subtle difference between flows and edges I have never seen a practical purpose for the difference although I have no doubt one exists. I'll use the term flow.

UML TECHNICAL SERVICE LAYER:-



SAMPLE CODE:-

BPO Organization

```
import java.util.Vector;

public class BPO organization {
    public String orgname;
    public String Address1;
    public String Address2;
    public String Address3;
    public String State;
    public Integer Contact_no;
```

Employee Details:-

```
import java.util.Vector;

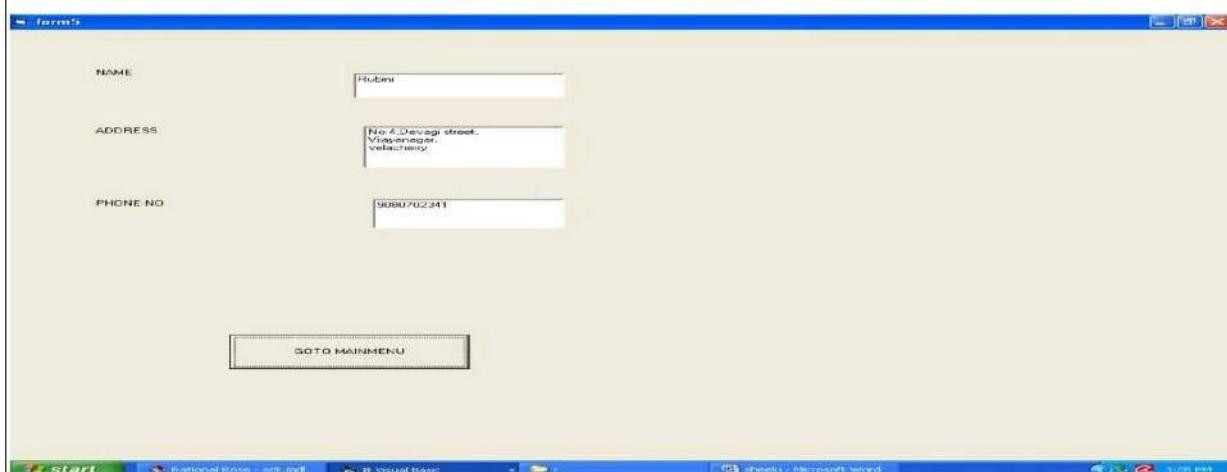
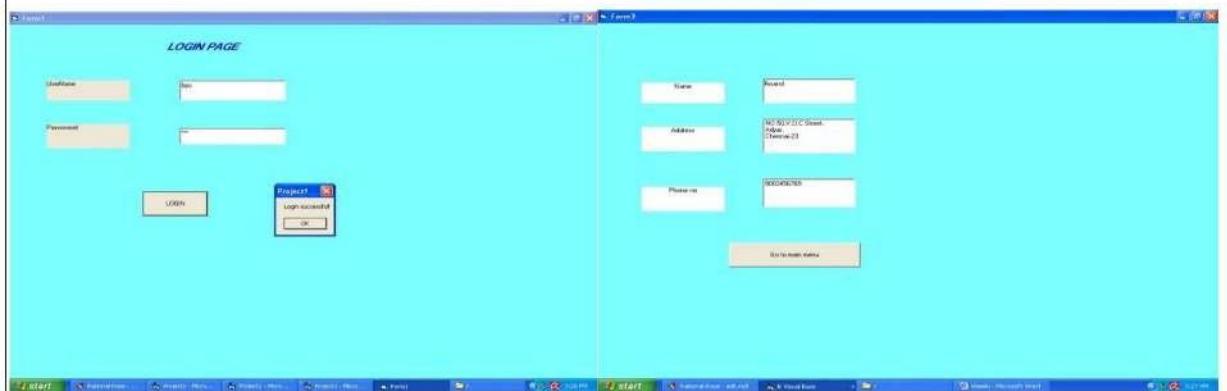
public class employee details {
    public String emp_name;
    public Char emp_dob;
    public Varchar emp_addr;
    public ph.no emp;
    public Vector myBPO organization;
    public Vector myproject Details;
```

```

public Varchar Email;
public Integer Toata;
public Vector myproject Details;
public Vector myemployee details;
public Vector myprocess;
public void addOrg0 {
}
public void updateOrg0 {
}
}

```

USER INTERFACE LAYER:-



Result:-

Thus the BPO Management System has been done successfully by using Argo-UML.

EX: NO:14	LIBRARY MANAGEMENT SYSTEM
-----------	---------------------------

AIM:

To design Library Management System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN

To simplify the process of applying Library Management System, software has been created by designing through ARGO-UML tool.

PROBLEM STATEMENT:-

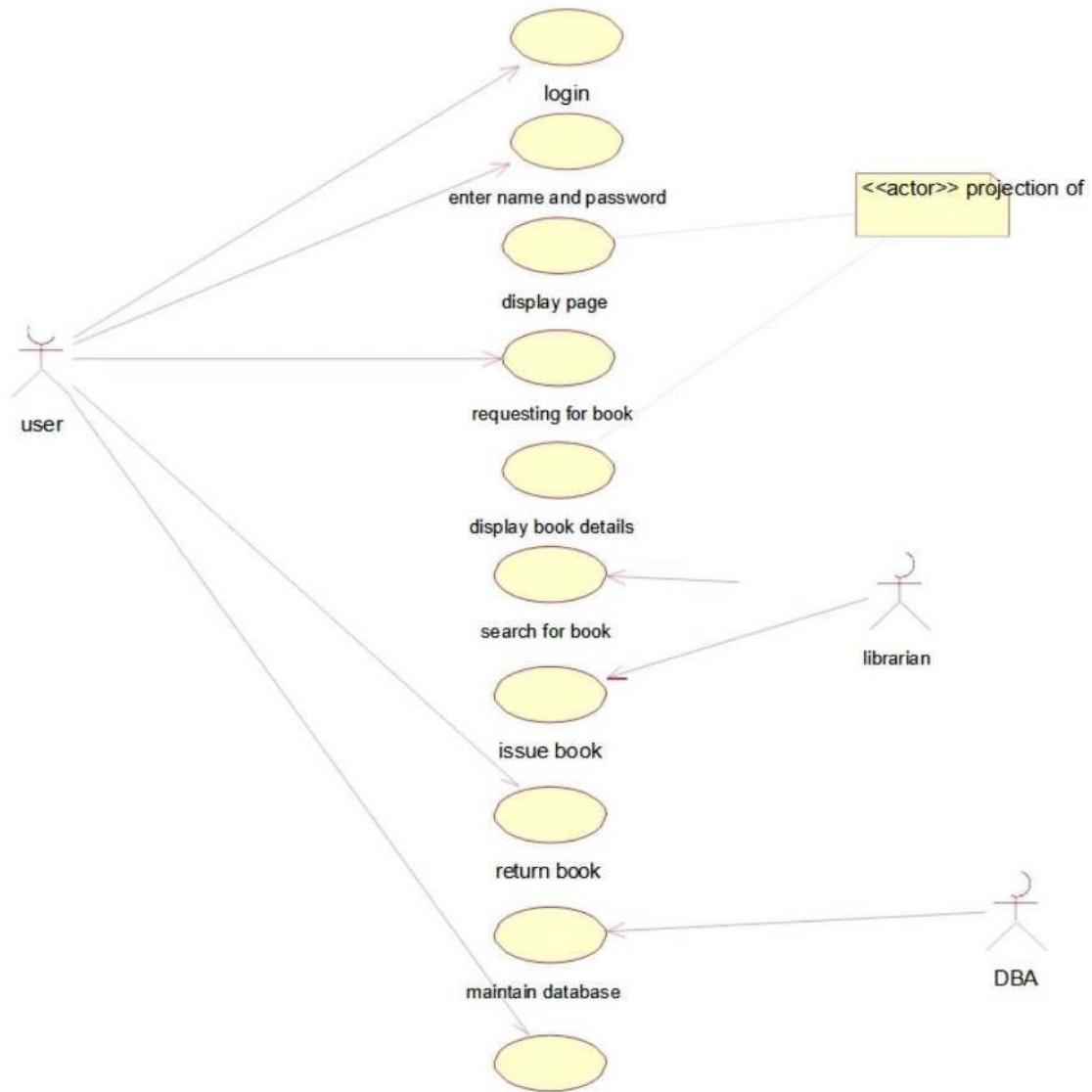
- The software to be designed will control a simulated library management system.
- As the student enters the library students gets a library card.
- The library card is used for borrowing, lending and paying fines for books.
- The librarian is the person who is an intermediate between the student and the database.
- The library will service more than one student
- A student will be required to show his/her library card to the librarian.
- Once the librarian gets to know that the borrowed book is lost .The librarian immediately accesses the database and updates that the book is lost in the login and fine is also calculated.

UML USECASE DIAGRAM:-

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.

Use case is a list of actions or events. Steps typically defining the interactions between a role and a system to achieve a goal. The use case diagram consists of various functionality performed by actors like user, librarian, system and DBA.



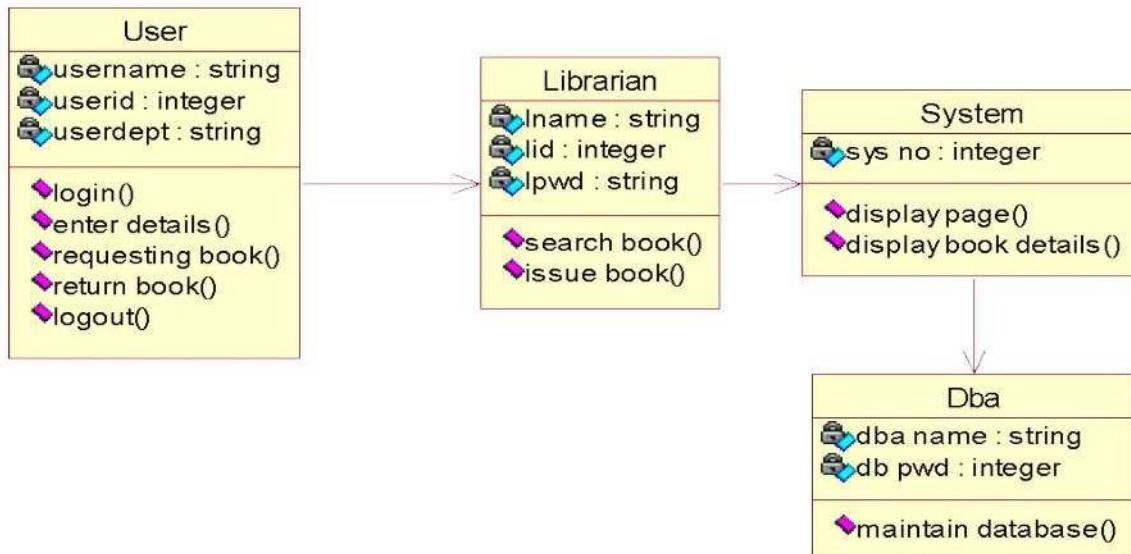
DOCUMENTATION OF USECASE DIAGRAM:-

1. Students and Staffs access his account by providing the correct userid and password.
2. The Students & Staffs searches for the Availability of Books.
3. The Students selects the Book which he wants from the listed stocked books.
4. The Book is issued to the Student or Staff and the database is updated.
5. The Student can either can continue to getting another books from library.

UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

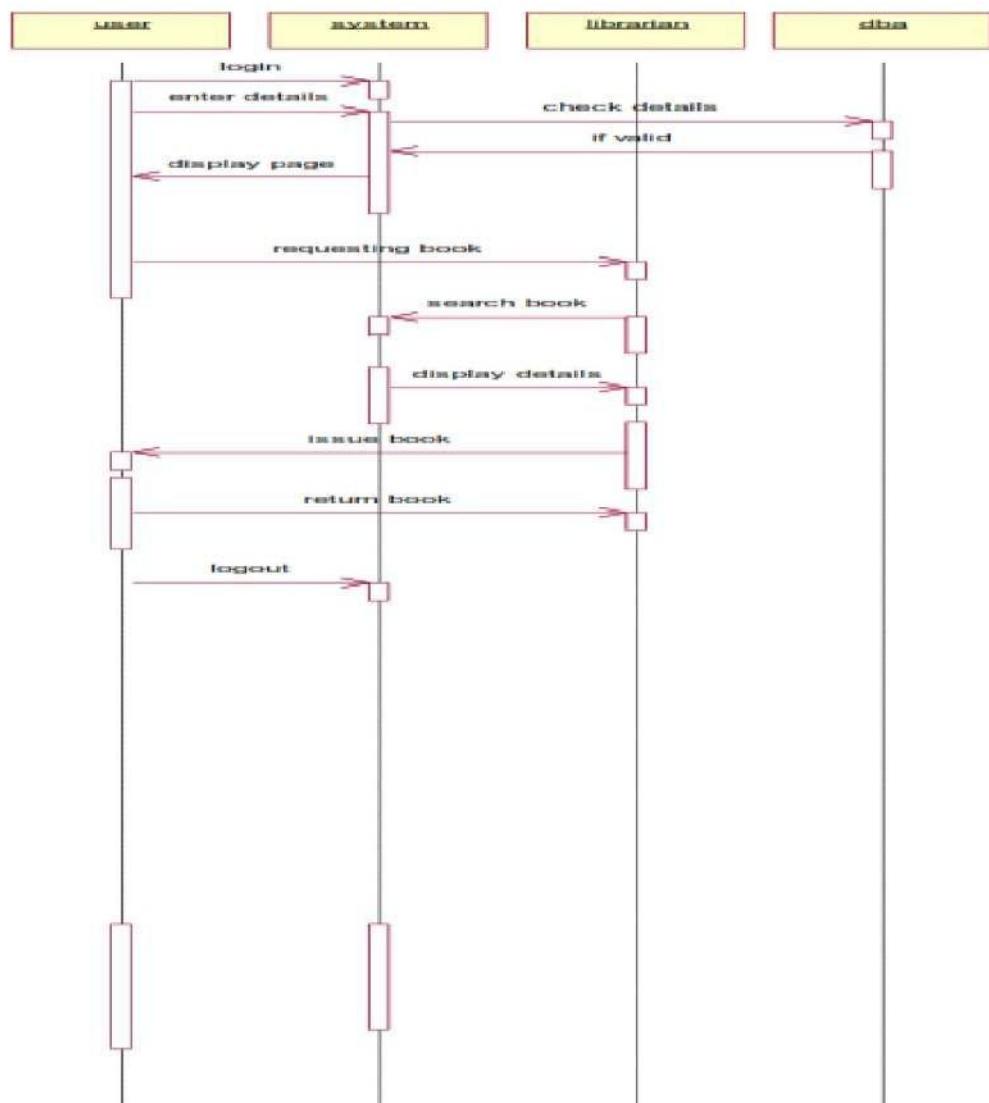
- Administration class
 - Get Membership
 - Search books
 - Issue book
 - Return book
 - Journals
-
- Here Administration class is related to get membership class.
 - Search books are related to journal class.
 - Issue book is related to return book.

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

- The student request membership to the admin.
- The admin generates membership to the student.
- The student enters the login id.

- The admin verifies the login id with the database.
- If the admin is correct authentication is granted by the admin. Otherwise the login id must be entered again.
- The student enters the details of the book (searching the book).
- The admin checks for the availability of the particular book.
- If that book is available it is borrowed.
- After borrowing book the admin updates the status in the student database.

UML STATECHART DIAGRAM:-

Description:-

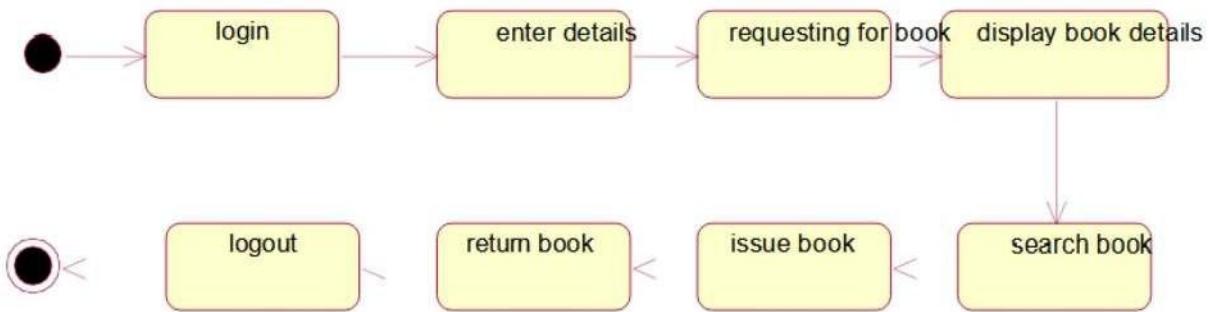
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

How activities are coordinator to provide a service.

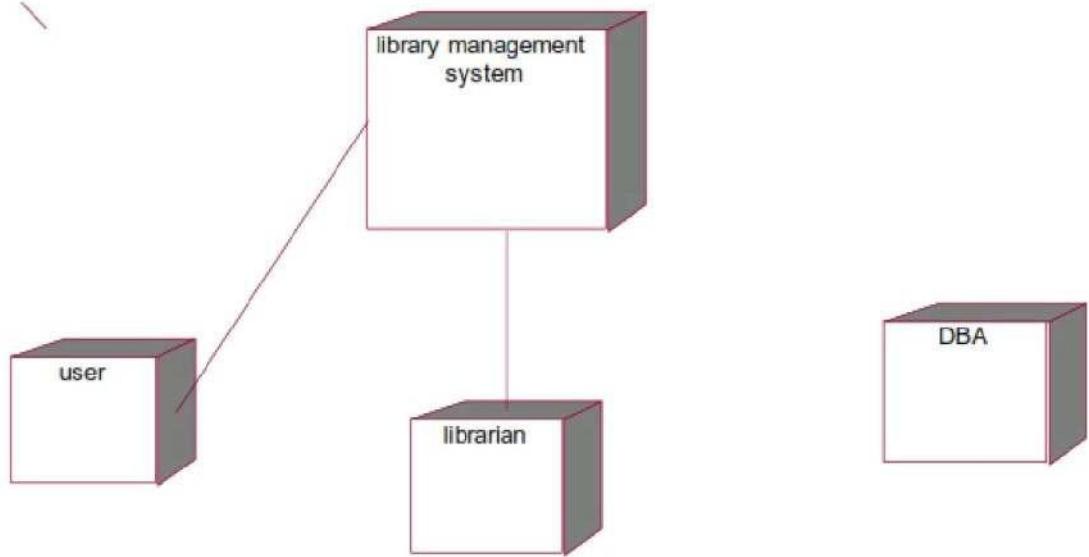
The events needed to achieve some operation.

How events in a single use case relate to one another.



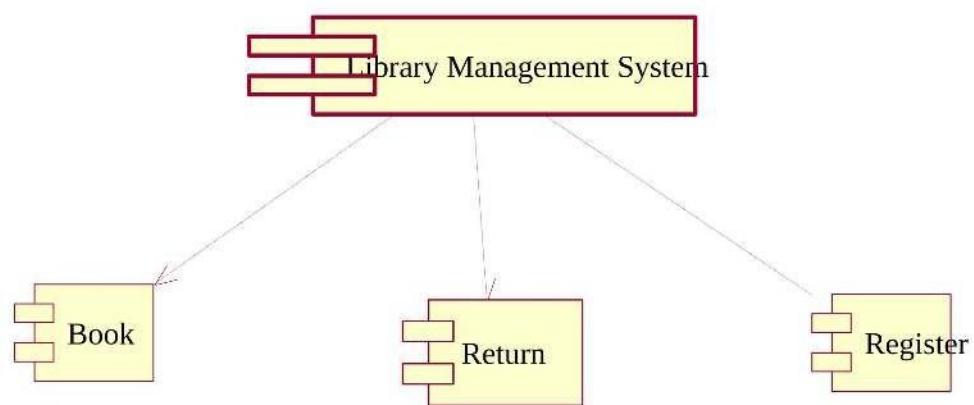
UML DEPLOYMENT DIAGRAM:-

Deployment diagram is a structure diagram which shows architecture of the system as deployment of software artifacts to deployment target. It is the graph of nodes connected by communication association. It is represented by three dimensional box. The device node is library management system and execution environment nodes are user, librarian, system and DBA.



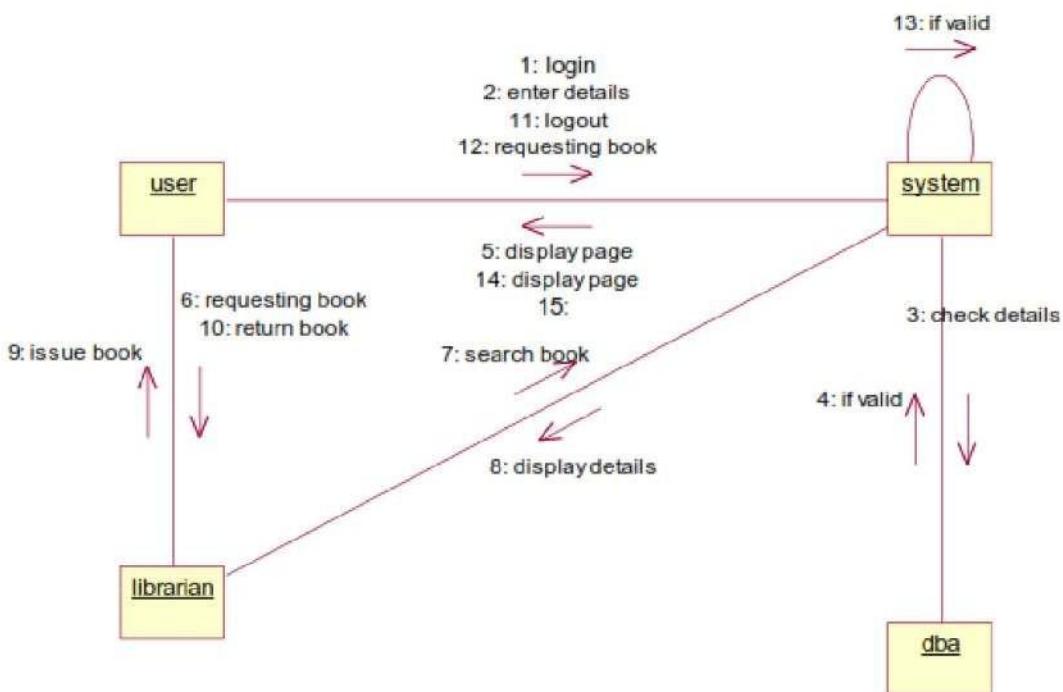
UML COMPONENT DIAGRAM:-

Component diagram shows the dependencies and interactions between software components. Component diagram carries the most important living actors of the system i.e, user, librarian and DBA.



UML COLLABORATION DIAGRAM:-

Like sequence diagram collaboration diagrams are also called as interaction diagram. Collaboration diagram convey the same informations as sequence diagram but focus on the object roles instead of the times that messages are sent. Here the actions between various classes are represented by number format for the case of identification.



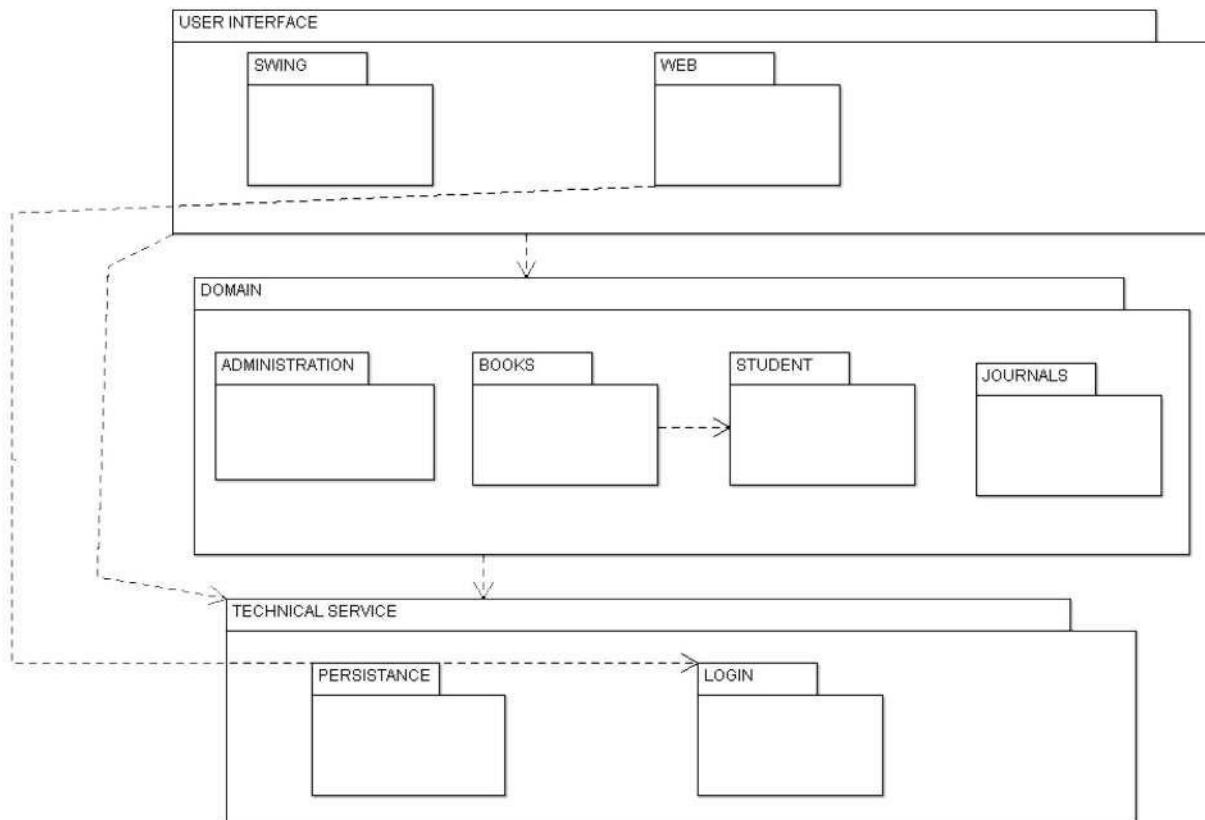
UML PACKAGE DIAGRAM:

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.

It is the graph of nodes connected by communication association. It is represented by three dimensional box. The device node is library management system and execution environment nodes are user, librarian, system and DBA.

The package diagram involves eight stages such as login, enter details, requesting for book, display book details, search book, issue book, return book and logout.



UML TECHNICAL SERVICE LAYER:-

S.No	Name of the user	Book Name	Book No	Date of Issue	Date of Return
1.	Monisha	Java	2426	17-06-2017	26-06-2017
2.	Swetha	Networks	1282	15-07-2017	24-07-2017
3.	Mutta	DPSD	3782	11-08-2017	20-08-2017

S.No	Book Name	Book Author	Book No	Price
1.	Java	Complete Reference	2426	300/-
2.	Computer Networks	S.Davie	1282	500/-
3.	Software Engineering	R.S.Pressmen	9090	400/-

SAMPLE CODE:-

```

Public class Books {
    Private object Name;
    Private object Identify;
    Private object Book Details;
    Private object Edition;
    Public void Book Title () {
    }
    Public void Edition () {
    }

```

```

Public class Staff {
    Private object Name;
    Private object identity;
    Private object Address;
    Public void Allotment of Books () {
    }
    Public void Return Books () {
    }
    Public void Fine for Missing () {

```

```

Public void Price () {
}
Public void Remarks () {
}
Public void Savings () {
}
}

```

USER INTERFACE LAYER:-



Result:-

Thus the Library Management System has been done successfully by using Argo-UML.

EX.NO:15	STUDENT INFORMATION SYSTEM
----------	----------------------------

AIM:

To design Students Information System by using Argo-UML tool.

PROBLEM ANALYSIS AND PROJECT PLAN :-

To simplify the process of applying, software has been created by designing through ARGO-UML tool.

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

PROBLEM STATEMENT

- Effective for Administration Purpose
- Cheap
- Better Service

UML USECASE DIAGRAM

Description:

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It is represented using ellipse. Actor is any external entity that makes use of the system being modeled.

DOCUMENTATION OF USECASE DIAGRAM:-

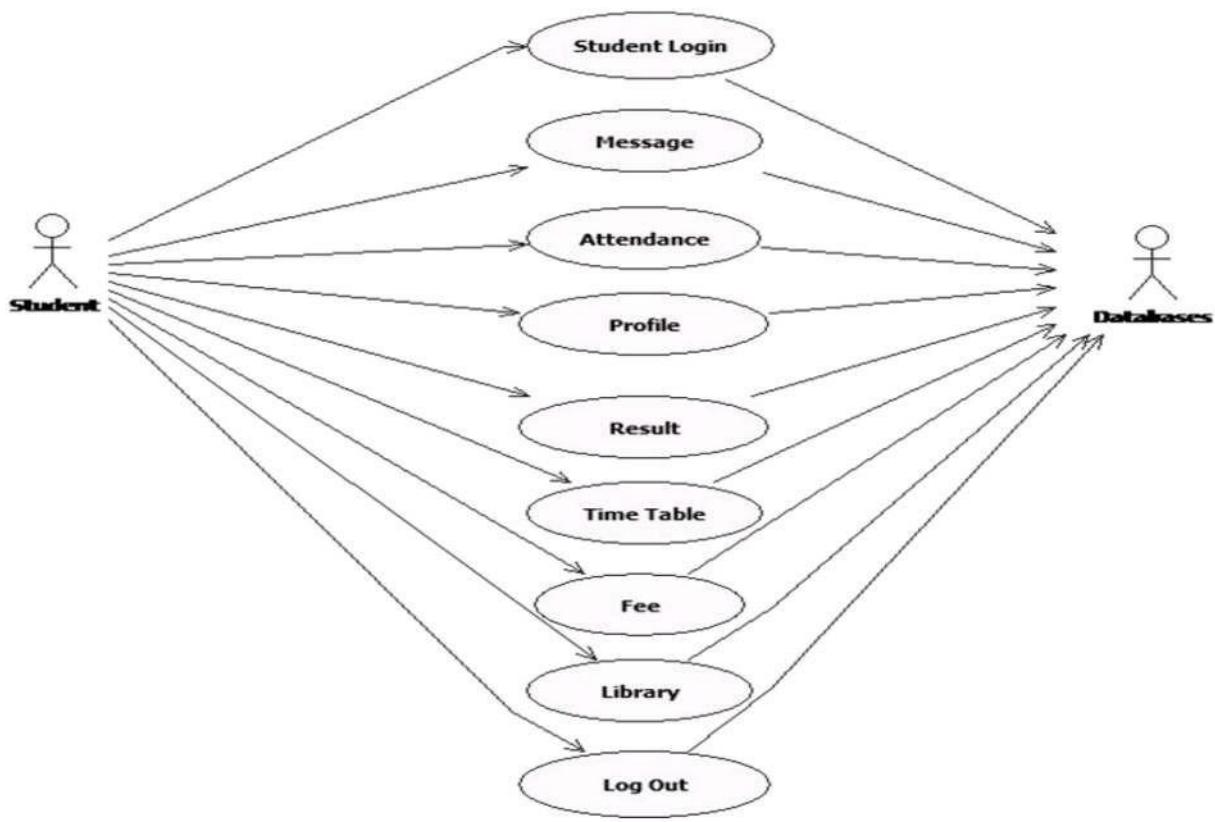
The actors in this use case diagram are Admin, Student, and Database. The use cases are the activities performed by actors.

Admin register login, and store the student records details in database.

Student Register from the Student Login process.

Then the database is searched for details and verified.

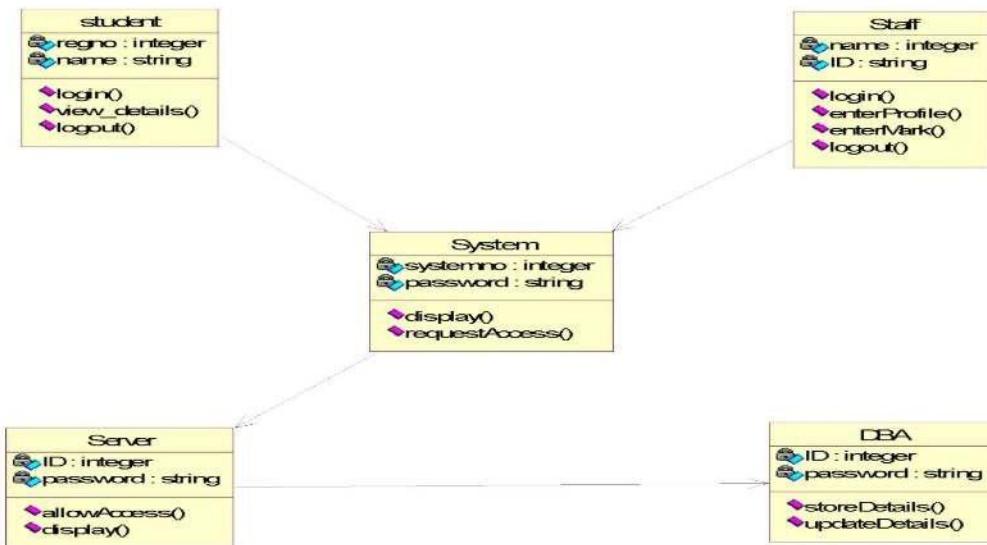
Database stores the details and returns acknowledgement



UML CLASS DIAGRAM:-

Description:-

A class is drawn as rectangle box with three compartments or components separated by horizontal lines. The top compartment holds the class name and middle compartment holds the attribute and bottom compartment holds list of operations.



DOCUMENTATION OF CLASS DIAGRAM:-

This class diagram has three classes Login, Student details and Update details in database.

a. Students – is the class name. Its attributes are name, Address, DOB, Gender, College, Subjects, Semester, Year, Degree, and Branch. The operations Performed in the students class, Store database and Update.

b. Administration– is the class name. Its attributes are Login, Password and database. The operations performed are Student Details store in database and send acknowledgement.

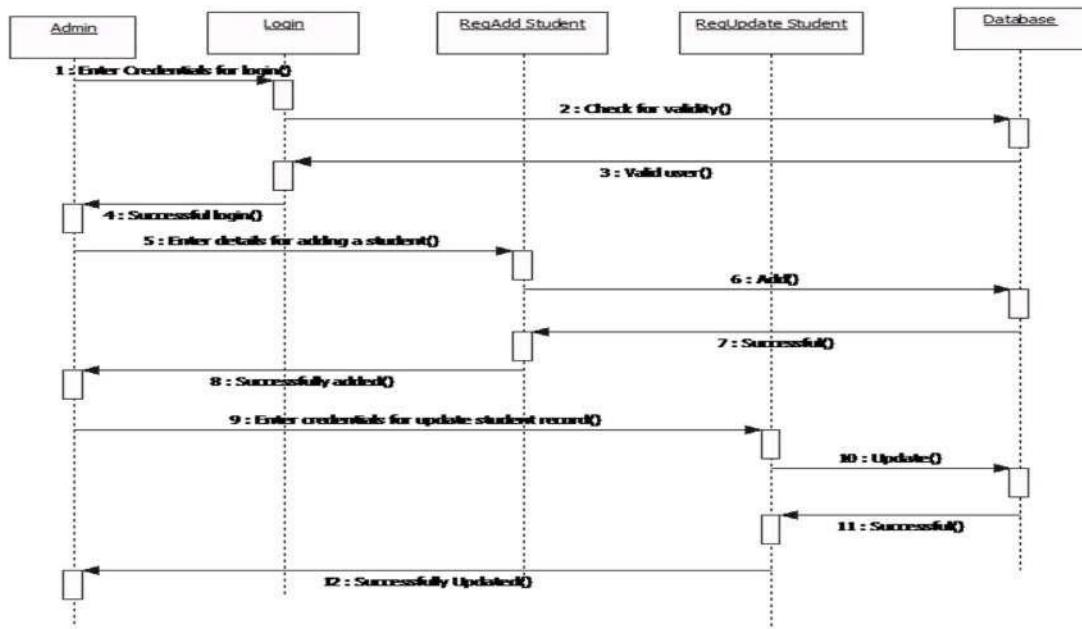
c. Database – is the class name. The operations performed are storing Search and storing the values.

UML INTERACTION DIAGRAM:-

Description:-

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.



DOCUMENTATION OF SEQUENCE DIAGRAM:-

The sequence diagram describes the sequence of steps to show

- The Admin login and registering for Add Student Details.
- The verification done by the interface and sending acknowledgement for registration.
- Searching the database with login and displaying it for maintenance.

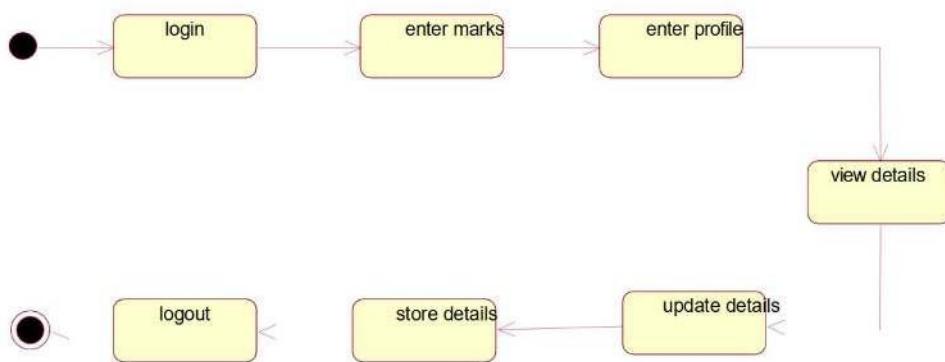
UML STATECHART DIAGRAM:-

Description:-

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

Activity diagram Describes:-

- How activities are coordinator to provide a service.
- The events needed to achieve some operation.
- How events in a single use case relate to one another.



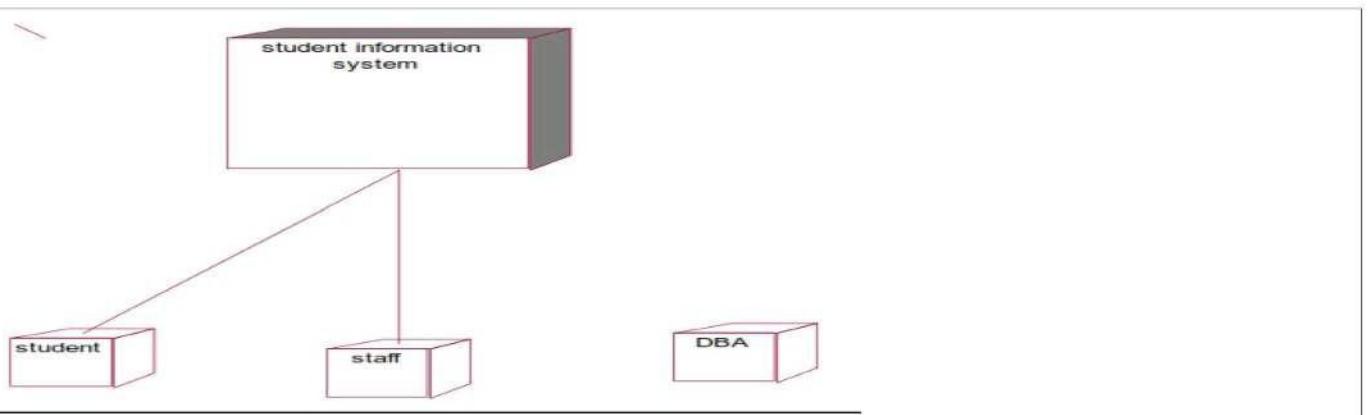
DOCUMENTATION OF STATECHART DIAGRAM:-

This activity diagram flow of stepwise activities performed in recruitment system.

- The student details are Add and stored in database.
- Select the course from the given Course by student.
- Search Profile and Result with login and if data present in the database.
- The searched data is displayed if available and then Log Out.

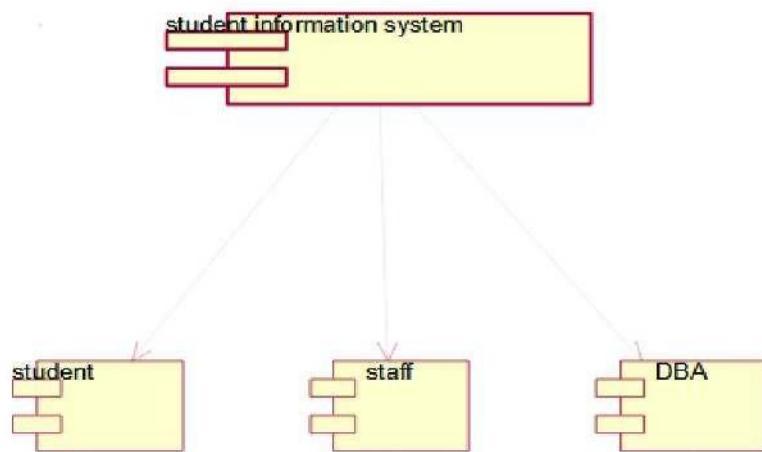
UML DEPLOYMENT DIAGRAM:-

Deployment diagram shows the configuration of runtime processing elements and the software components processes and objects that live in them. Component diagram are used in conjunction with deployment diagram to show how physical modules code are distributed on various hardware platform. The processor node in the system is student information system and the execution environment nodes or device nodes are student, staff and DBA.



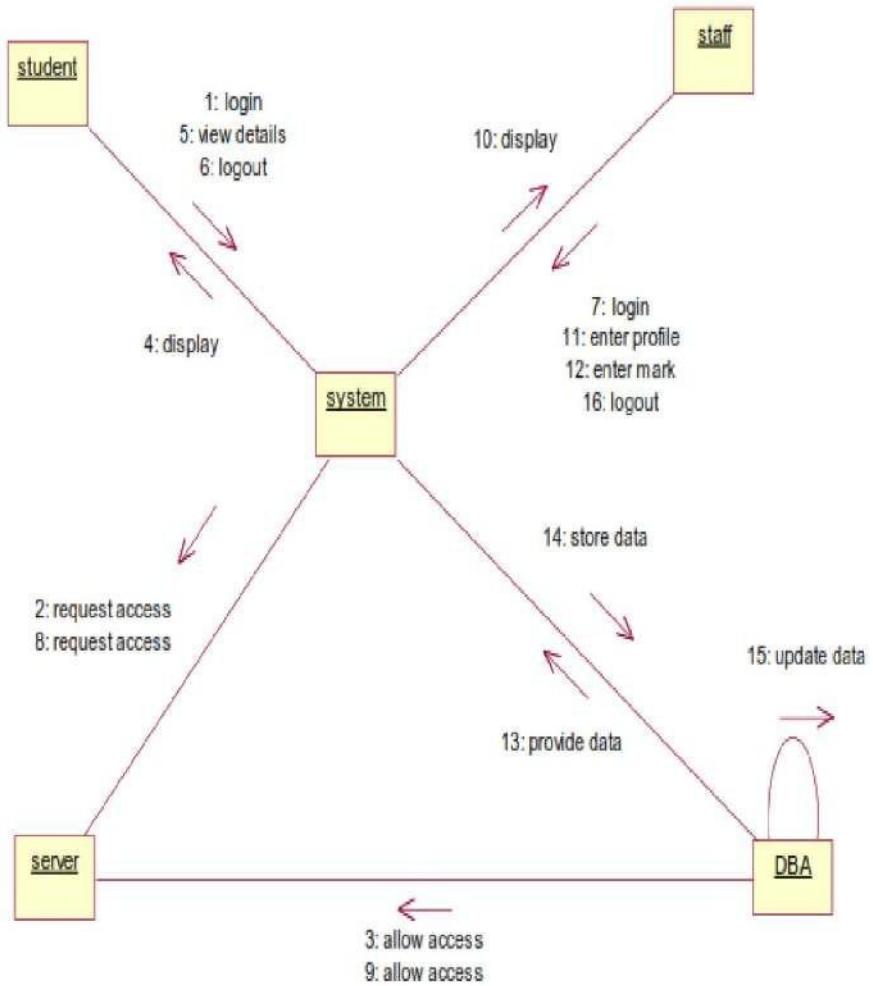
UML COMPONENT DIAGRAM:-

Component diagram carries the major living actors of the system. The component diagram main purpose is to show the structural relationship between components of the system. The main component of the system is student information system and the other components of the system are student, staff and DBA.



UML COLLABORATION DIAGRAM:-

A Collaboration diagram represents the collaboration in which is a set of objects related to achieve a desired outcome. In collaboration, the sequence is indicated by numbering the message several numbering schemes are available. Login, request access, allow access, display, view details, logout, login, request access, allow access, display, enter profile, enter mark, provide data, logout, store data, update data.



PACKAGEDIAGRAM:-

Description:

The Logical architecture is the large-scale organizations of the software classes into packages, subsystems and layers. It is called the logical architecture because there no decision about how these elements are developed across different operating systems processes are across physical computers in a network.

UML TECHNICAL SERVICE LAYER

Register No	Name	Application No	Hall Ticket	Center	Date of Examiner
510615104078	Saravana	1014327	654789	SOC COLLEGE	06-10-2017
501615104088	Surya	1014329	698547	CACHET	05-11-2107
510615104105	Vino	1014320	647895	CIT	05-12-2017

SAMPLE CODE:-

```
import java.util.Vector;
public class students Details {
    public String name;
    public varchar Address;
    public int age;
    public varchar Course;
    public String Department;
    public Int Phone_no;
    public Vector mylogin;
    public Varchar get_date()
```

```
import java.util.Vector;
public class update Details {
    public varchar
Students_details;
    public String Results_details;
    public Vector mylogin;
    public void update_info() {
    }
    public void get_details() {
    }
```

USER INTERFACE LAYER

The screenshot displays the 'STUDENT INFORMATION SYSTEM(SIS)' application. At the top, there are three navigation buttons: 'Home', 'Administration', and 'Registration'. The status bar indicates the user is connected as 'dev' (Group: ADMIN) on Monday, 14th February, 2005, at 3:49pm. The main content area is titled 'Drill-Down Interactive Reports (SIS)'. It includes the University of Uyo logo and a descriptive message about the drill-down reports. Below this, there is a section for 'Personal Data by Registration Number' with a 'Show Registration Details' button. Further down, there is a form for 'Course Registration by Registration Number, Level, Semester and Session', requiring compulsory form fields. The form includes fields for Registration Number, Level (dropdown menu showing '100'), Semester (dropdown menu showing 'FIRST'), and Session (dropdown menu). A 'Start Course Registration' button is located at the bottom of this form.

Result:-

Thus the Student Information System has been done successfully by using Argo-UML.

ADDITIONAL EXPERIMENT

AIM : TO design and implement Electronic Cash Counter system through Class Diagram

1.0 PROBLEM DEFINITION

To develop an automated banking system, which is required to perform the following functions?

- 1.1** The customer logs into the system using card number and pin number. The system checks for validation.
- 1.2** The system queries the customer for the type of account either fixed deposit or credit account. After getting the type of account the system shows the balance left.
- 1.3** The system queries the customer for the transaction type either withdrawal or deposit and the required amount. The user enters the amount and the transaction is carried out

2.0 SRS DOCUMENT FOR AUTOMATED BANKING SYSTEM

2.1 INTRODUCTION

2.1.1 Purpose

- 2.1.1.1** The purpose of this SRS is to describe the requirements involved in developing an Automated Banking System(ABS).
- 2.1.1.2** The intended audience is any person who wants
 - 2.1.1.2.1** To create account.
 - 2.1.1.2.2** To withdraw or deposit either in fixed deposit or credit account.

2.1.2 Scope

- 2.1.2.1** The product is titled Automated Banking System (ABS).
- 2.1.2.2** The product will perform the following tasks
 - 2.1.2.2.1** Allow a new user to create an account, either fixed or credit account by entering the details and by depositing an initial amount.
 - 2.1.2.2.2** Allow the existing user to enter his account details like card number, pin number and account type to view his balance.
 - 2.1.2.2.3** Allow the existing user to deposit an amount by entering the amount to be deposited after the balance had been viewed.
 - 2.1.2.2.4** Allow the existing user to withdraw an amount by entering the amount to be withdrawn after the balance had been viewed.
 - 2.1.2.2.5** The primary benefits expected of the system are: user friendly, continuous connectivity without failure, fault tolerant and involves lesser manpower.

2.1.3 Definitions, Acronyms and Abbreviations

2.1.3.1 ABS: Automated Banking System.

2.1.4 References

2.1.4.1 IEEE standard 830-1998 recommended practice for Software Requirements Specifications-Description.

2.1.4.2 IEEE Software Requirements Specifications Template

http://www.cas.master.ca/~carette/SE3M04/2003/files/srs_template.doc

2.1.5 Overview

2.1.5.1 The SRS contains an analysis of the requirements necessary to help easy design.

2.1.5.2 The overall description provides interface requirements for the Banking system, product perspective, hardware interfaces, software interfaces, communication interface, memory constraints, product functions, user characteristics and other constraints.

2.1.5.3 Succeeding pages illustrate the characteristics of typical naïve users accessing the system along with legal and functional constraints enforced that affect banking system in any fashion.

2.2 THE OVERALL DESCRIPTION

2.2.1 Product perspective

2.2.1.1 Hardware interfaces

2.2.1.1 Hard disk: The database connectivity requires a hardware configuration that is on-line. This makes it necessary to have a fast database system (such as any RDBMS) running on high rpm hard-disk permitting complete data redundancy and backup systems to support the primary goal of reliability.

2.2.1.2 The system must interface with the standard output device, keyboard and mouse to interact with this software.

2.2.1.2 Software interfaces

2.2.1.2.1 Back End: MS Access 2007

2.2.1.2.2 Front End: Microsoft Visual Basic 6.0

2.2.1.3 Operations

2.2.1.3.1 The user can create a new account.

2.2.1.3.2 The existing user can access his account and view his balance by entering his details.

2.2.1.3.2 The user can deposit and withdraw money from his account.

2.2.2 Product Functions

2.2.2.1 Creating a New Account

The user should provide his personal details to facilitate the bankclerk to create a new account. The user should provide:

- 2.2.2.1.1 Customer Name.
- 2.2.2.1.2 Customer address.
- 2.2.2.1.3 Required account type.
- 2.2.2.1.4 Pin Number.
- 2.2.2.1.5 Initial deposit.

2.2.2.2 Operating with created account

The user should be able to operate with his new account after:

- 2.2.2.2.1 Entering card number.
- 2.2.2.2.2 Entering pin number.
- 2.2.2.2.3 Entering the account type, transaction type and amount involved in the transaction.

2.2.3 User characteristics

- 2.2.3.1 The intended users of this software need not have specific knowledge as to what is the internal operation of the system. Thus the end user is at a high level of abstraction that allows easier, faster operation and reduces the knowledge requirement of end user
- 2.2.3.2 The Product is absolutely user friendly, so the intended users can be the naïve users.
- 2.2.3.3 The product does not expect the user to possess any technical background. Any person who knows to use the mouse and the keyboard can successfully use this product.

2.2.4 Constraints:

- 2.2.4.1 At the time of creating the new account, each user gives a pin number and is provided with a unique card number that must be used for further transactions. Hence the user is required to remember or store these numbers carefully.
- 2.2.4.2 At the time of creating the new account, the initial deposit should not be less than the specified amount.

2.3 SPECIFIC REQUIREMENTS

2.3.1 Logical Database Requirements

- 2.3.1.1 The system should contain databases that include all the necessary information for the product to function according to the requirements. These include relations such as Customer Details and Account Details.
- 2.3.1.2 Customer details refer to the customer's name and address. Account details of the customer include the card number, account type, transaction type and the pin number given by the user to be used at the time of the transaction at the bank.

2.4 FRONT – END DESCRIPTION

The front end for the Automated Banking System (ABS) is designed using Microsoft Visual Basic 6.0. The front end contains a user-friendly interface. The first form contains a welcome screen that provides an option for the user to either create a new account or to operate through an existing account. The “create

account" module contains a provision to create a new account after collecting the customer name, address and other details. The card number and pin number of the user is obtained every time there is a transaction. The user is requested to select the required type of transaction and the amount involved in the transaction.

2.5 BACK - END DESCRIPTION

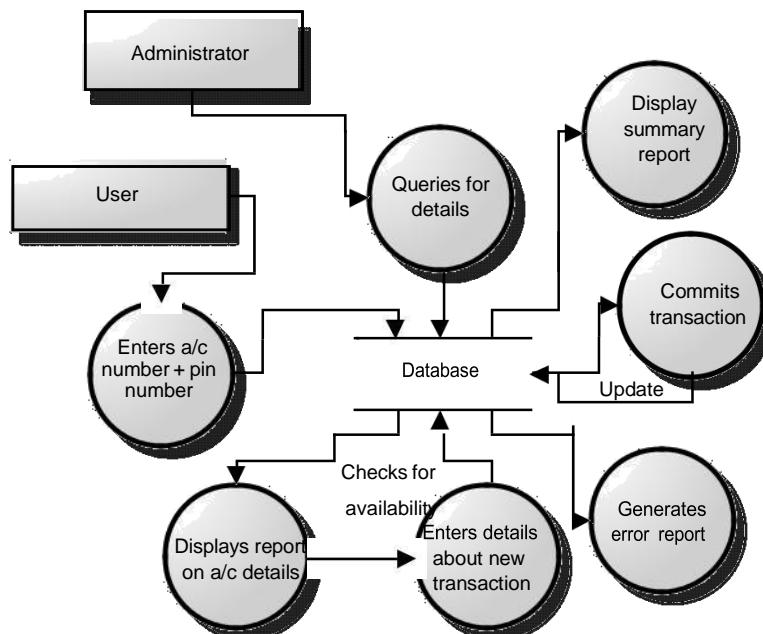
The Automated Banking System (ABS) database contains only one table. It correlates a unique card number, customer name, account type, pin number and the balance.

2.6 DATA STRUCTURES

2.6.1. ACCOUNT DETAILS

FIELD NAME	TYP E	CONSTRAINTS
NAME	TEXT	
ACC_NO	AUTONUMBER	NOT NULL
AGE	NUMBER	
GENDER	TEXT	
EMAIL	TEXT	
PHONE_NO	NUMBER	
PASSWORD	TEXT	
BALANCE	NUMBER	

2.7 DATA FLOW DIAGRAM:



PROCEDURE:

The purpose of the class diagram is to model the static view of an application. The class diagrams are the only diagrams which can be directly mapped with object oriented languages and thus widely used at the time of construction. The UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application but class diagram is a bit different. So it is the most popular UML diagram in the coder community. So the purpose of the class diagram can be summarized as:

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

Class diagrams commonly contain the following things

- Classes
- Interfaces
- Collaborations
- Dependency, generalization and association relationships

Steps :

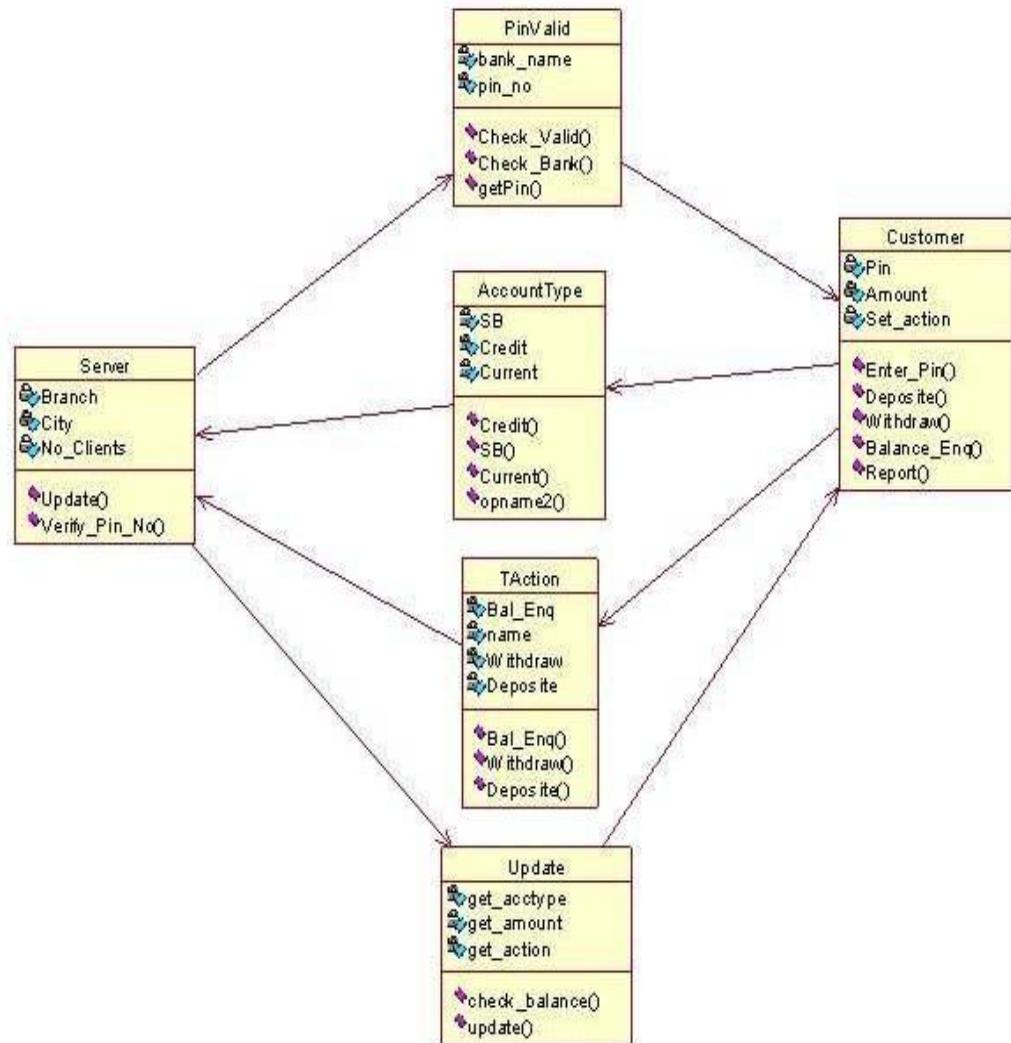
Step1: First Classes are created.

Step2: Named as Pin Valid, Account Type,

Transaction, Update, Server, Customer classes are created.

Step3: Appropriate relationships are provided between them as association.

DIAGRAM:



Inferences

- :
- 1. Understand the concept of classes
- 2. Identify classes and attributes and operations for a class
- 3. Model the class diagram for the system

Applications

- :
- Online transaction**
- Online banking**

B) NAME OF EXPERIMENT: Use case diagram for ATM System.

AIM: To design and implement ATM System through Use case Diagram.

Purpose:

The purpose of use case diagram is to capture the dynamic aspect of a system. Because other four diagrams (activity, sequence, collaboration and State chart) are also having the same purpose. So we will look into some specific purpose which will distinguish it from other four diagrams. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analysed to gather its functionalities use cases are prepared and actors are identified.

So in brief, the purposes of use case diagrams can be as follows:

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

Procedure:

Step1: First an Actor is Created and named as User/Customer.

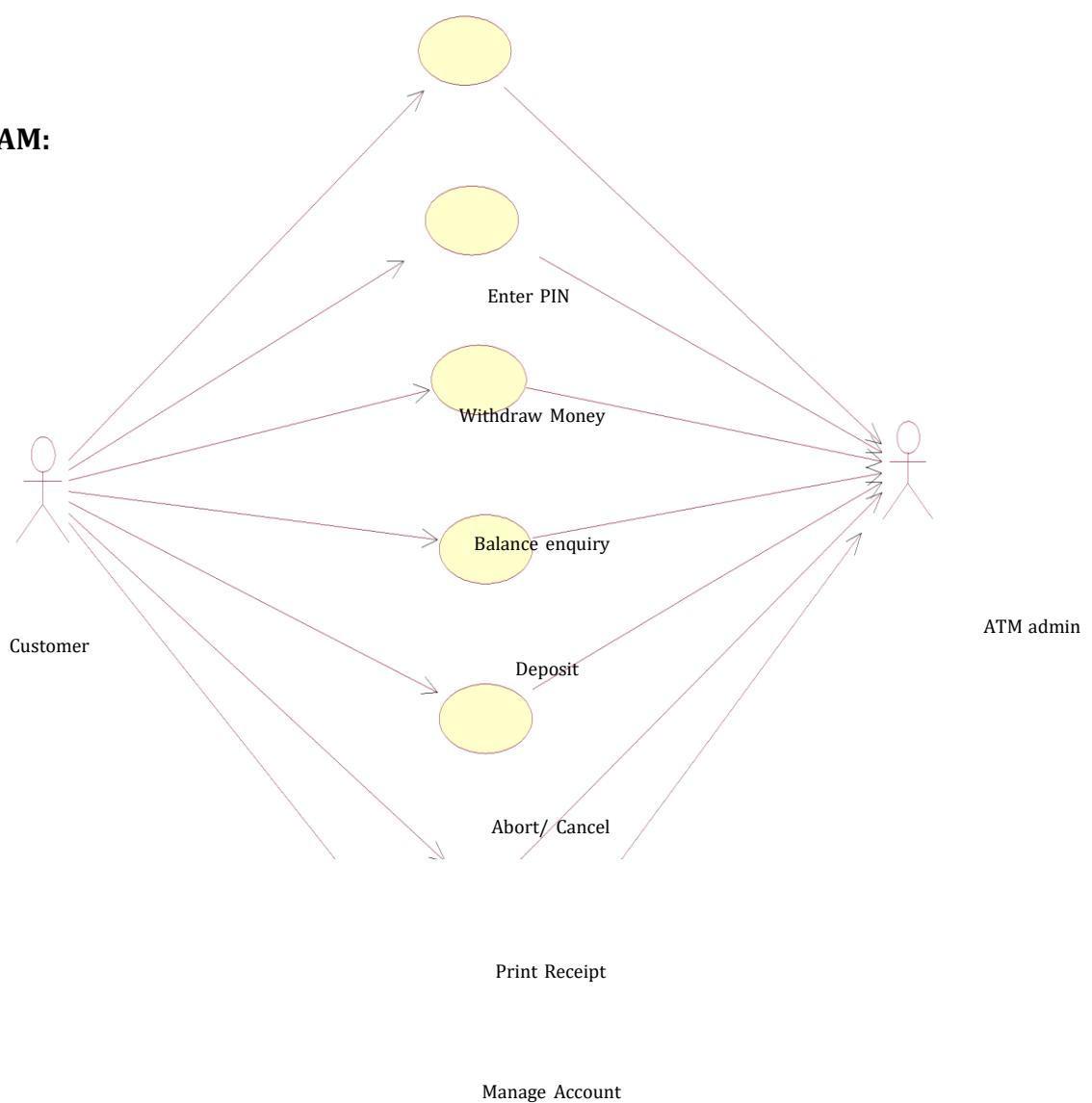
Step2: Secondly a system is created for ATM.

Step3: A use case Enter PIN, Withdraw money is created and connected with user as association relationship.

Step4: Similarly various use cases like Deposit money, Balance Enquiry, Manage Account etc. are created and appropriate relationships are associated with each of them.

Inferences:

1. Identification of use cases.
2. Identification of actors.

DIAGRAM:**Withdrawal Use Case**

A withdrawal transaction asks the customer to choose a type of account to withdraw from (e.g. checking) from a menu of possible accounts, and to choose an amount from a menu of possible amounts. The system verifies that it has sufficient money on hand to satisfy the request before sending the transaction to the bank. (If not, the customer is informed and asked to enter a different amount.) If the transaction is approved by the bank, the appropriate amount of cash is dispensed by the machine before it issues a receipt. A withdrawal transaction can be cancelled by the customer pressing the Cancel key any time prior to choosing the amount.

INTERACTION DIAGRAMS

We have two types of interaction diagrams in UML. One is sequence diagram and the other is a collaboration diagram. The sequence diagram captures the time sequence of message flow from one object to another and the collaboration diagram describes the organization of objects in a system taking part in the message flow.

So the following things are to be identified clearly before drawing the interaction diagram:

1. Objects taking part in the interaction.
2. Message flows among the objects.
3. The sequence in which the messages are flowing.
4. Object organization.

Purpose:

1. To capture dynamic behaviour of a system.
2. To describe the message flow in the system.
3. To describe structural organization of the objects.
4. To describe interaction among objects.

Contents of a Sequence Diagram

**Objects
Focus of control
Messages
Life line**

Contents of a Collaboration Diagram

**Objects
Links
Message
s**

C) NAME OF EXPERIMENT: Sequence diagram for ATM System.
AIM: To design and implement ATM System through Sequence Diagram.

Procedure:-

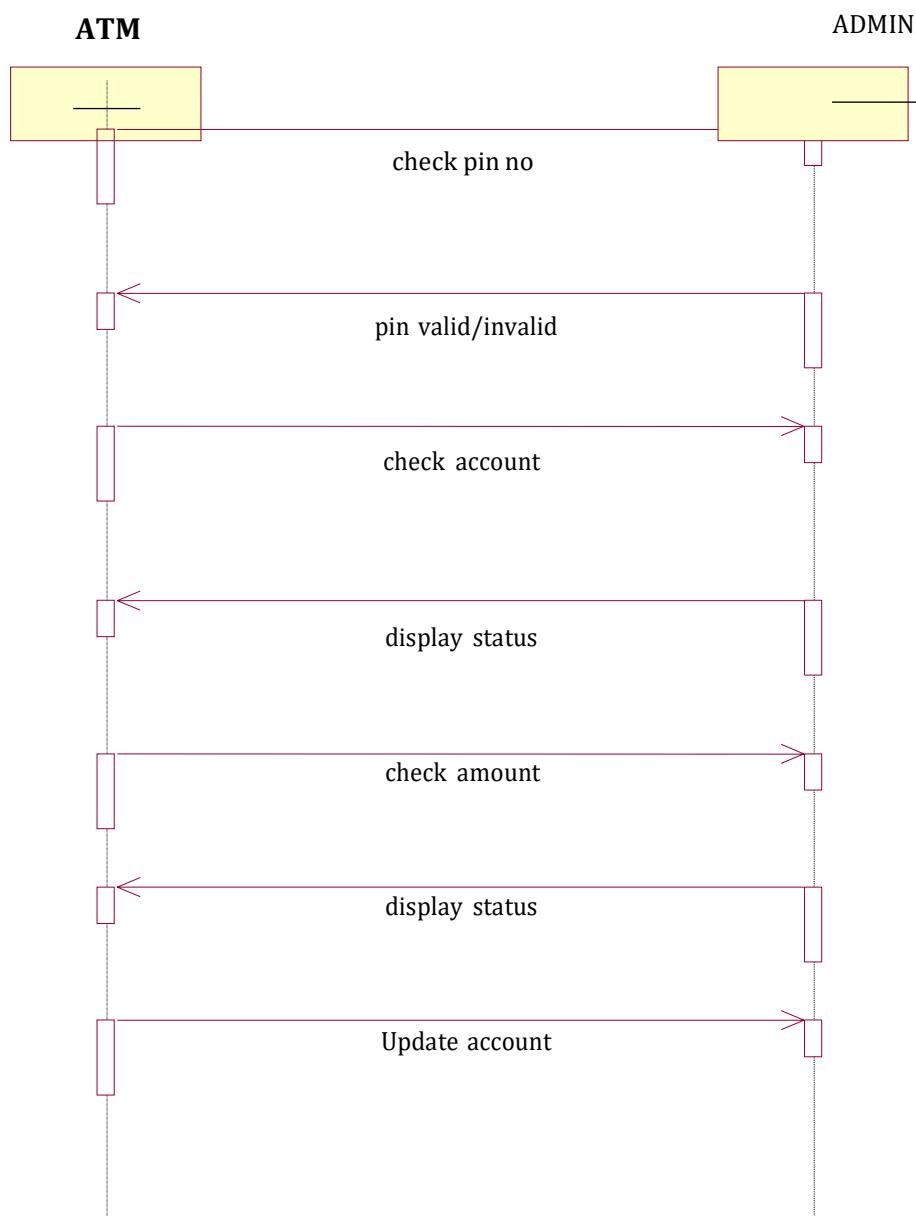
Step1: First An actor is created and named as user.

Step2: Secondly an object is created for Atm.

Step3: Timelines and lifelines are created automatically for them.

Step4: In sequence diagram interaction is done through time ordering of messages. So appropriate messages are passed between user and ATM is as shown in the figure.

DIAGRAM:



D) NAME OF EXPERIMENT: collaboration for ATM System.

AIM: To design and implement ATM System through Collaboration diagram.

Procedure:-

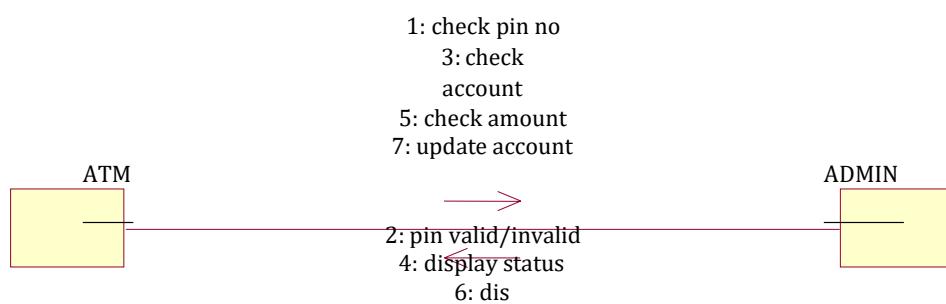
Step1: First an actor is created and named as user.

Step2: Secondly an object is created for ATM.

Step3: In collaboration diagram interaction is done through organization.

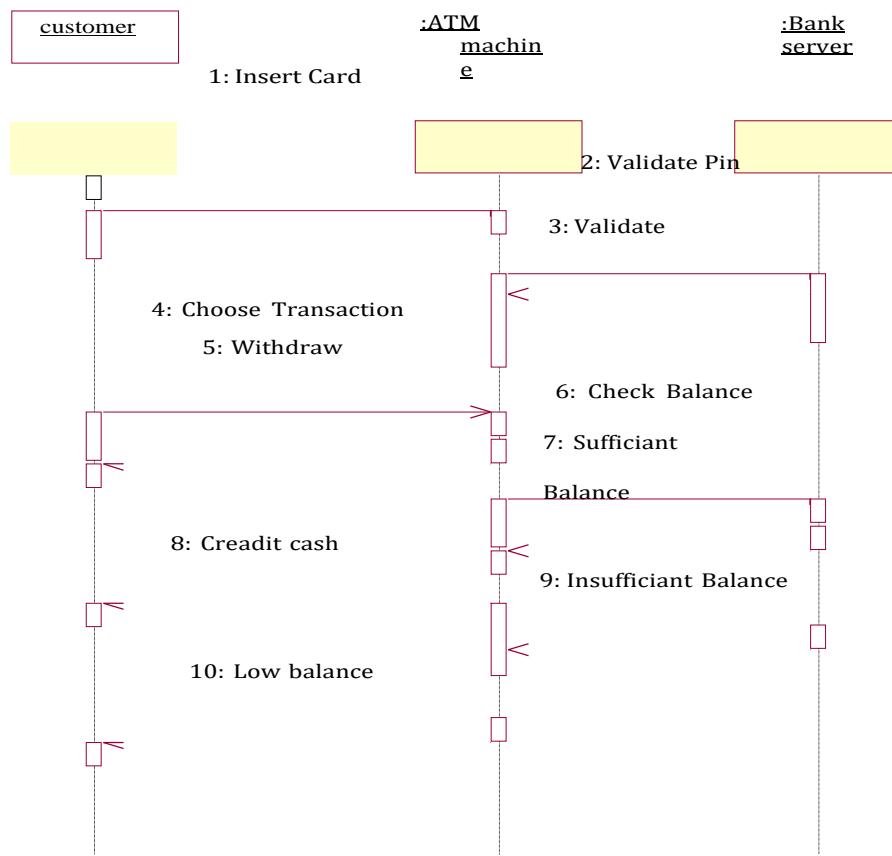
Step4: So appropriate messages are passed between user and ATM as shown in the figure.

DIAGRAM:

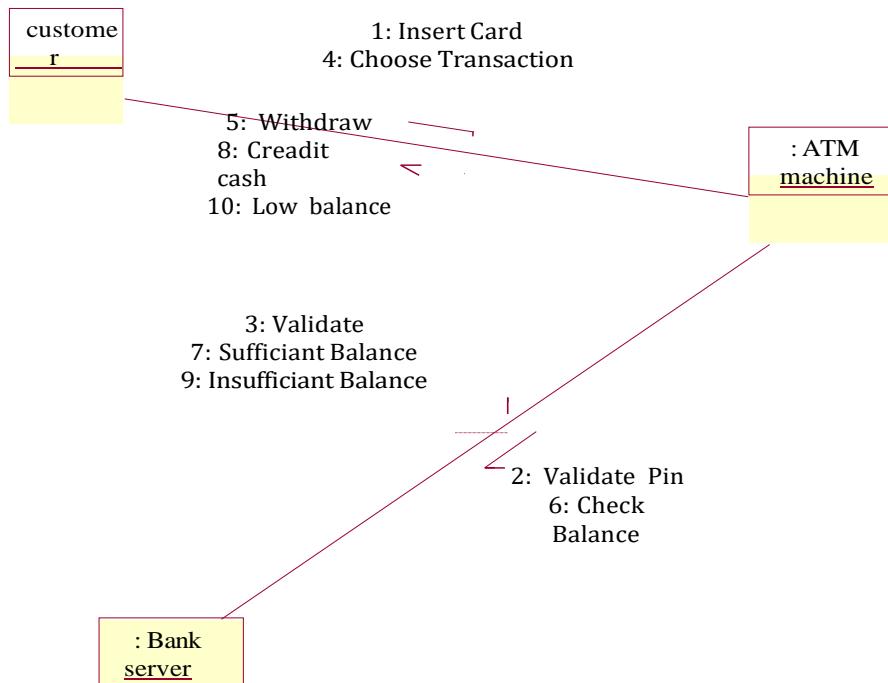


WITHDRAW:

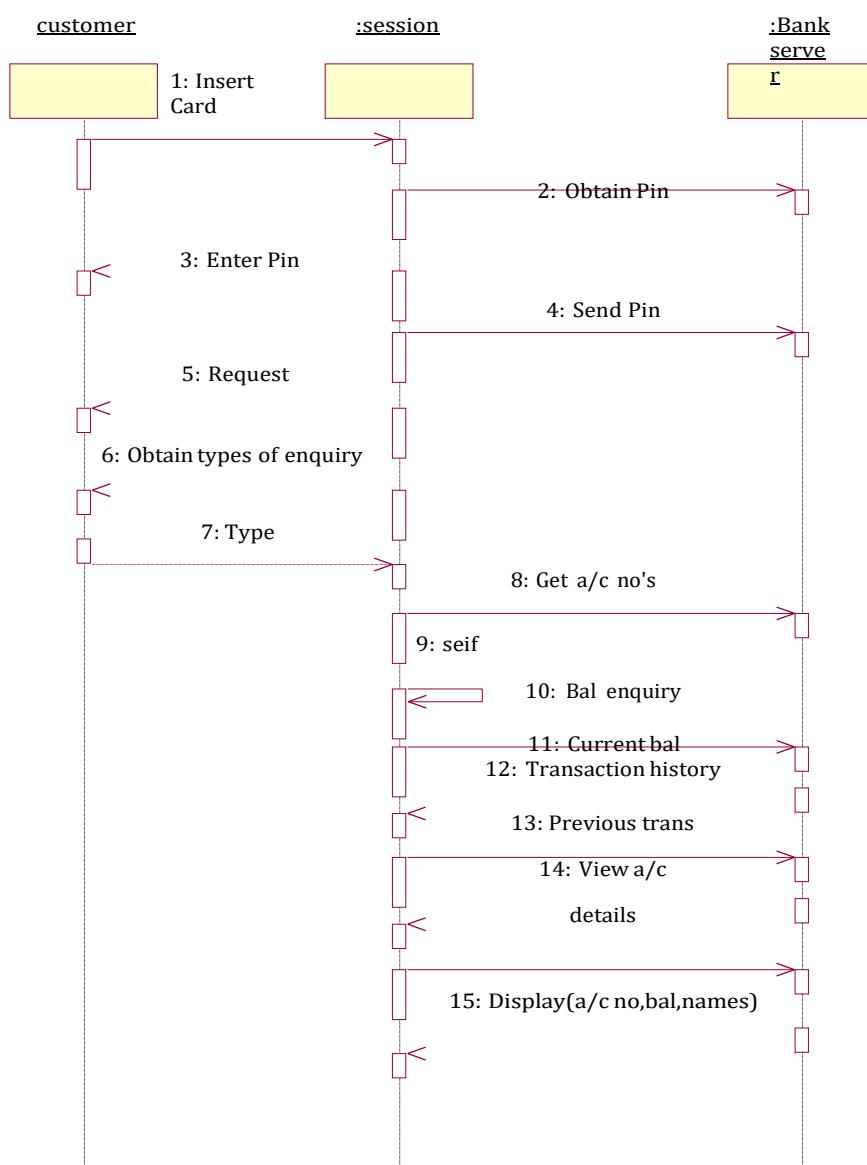
SEQUENCE DIAGRAM



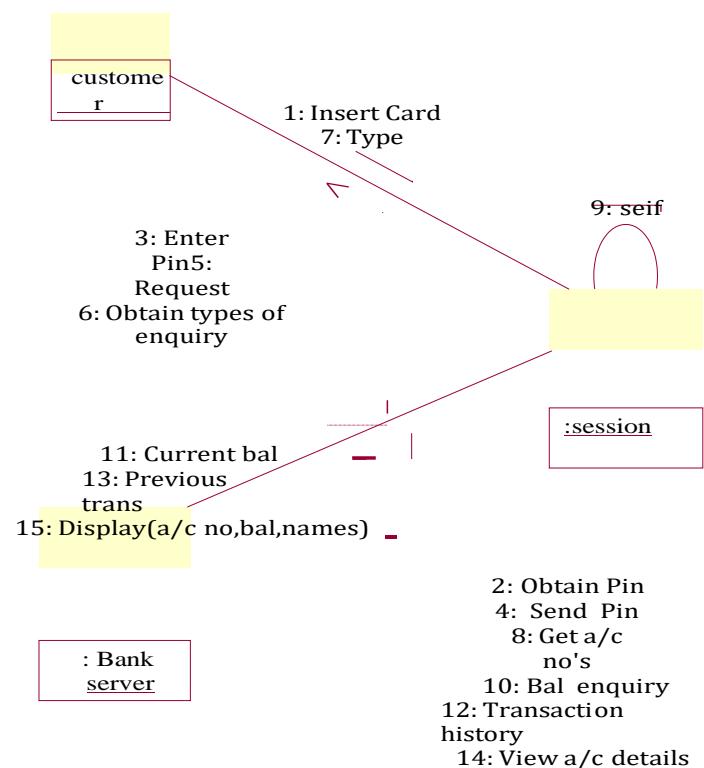
COLLABORATION DIAGRAM



ENQUIRY:
SEQUENCE DIAGRAM:

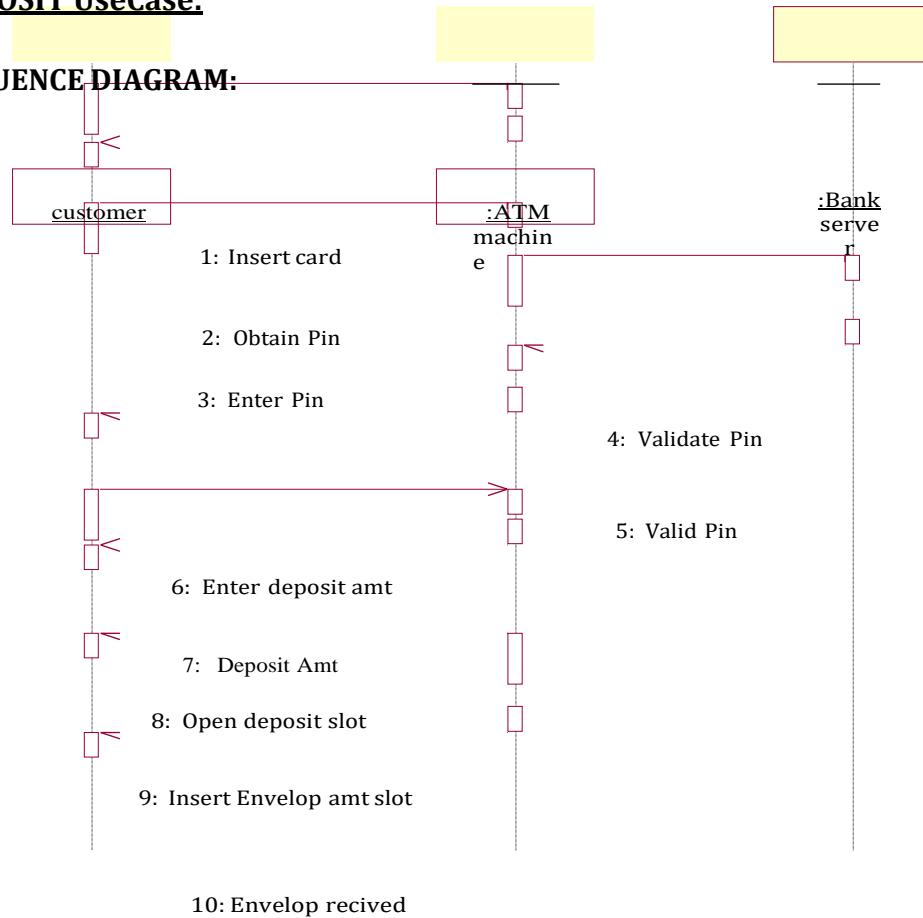


COLLABORATION DIAGRAM:

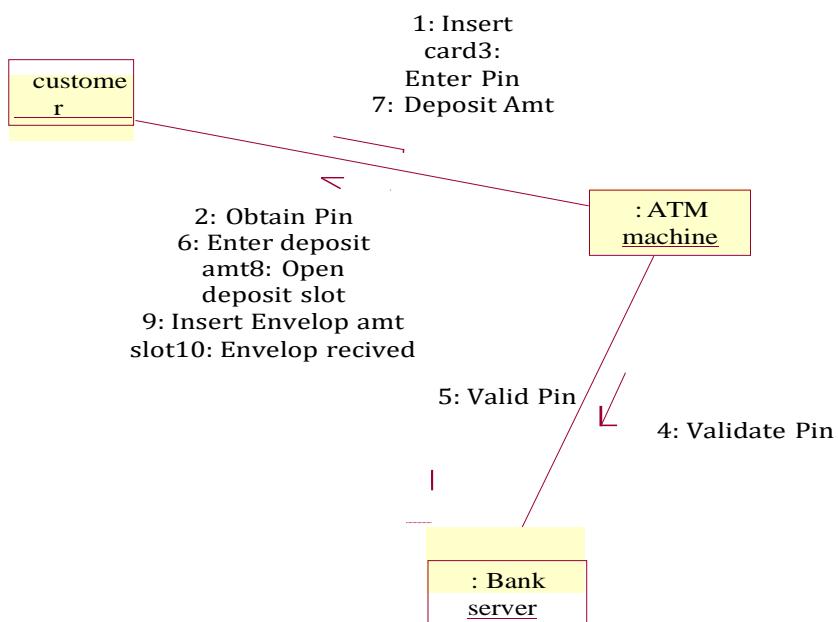


DEPOSIT UseCase:

SEQUENCE DIAGRAM:



COLLABARATION DIAGRAM:



STATECHART Diagram :

State Chart diagram is used to model dynamic nature of a system. They define different states of an object during its lifetime. And these states are changed by events. 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. But the main purpose is to model reactive system.

Contents

- Simply state and composite states
 - Transitions, including events and actions

E) NAME OF EXPERIMENT: State chart diagram for ATM System.

AIM: To design and implement ATM System through State Chart diagram.

Purpose:

Following are the main purposes of using State chart diagrams:

1. To model dynamic aspect of a system.
2. To model life time of a reactive system.
3. To describe different states of an object during its life time.
4. Defines a state machine to model states of an object.

Procedure:-

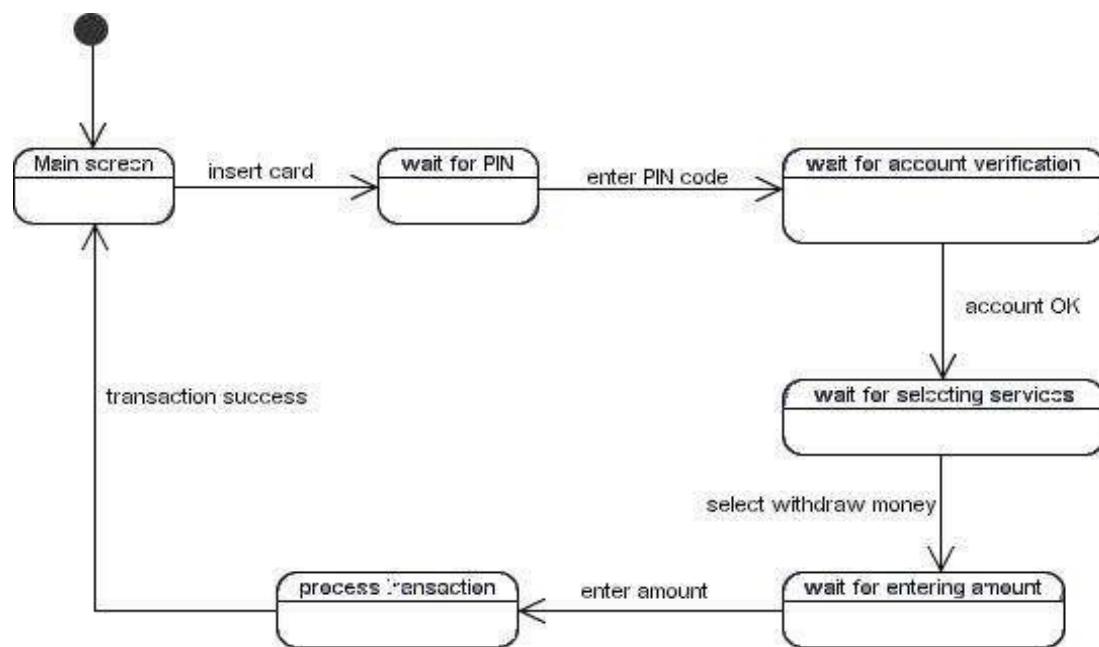
Step1: First after initial state control undergoes transition to ATM screen.

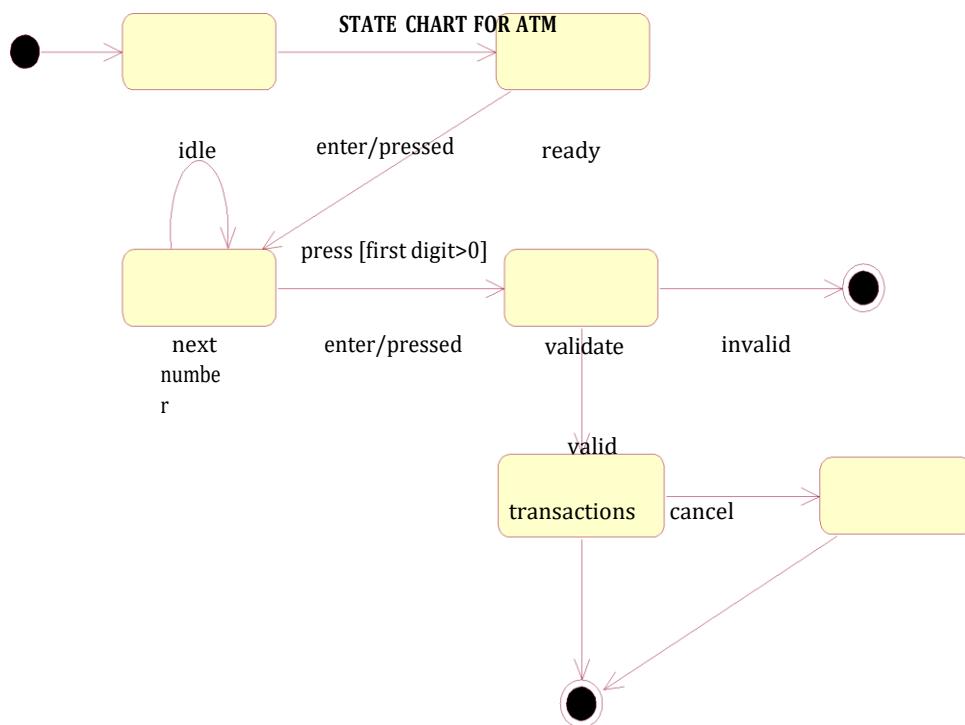
Step2: After inserting card it goes to the state wait for pin.

Step3: After entering pin it goes to the state account verification.

Step4: In this way it undergoes transitions to various states and finally reaches the ATM screen stateas shown in the fig.

DIAGRAM:





Cancel pressed

F) NAME OF EXPERIMENT: Activity diagram for ATM System.

AIM: To design and implement ATM System through Activity Diagram.

THEORY: An activity diagram shows the flow from activity to activity .An activity is an ongoing nonatomic execution within a state machine .Activities ultimately results in some action, which is made up of executable atomic computations. We can use these diagrams to model the dynamic aspects of a system.

Activity diagram is basically a flow chart to represent the flow from one activity to another . The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagrams deals with all type of flow by using elements like fork, join etc.

Contents

Initial/Final State, Activity, Fork & Join, Branch, Swim lanes

Fork

A fork represents the splitting of a single flow of control into two or more concurrent Flow of control.A fork may have one incoming transition and two or more outgoing transitions, each of which represents an independent flow of control. Below fork the activities associated with each of these paths continues in parallel.

Join

A join represents the synchronization of two or more concurrent flows of control. A join may have two or more incoming transition and one outgoing transition. Above the join the activities associated with each of these paths continues in parallel.

Branching

A branch specifies alternate paths takes based on some Boolean expression Branch is represented by diamond Branch may have one incoming transition and two or more outgoing one on each outgoing transition, you place a Boolean expression shouldn't overlap but they should cover all possibilities.

Swim lane:

Swim lanes are useful when we model workflows of business processes to partition the activity states on an activity diagram into groups. Each group representing the business organization responsible for those activities, these groups are called Swim lanes .

Procedure:-

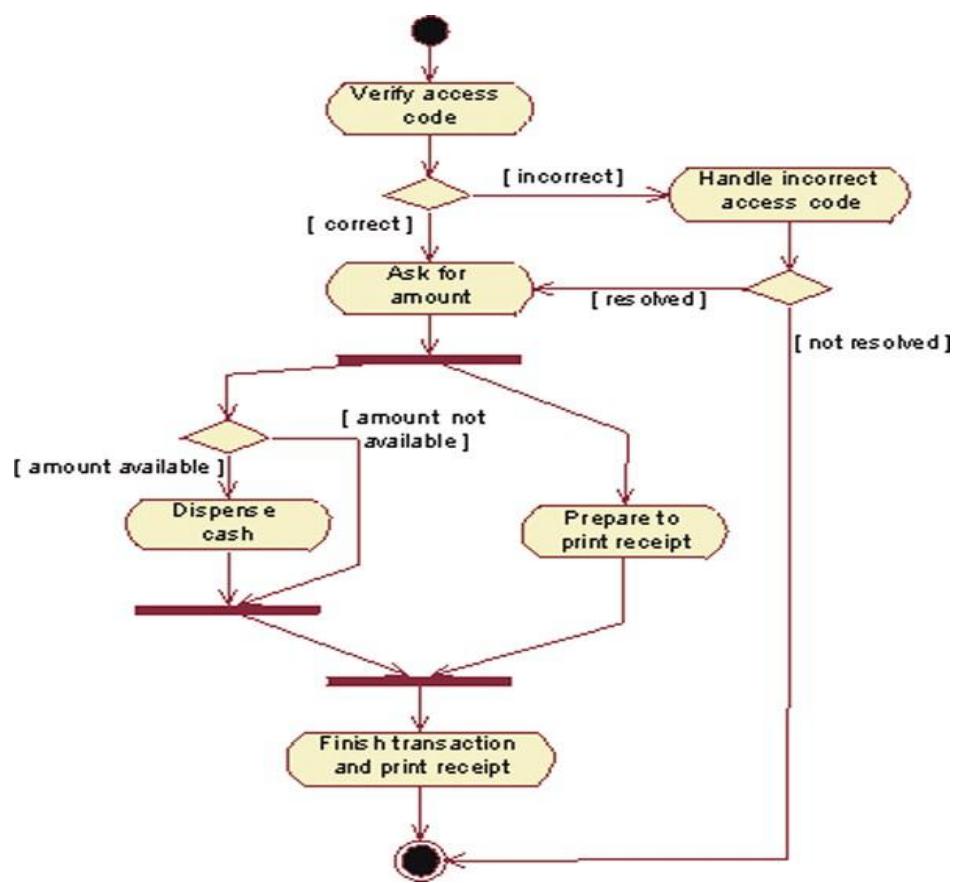
Step1: First initial state is created.

Step2: After that it goes to the action state insert card.

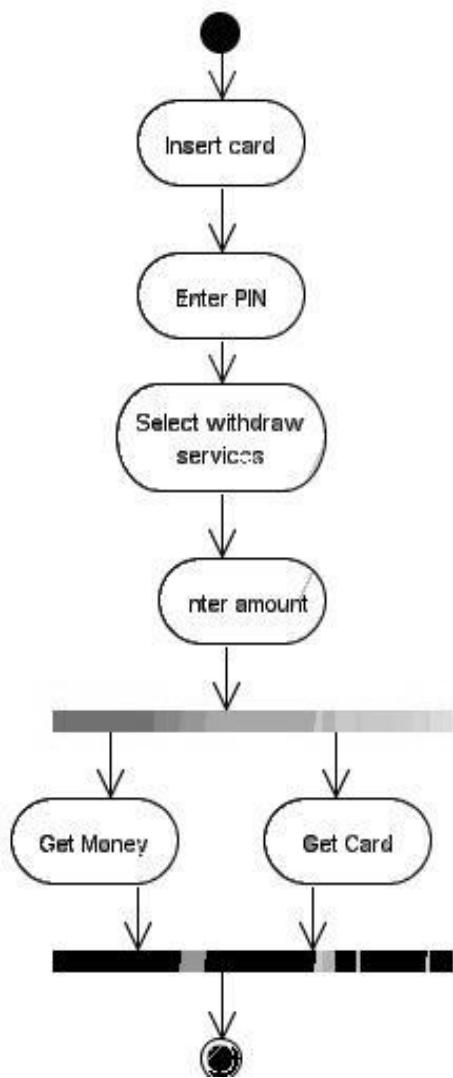
Step3: Next it undergoes transition to the state enter pin

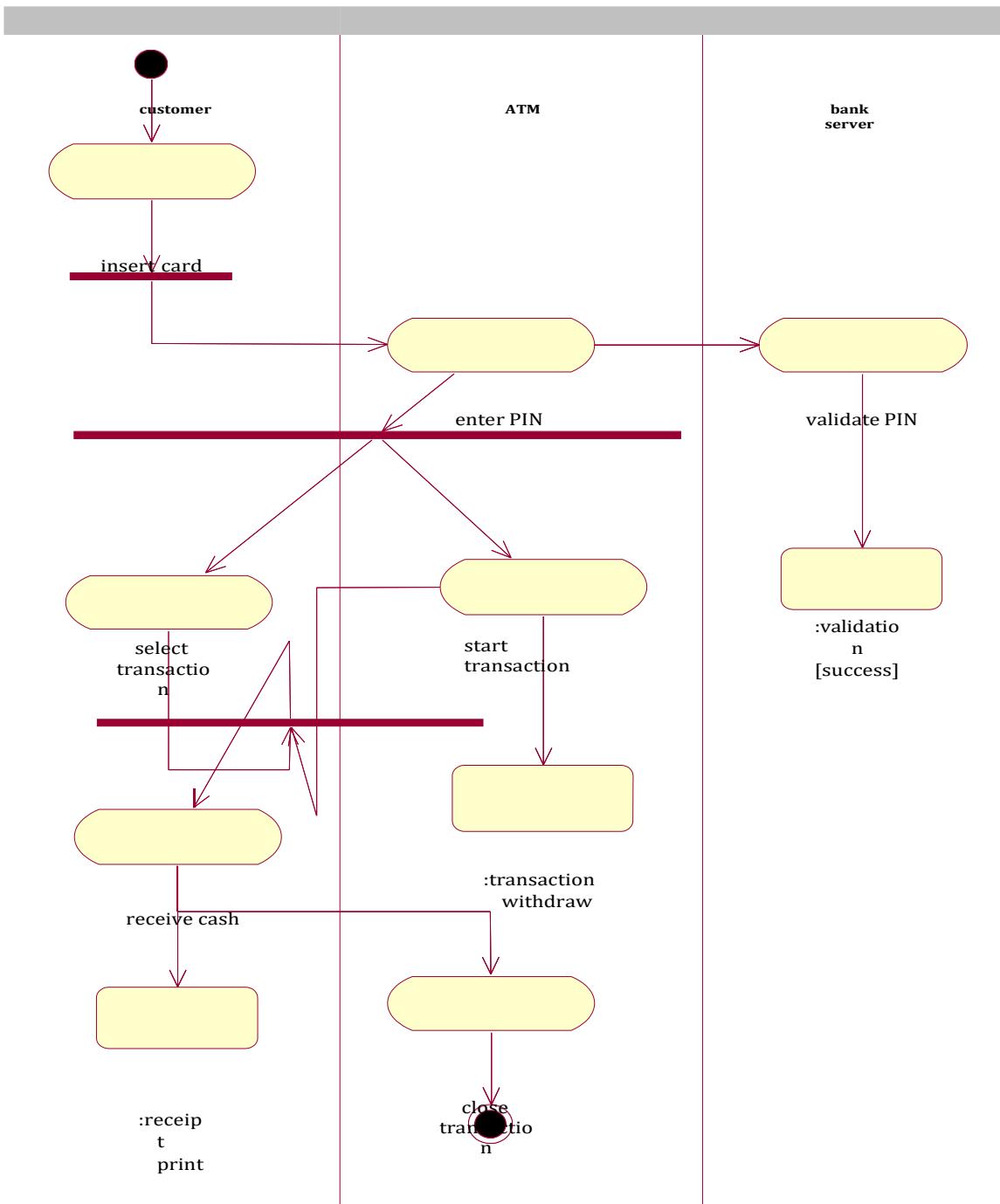
Step4: In this way it undergoes transitions to the various

states.**Step5:** Use forking and joining wherever necessary.

DIAGRAM:**Activity diagram for Transactions:**

Activity diagram for Withdraw





Inferences:

1. Identify the action states of the objects .
2. Understand the transitions and events for various objects.

G) NAME OF EXPERIMENT: Component diagram for ATM System.

AIM: To design and implement Component diagram for ATM

System.THEORY:

Component diagrams are used to model physical aspects of a system. Physical aspects are the elements like executable, libraries, files, documents etc. which resides in a node. So component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

Purpose:

Component diagrams can be described as a static implementation view of a system. Static implementation represents the organization of the components at a particular moment. A single component diagram cannot represent the entire system but a collection of diagrams are used to represent the whole.

Before drawing a component diagram the following artifacts are to be identified clearly:

- Files used in the system.
- Libraries and other artifacts relevant to the application.
- Relationships among the artifacts.
- Now after identifying the artifacts the following points needs to be followed:
 - Use a meaningful name to identify the component for which the diagram is to be drawn.
 - Prepare a mental layout before producing using tools.
 - Use notes for clarifying important points.

Contents

Components, Interfaces, Relationships

Procedure:-

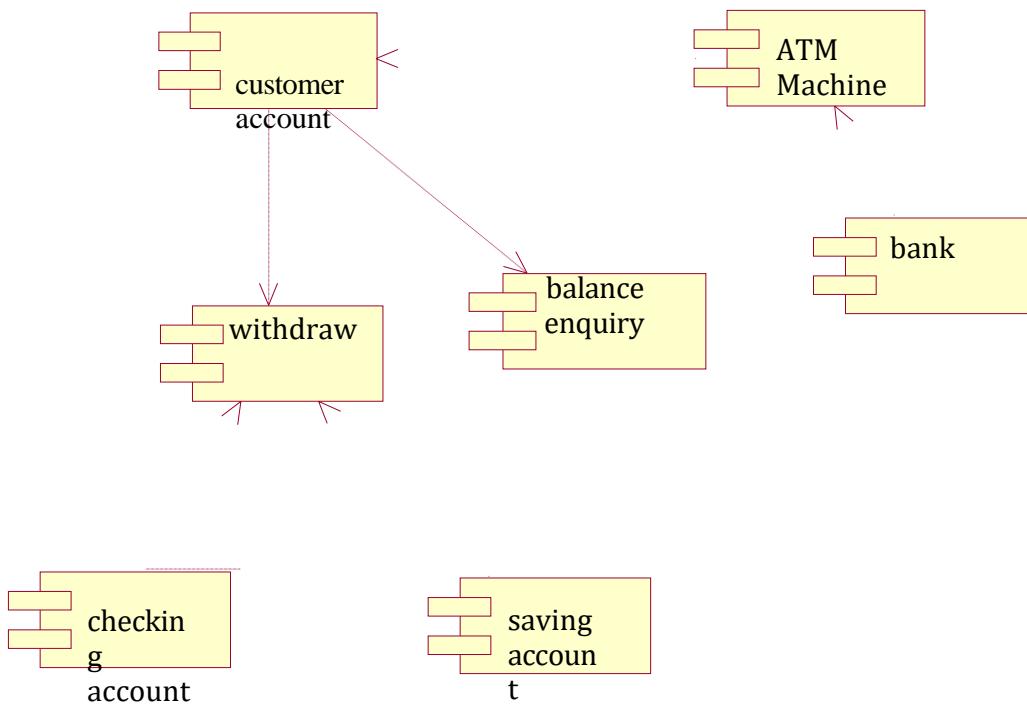
Step1: First user component is created.

Step2: ATM system package is created.

Step3: In it various components such as withdraw money, deposit money, check balance, transfer money etc. are created.

Step4: Association relationship is established between user and other components.

COMPONENT DIAGRAM:



NAME OF EXPERIMENT: Deployment diagram for ATM System

Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed. So deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams are used for describing the hardware components where software components are deployed. Component diagrams and deployment diagrams are closely related. Component diagrams are used to describe the components and deployment diagrams shows how they are deployed in hardware.

Contents: Nodes, Dependency & Association relationships

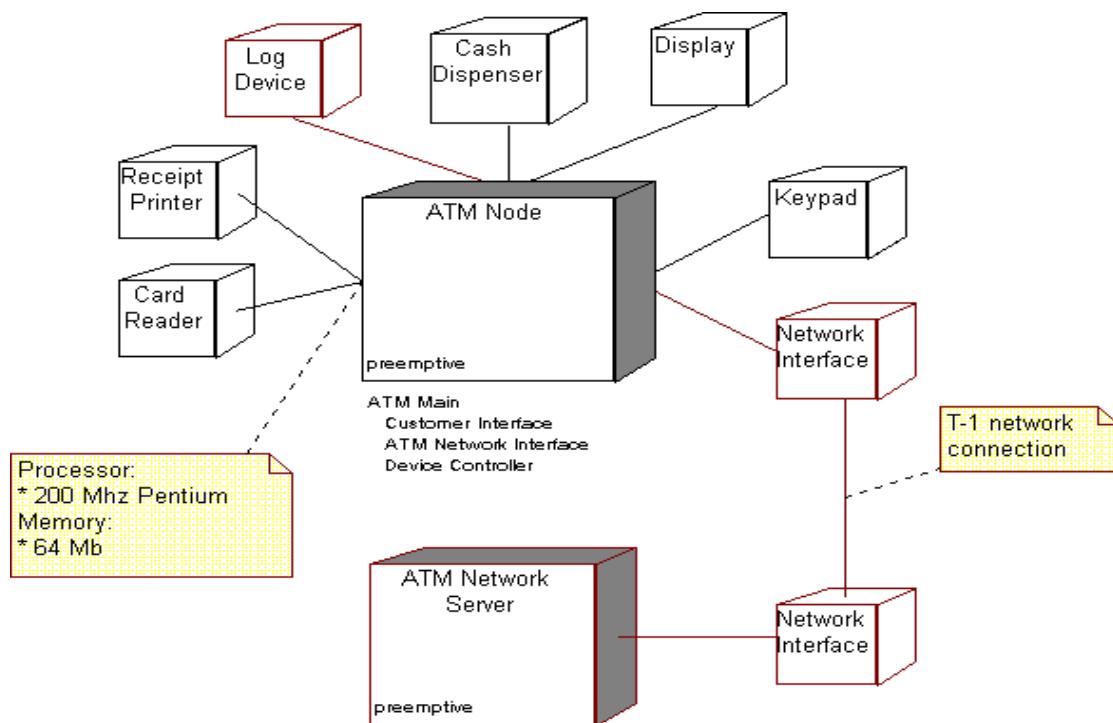
Procedure:-

Step1: First user node is created

Step2: various nodes withdraw money, deposit money, and check balance, transfer money etc. are created.

Step4: Association relationship is established between user and other nodes.

Step5: Dependency is established between deposit money and check balance.



AME OF EXPERIMENT: Test Design for ATM System
AIM: To design and Write the test cases for ATM System

PURPOSE:
Test Design is creating a set of inputs for given software that will provide a set of expected outputs. The idea is to ensure that the system is working good enough and it can be released with as few Problems as possible for the average user

Test Design for ATM Machine

1. Machine is accepting ATM card
2. Machine is rejecting expired card
3. Successful entry of PIN number
4. Unsuccessful operation due to enter wrong PIN number 3 times
5. Successful selection of language
6. Successful selection of account type
7. Unsuccessful operation due to invalid account type
8. Successful selection of amount to be withdraw
9. Successful withdrawal.
10. Expected message due to amount is greater than day limit
11. Unsuccessful withdraw operation due to lack of money in ATM
12. Expected message
Due to amount to withdraw is greater than possible balance.
13. Unsuccessful withdraw operation due to click cancel after insert card

Write the Test Cases for ATM

Test Cases For Atm Machine

1. Successful inspection of ATM card
2. Un successful operation due to insert card in wrong angle
3. Un successful operation due to invalid account Ex:other bank card or time expiredCard
4. Successful entry of PIN number
- 5.un successful operation due to enter wrong PIN number 3times
6. Successful selection of language
7. Successful selection of account type
8. un successful operation due to invalid account type
10. Successful selection of withdrawl operation
11. Successful selection of amount to be withdrawl
12. Successful withdrawl operation
13. Unsuccessful withdrawl operation due to wrong denominations
14. Unsuccessful withdrawl operation due to amount is greater than day limit
15. Unsuccessful withdrawl operation due to lack of money in ATM
16. Unsuccessful withdrawl operation due to amount is greater than possible balance

17. Unsuccessful withdrawl operation due to transactions is greater than day limit
18. Unsuccessful withdrawl operation due to click cancel after insert card
19. Unsuccessful withdrawl operation due to click cancel after insert card & pin number
20. Unsuccessful withdrawl operation due to click cancel after insert card , pin number

2.8 RESULT

Thus the requirements involved in developing an Automated Banking System was completed successfully