

Assignment ⇒ 01Use - case Diagram in Rational Rose Software.

1.) Hospital Reception is a module of a Hospital management System. Here, the receptionist is the leading actor. The receptionist interacts with multiple use cases like schedule patient appointment, patient admission in the hospital, patient registration etc. These cases are related to each other. Patient admission can be 2 types - outpatient and inpatient hospital admission. Inpatient hospital admitted patient will be allocated bed. The receptionist is also responsible for file insurance forms / claims and files medical reports. Draw the use-case diagram for the particular scenario.

⇒ Actor : Here primary actor is receptionist.

Use Cases : Here Use cases are :-

- a.) schedule patient appointment
- b.) Schedule patient hospital admission
- c.) Patient registration
- d.) Patient hospital admission
- e.) File insurance
- f.) File medical reports

- g) Outpatient admission
- h) Inpatient admission

Use Case diagrams in Rational Rose Software :-

A Use Case Diagram is a vital tool in system design, it provides a visual representation of how users interact with a system. It serves as a ~~blueprint~~ for understanding the functional requirements of a system from a user's perspective.

Use Case Diagram Notations :-

- 1) Actors :- Actors are external entities that interact with the system. These can include users, other systems or hardware devices.
- 2) Use Cases :- Use cases are like scenes in the play. They represent specific things your system can do. In the online shopping system, examples of use cases could be "place Order", "Track Delivery" or "Update product info". Use case are represented by ovals.
- 3) System Boundary :- The system boundary is a visual representation of the scope or limits of the system you're modelling. It defines what is inside the system and what is outside.

The boundary helps to establish a clean distinction between the elements that are part of the system and those that are external to it. The system boundary is typically represented by a rectangular box that surrounds all the use cases of the system.

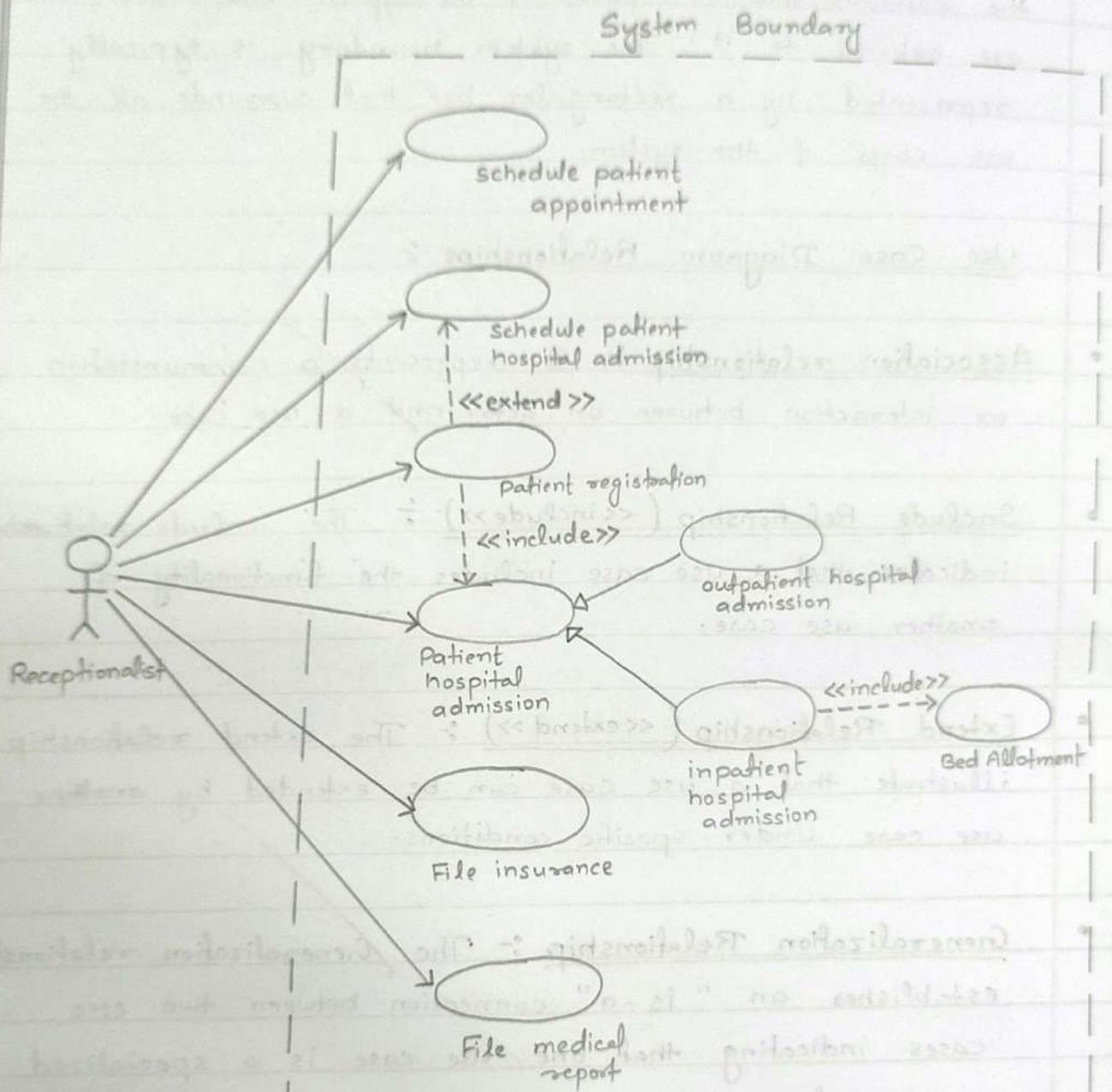
### Use Case Diagram Relationships :-

Association relationship :- It represents a communication or interaction between an actor and a use case.

Include Relationship ( <<include>> ) :- The include relationship indicates that a use case includes the functionality of another use case.

Extend Relationship ( <<extend>> ) :- The extend relationship illustrates that a use case can be extended by another use case under specific conditions.

Generalization Relationship :- The generalization relationship establishes an "is-a" connection between two use cases indicating that one use case is a specialized version of another.



label?.

 → Actor  
 → Use Case  
 ----- → include relation  
 <----- extend relation  
 —→ Generalised relation

### Use - Case Component :

#### Component

Actor

Use - Case

System

#### Relationships

#### Symbol

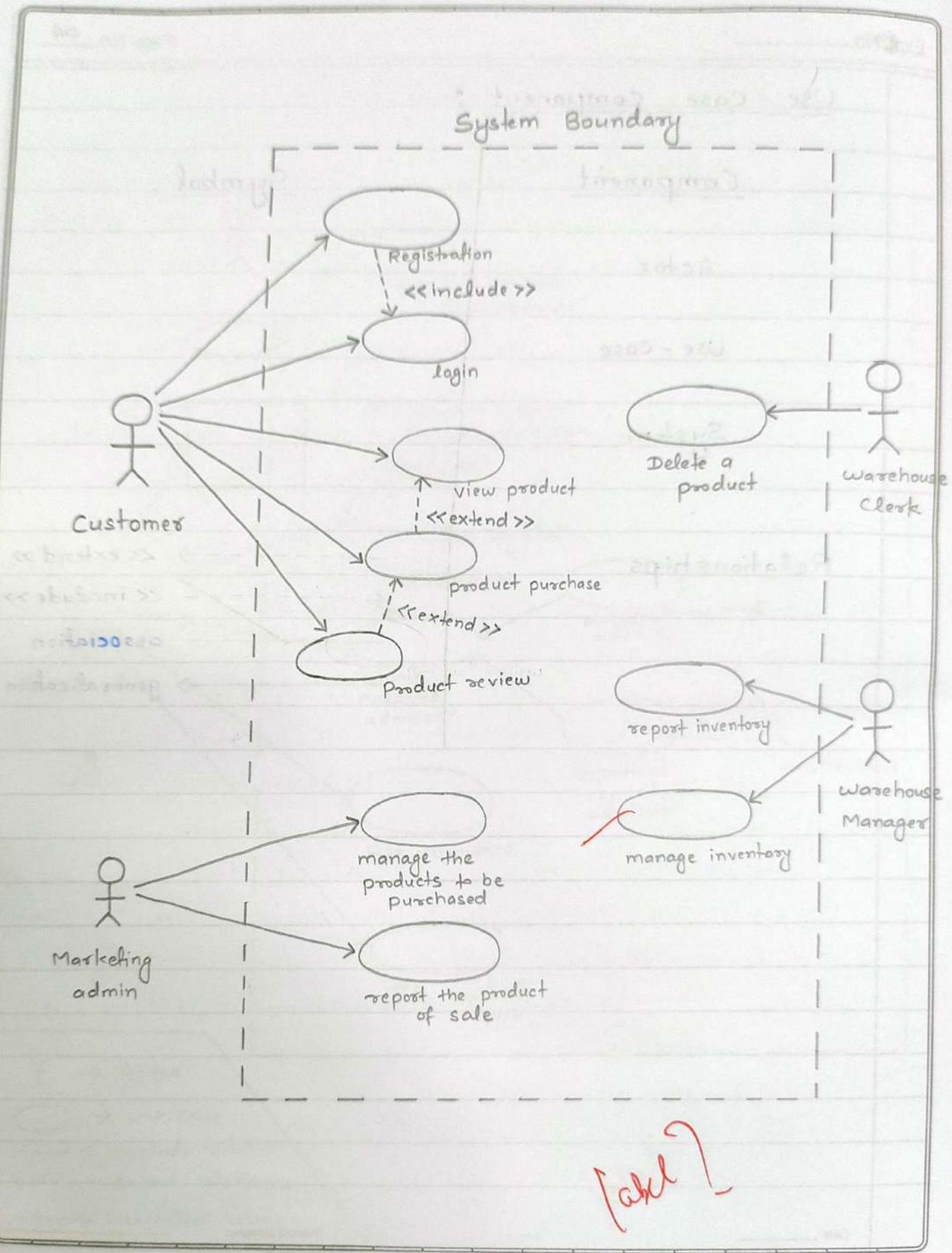


-----> << extend >>

<-----> << include >>

—> association

—> generalization



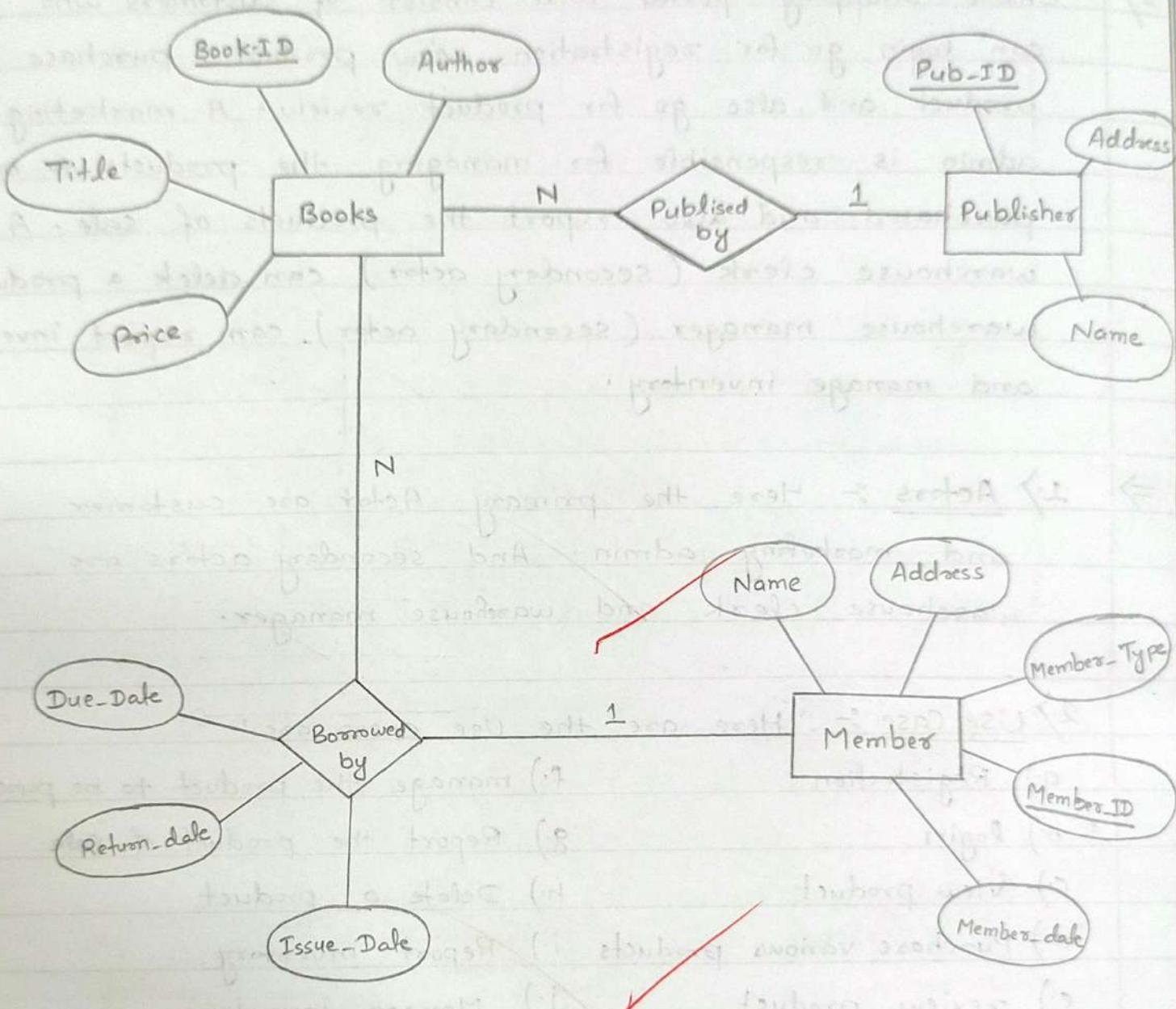
2) Online shopping portal will consist of customers who can login go for registration, view product, purchase product and also go for product review. A marketing admin is responsible for managing the products to be purchased and also report the products of sale. A warehouse clerk (secondary actor) can delete a product. warehouse manager (secondary actor) can report inventory and manage inventory.

⇒ 1) Actors :- Here the primary Actor are customer and marketing admin. And secondary actors are warehouse clerk and warehouse manager.

2) Use Case :- Here are the Use cases are :-

- a.) Registration
- b.) login
- c.) View product
- d.) Purchase various products
- e.) review product
- f.) manage the product to be purchased
- g.) Report the product of sale
- h.) Delete a product
- i.) Report inventory
- j.) Manage inventory

Q 14/03/24



Library Management System

### Lab Assessment - 02

- Q1) Draw the ER diagram for the library management system.

Smart Draw gives us the power to create engineering drawing of all kind more easily and more affordably than any other engineering design software on market.

Smart draw includes a large collection of templates and mechanical engineering and more types of engineering and drawing.

Smart draw is web-based diagram tool used to create flowcharts, organisation charts, mind-maps, projects charts and other business visualization.

• Entities and their Attributes :-

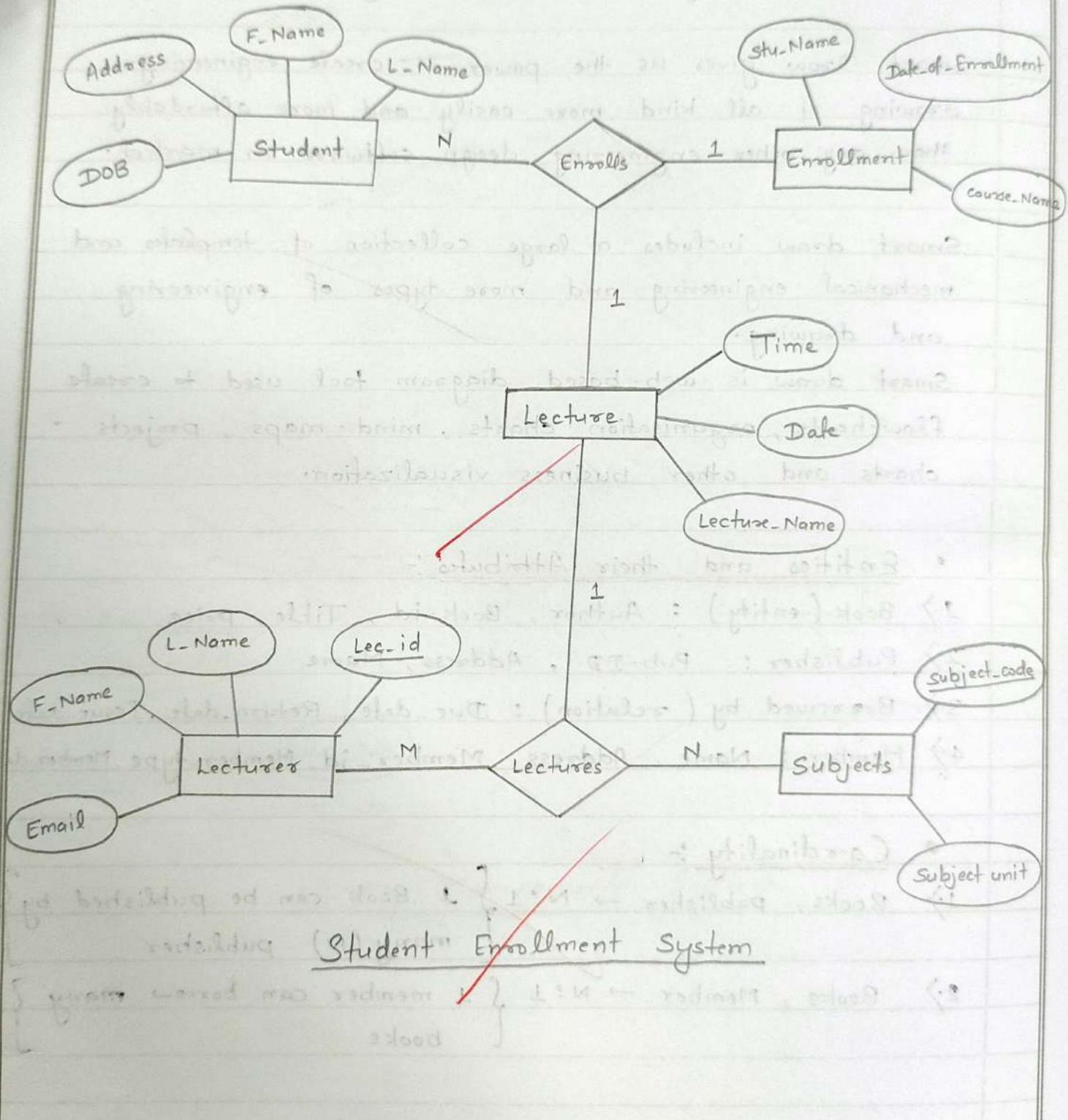
- 1) Book (entity) : Author, Book-id, Title, Price
- 2) Publisher : Pub-ID, Address, Name
- 3) Borrowed by (relation) : Due-date, Return-date, Issue-date
- 4) Member : Name, Address, Member-id, Member-type, Member-date

• Cardinality :-

- 1) Books, publisher  $\rightarrow$  N:1  $\left\{ \begin{array}{l} 1 \text{ Book can be published by} \\ \text{many (N) publisher} \end{array} \right\}$
- 2) Books, Member  $\rightarrow$  N:1  $\left\{ \begin{array}{l} 1 \text{ member can borrow many} \\ \text{books} \end{array} \right\}$

22-09-2022

### SO - from 2021 doc



Expt. No. \_\_\_\_\_

02) Draw the ER diagram for student enrollment system.

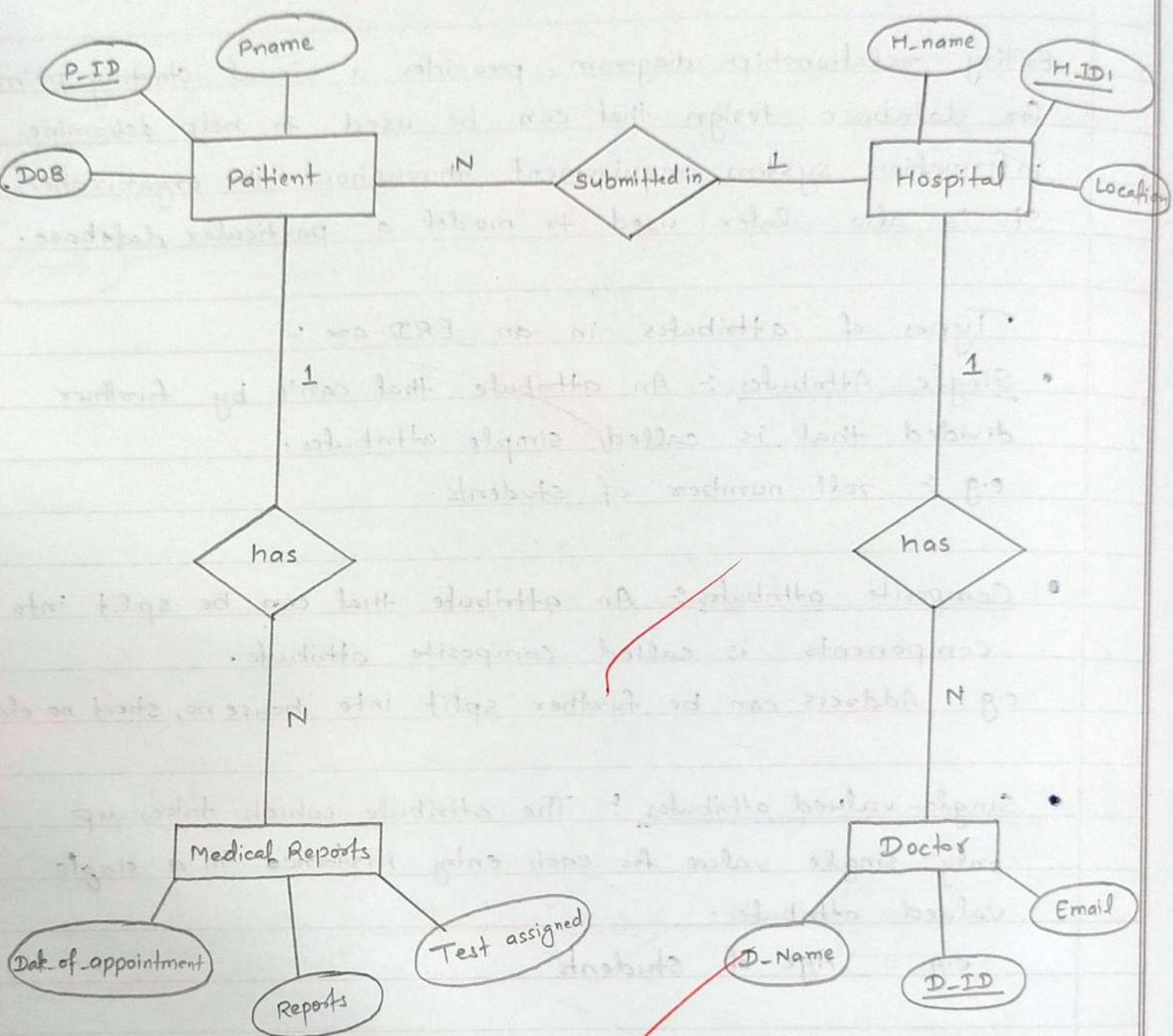
Entity relationship diagram provides a visual starting point for database design that can be used to help determine information system requirement throughout the organization. It is also later used to model a particular database.

Types of attributes in an ERD are :-

- Single Attributes :- An attribute that can't be further divided that is called simple attributes.  
e.g :- roll number of students.
- Composite attributes :- An attribute that can be split into components is called composite attribute.  
e.g :- Address can be further split into house no, street no etc.
- Single-valued attributes :- The attribute which takes up only single value for each entry instance in a single valued attribute.  
e.g :- Age of students.

Cardinality : (Students enrollment System)

- (1.) Student, enrollment :- N : 1
- (2.) Student, lecture → N : 1
- (3.) Lecturer, lecture → N : 1
- (4.) Lecturer, subject → N : M



hospital Management System

3.) Draw the ER diagram for hospital management system.

- Multivalued Attribute :- The attribute which takes up more than a single value for each entry instance is a multi-valued attribute.

e.g :- Phone no, location etc.

- Derived Attributes :- An attributes that can be derived from other attributes is known as derived attribute.

e.g :- Age can be derived from Date of Birth.

- Complex Attribute :- If any attribute has the combining property of multivalued and composite attributes then it is called complex Attribute.

- Entity and its attributes :-

- 1.) Patient  $\rightarrow$  P-name, P-id, DOB
- 2.) Hospital  $\rightarrow$  H-name, H-id, location
- 3.) Medical records  $\rightarrow$  Reports, Date-of-appointment, Tests given
- 4.) Doctor  $\rightarrow$  D-name, Doc-ID, email.

- Cardinality :-

- 1.) Patient, Hospital  $\rightarrow$  N:N
- 2.) Hospital, Doctor  $\rightarrow$  1:N
- 3.) Patient, Medical Records  $\rightarrow$  1:N

## LAB ASSIGNMENT - 3

### ACTIVITY DIAGRAM :-

Activity diagrams are used to illustrate the flow of control in a system and refers to the step involved in the execution of a use case. We can depict both sequential processing and concurrent processing of activities using an activity diagram i.e. an activity diagram focuses on the condition of flow and sequence in which it happens.

- Activity Diagram Notations :-

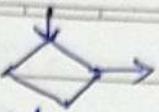
(i) Initial State :- We use a black filled circle to depict the initial state of a system. For objects, this is the state when they are instantiated. It marks the entry point and the initial activity state.  $\Rightarrow \bullet$

(ii) Action state :- Action state

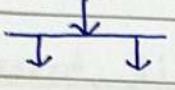
It is represented using a rectangle with rounded corners. Basically any action or event that takes place is represented using an activity.

(iii) Action flow or control flow :-  $\Rightarrow \rightarrow$

This is also referred to as paths and edges. They are used to show the transition from one activity state to another state.

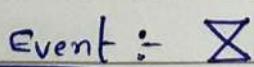
(iv) Decision node and Branching :-  $\Rightarrow$    
 When we need to make a decision before deciding the flow of control, we use the decision mode. The outgoing arrows from the decision node can be labelled with conditions.

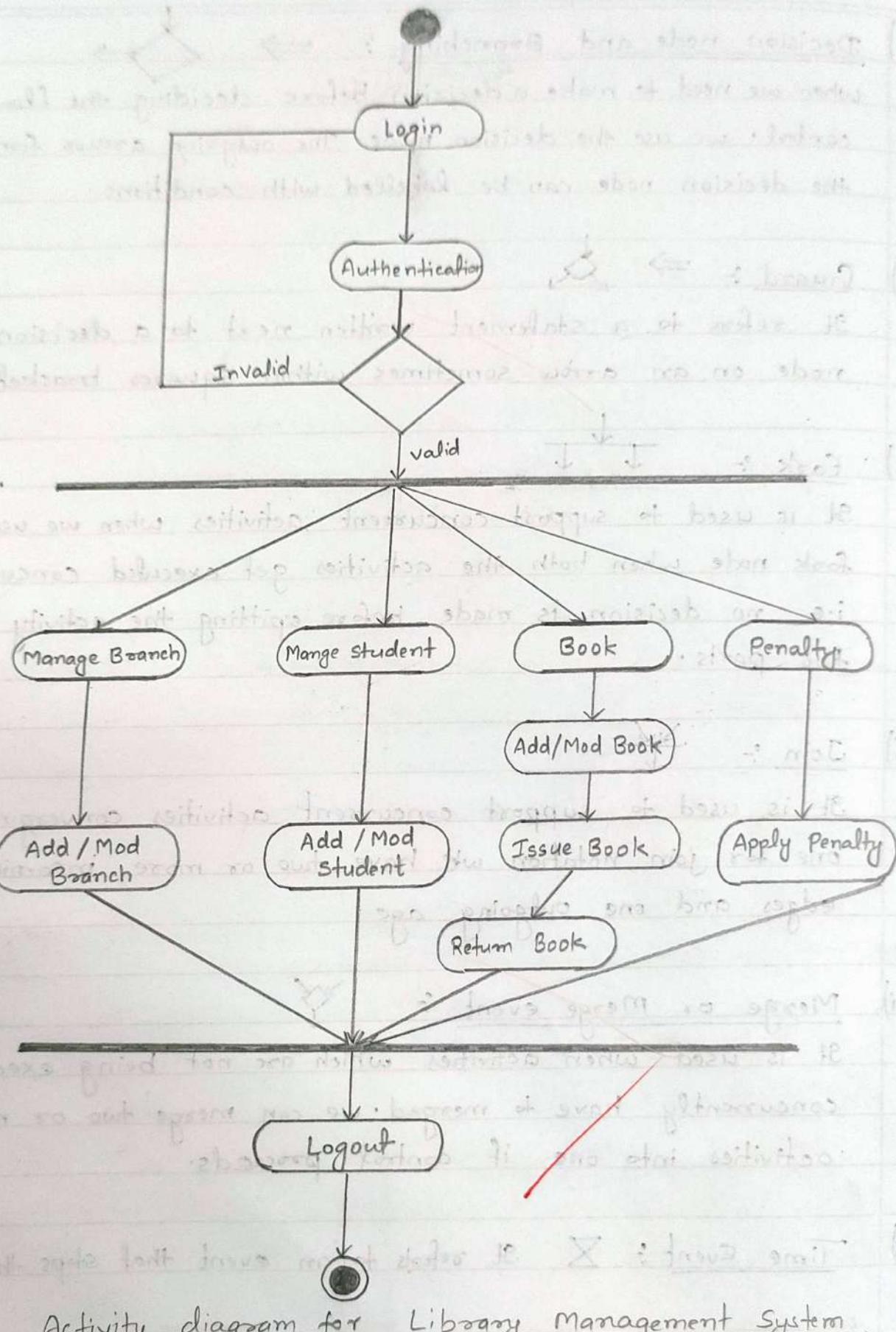
(v) Guard :-   
 It refers to a statement written next to a decision node on an arrow sometimes within square brackets.

(vi) Fork :-   
 It is used to support concurrent activities when we use a fork node when both the activities get executed concurrently i.e. no decision is made before splitting the activity into two parts.

(vii) Join :-   
 It is used to support concurrent activities converging into one for join notation we have two or more incoming edges and one outgoing edge.

(viii) Merge or Merge event :-   
 It is used when activities which are not being executed concurrently have to be merged. we can merge two or more activities into one if control proceeds.

(ix) Time Event :-   
 It refers to an event that stops the flow.



Activity diagram for Library Management System

for a time. It is an hourglass depicts it. It's used when an event takes some time to be completed.

(x) Final or End state :- It's used when the system reaches to its end. we use a filled circle notation to represent the final state.

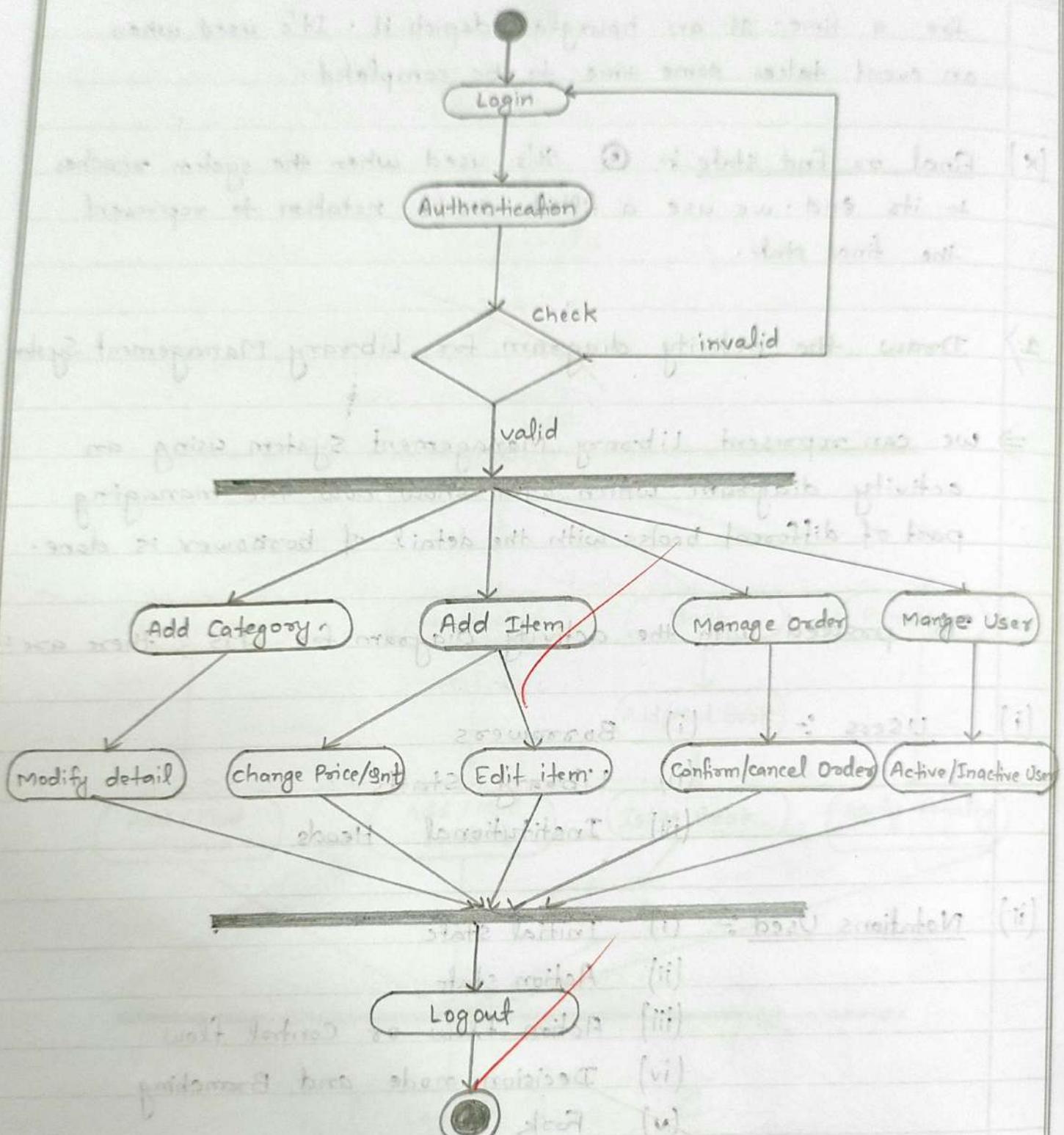
1) ~~Draw the activity diagram for Library Management System?~~

⇒ we can represent Library Management System using an activity diagram which will show how the managing part of different books with the detail of borrower is done.

To proceed with the activity diagram for LMS, there are :-

- (i) Users :-
- (i) Borrowers
  - (ii) Library Staffs
  - (iii) Institutional Heads

- ii) Notations Used :-
- (i) Initial state
  - (ii) Action state
  - (iii) Action flow or Control flow
  - (iv) Decision node and Branching
  - (v) Fork two
  - (vi) Join
  - (vii) Final state or End state



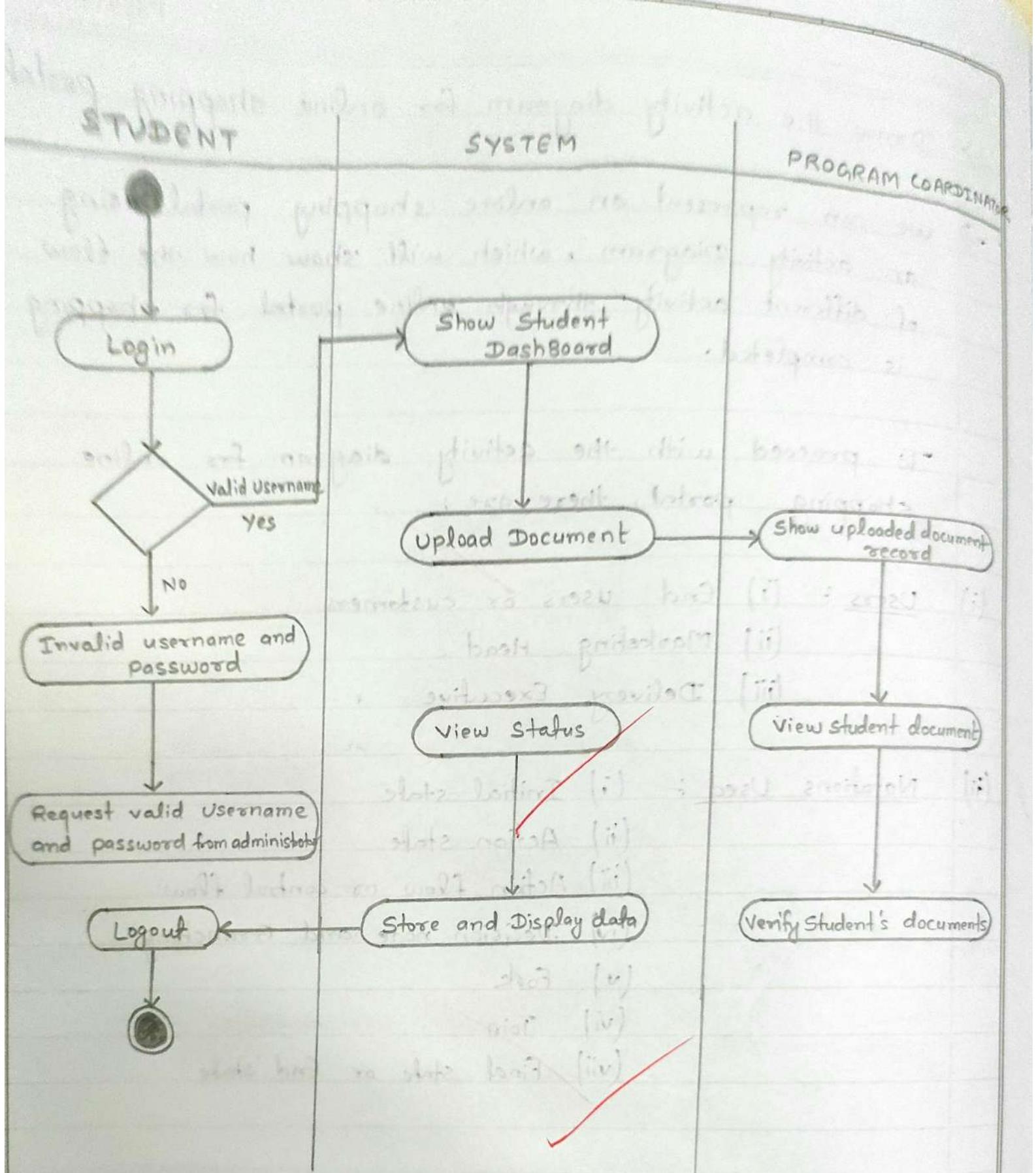
Activity diagram for Online shopping portal

2) Draw the activity diagram for online shopping portal:

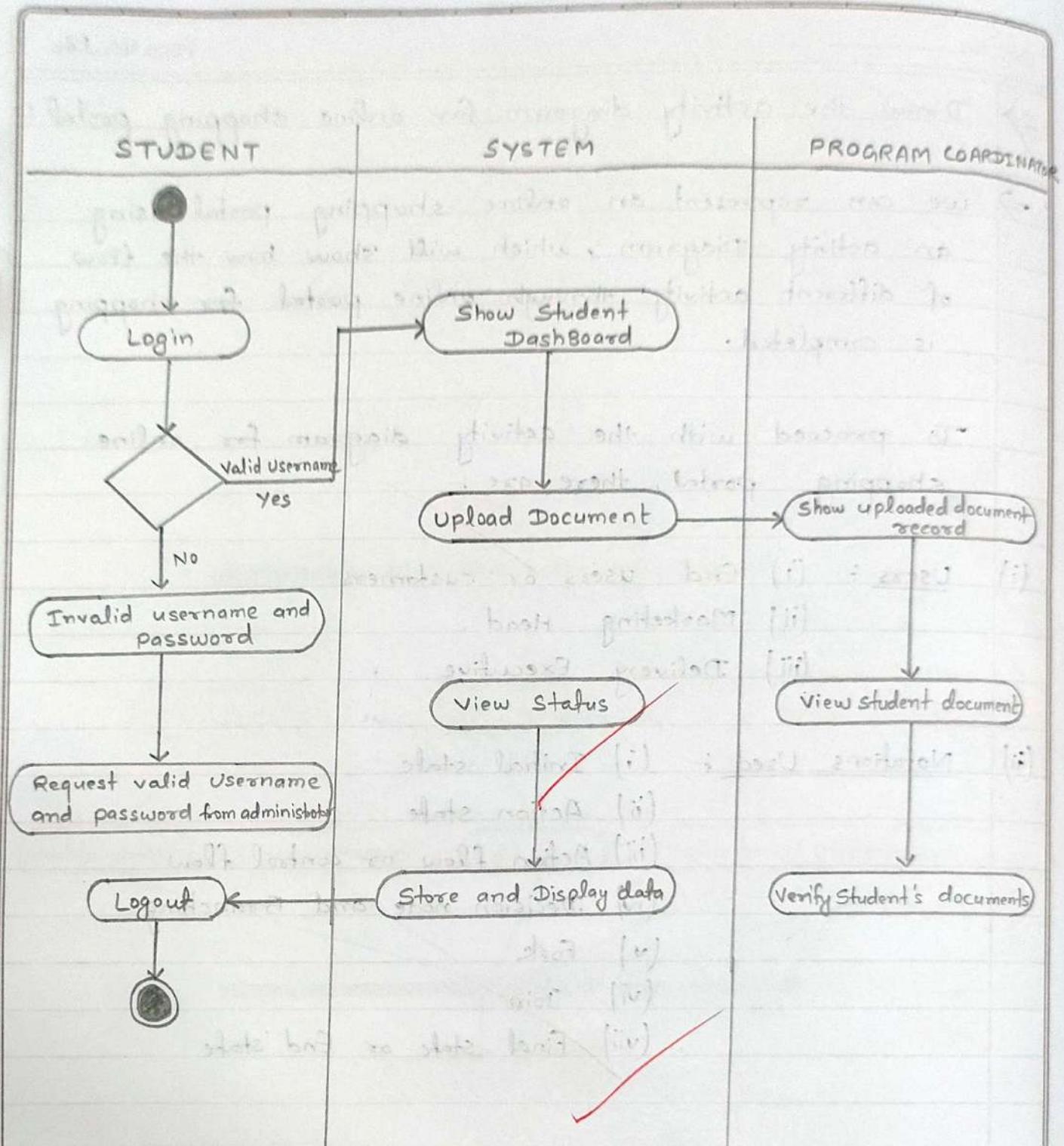
→ we can represent an online shopping portal using an activity Diagram, which will show how the flow of different activity through online portal for shopping is completed.

To proceed with the activity diagram for online shopping portal, there are :-

- (i) Users :-
  - (i) End users or customers
  - (ii) Marketing Head
  - (iii) Delivery Executive
  
- (ii) Notations Used :-
  - (i) Initial state
  - (ii) Action state
  - (iii) Action flow or control flow
  - (iv) Decision node and Branching
  - (v) Fork
  - (vi) Join
  - (vii) Final state or End state



Activity diagram for Student enrollment system



Activity diagram for Student enrollment system

3) Draw the activity diagram for student enrollment system?

⇒ student enrollment system is an important process in any institutions. To proceed with the activity diagram for student enrollment system there are :-

- (i) Users :- (i) Students  
(ii) System  
(iii) Program coordination
- ii) Notations Used :- (i) Initial state  
(ii) Action state  
(iii) Action flow or control flow  
(iv) Decision node and Branching  
(v) Final state or End state.

Q 18/04/21

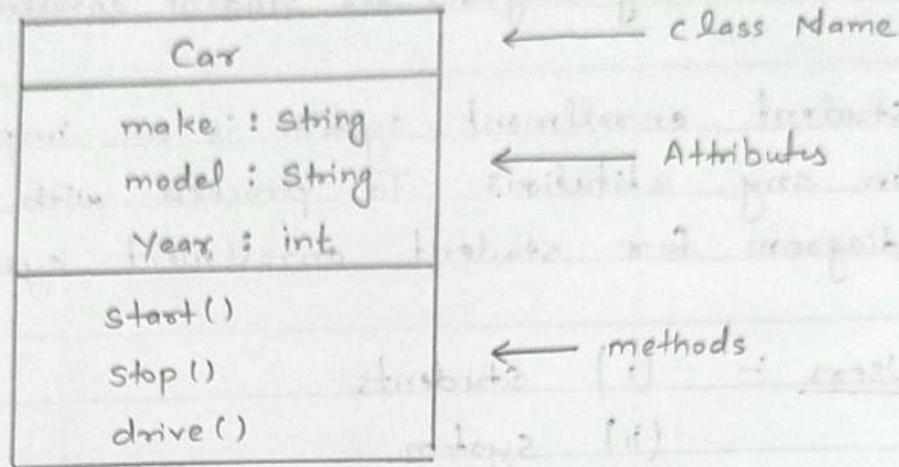


Fig1:-

UML Class Notation

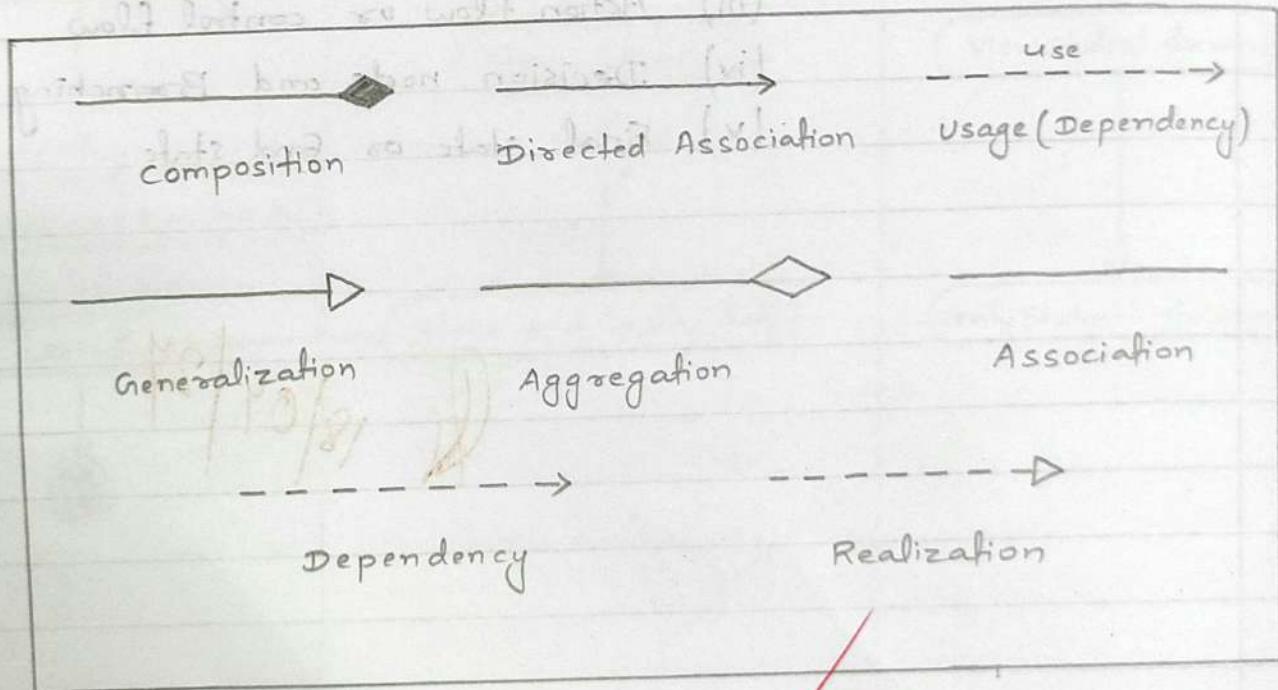


Fig2 :- class Diagram Relationships

## LAB ASSIGNMENT - 04

### CLASS DIAGRAM UML :-

Class diagrams are a type of UML (Unified Modelling Language) diagram used in software engineering to visually represent the structure and relationship of classes in a system. Class diagrams provide a high-level overview of a system's design, helping to communicate and document the structure of the software.

### UML Class Notation :-

Class notation is a graphical representation used to depict classes and their representation and relationship in object-oriented programming modelling. Following are present and represent in UML class Notation :- (see in Fig 1)

(i) Class Name    (ii) Attribute    (iii) Methods.

### Relationship between classes :-

In class diagrams, relationships b/w classes describe how classes are connected or interact with each other within a system. There are several types of relationships in object-oriented modelling, each having specific purpose. Here are some common types of relationship in class diagrams (see in fig 2) :-

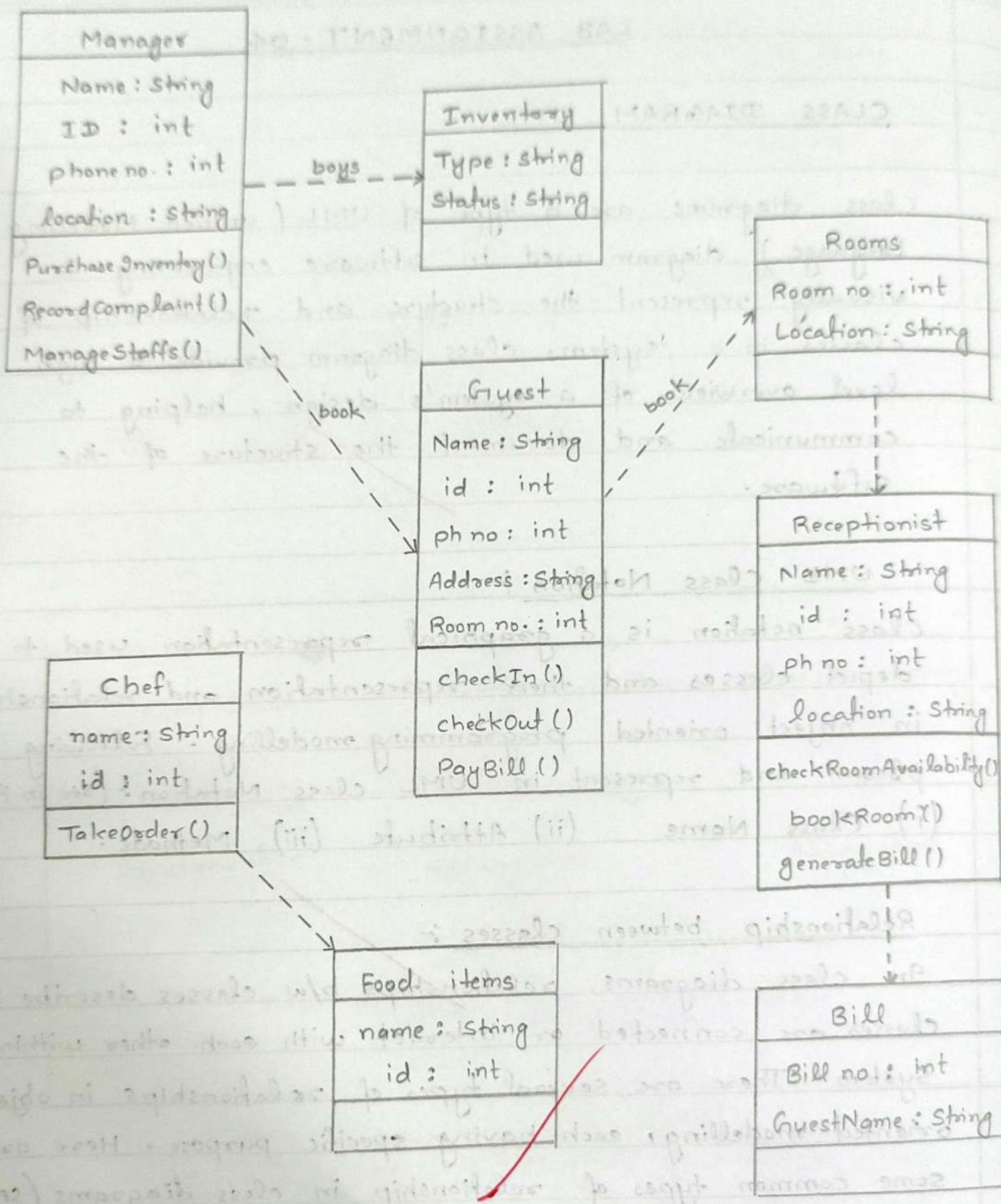


Fig:- Class Diagram for hotel management system

1) Draw a Class Diagram UML for Hotel management system

Creating a comprehensive class diagram for a hotel management system involves identifying the main class components and their relationships within the system.

The class which involves in the comprehensive class diagram of hotel management system are :-

- (i) Manager
- (ii) Inventory
- (iii) Guest
- (iv) Rooms
- (v) Receptionist
- (vi) Bill
- (vii) Food items
- (viii) ~~Bill~~ chef.

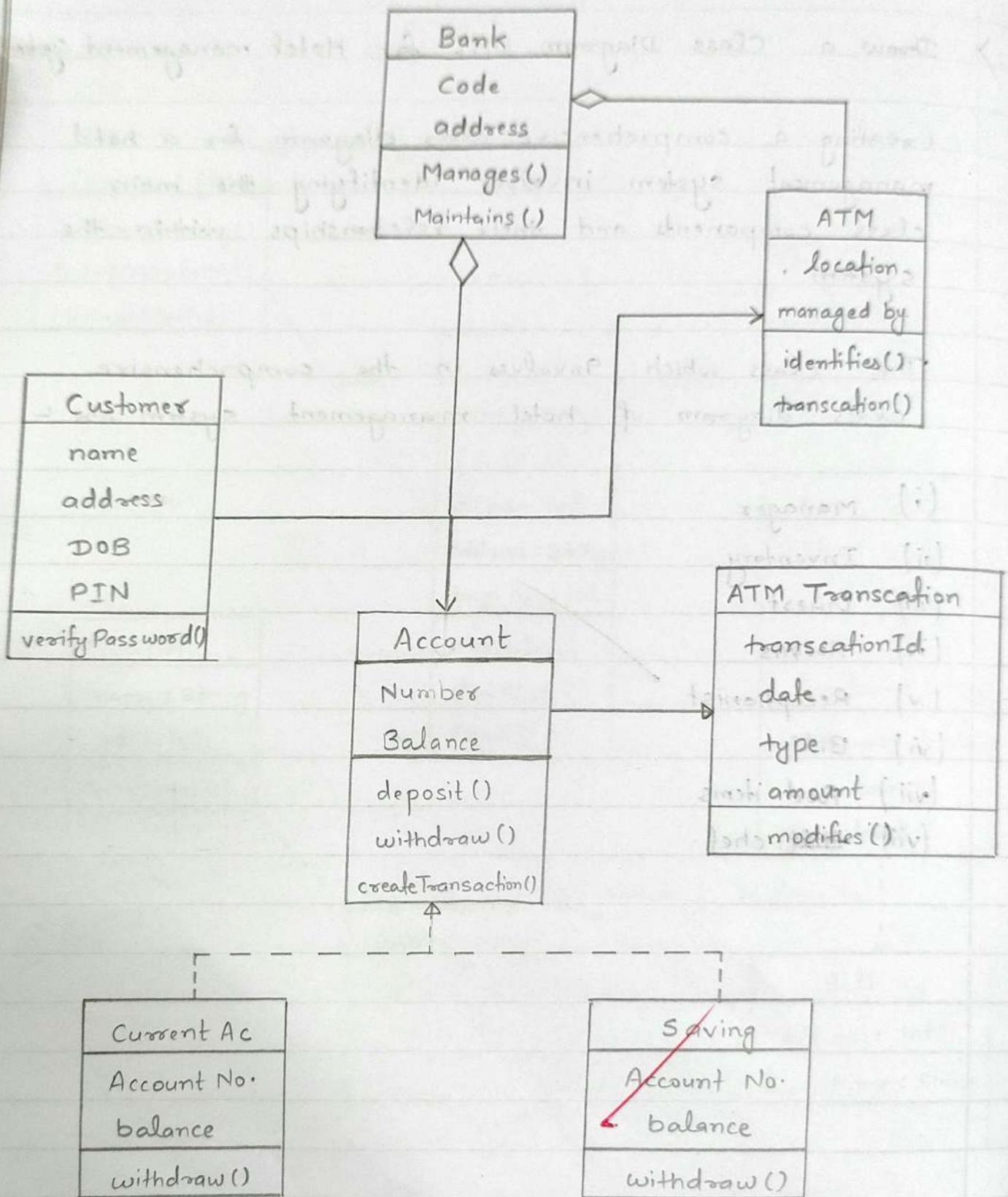


fig2 :- Class Diagram of bank management system.

2.) Draw a class diagram UML for Bank management system ?

Designing an comprehensive class diagram for bank management system involves identifying the main entities class involved in making banking management system and establishes the relationship between these entities classes within the system. The simplified figure (2) of a class diagram for a bank management system drawn aside page.

The class which involves in the comprehensive class diagram of bank management system are -

- (i) Bank
- (ii) Customer
- (iii) Account
- (iv) ATM
- (v) ATM Transaction
- (vi) Current Account
- (vii) Saving

Q 09/05/24

## LAB ASSIGNMENT-05

### DFD ( Data Flow Diagram ) :-

DFD is the abbreviation for Data Flow Diagram.

The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present.

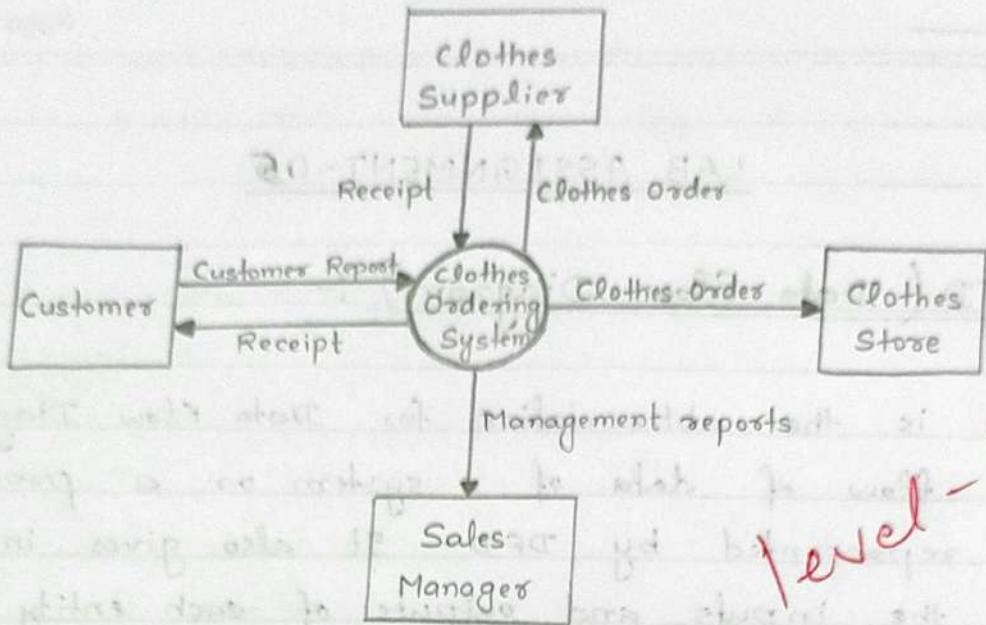
Specific operations depending on the type of data can be explained by a flowchart.

It is a graphical tool, useful for communicating with users, managers and other personnel. It is useful for analyzing existing as well as proposed system.

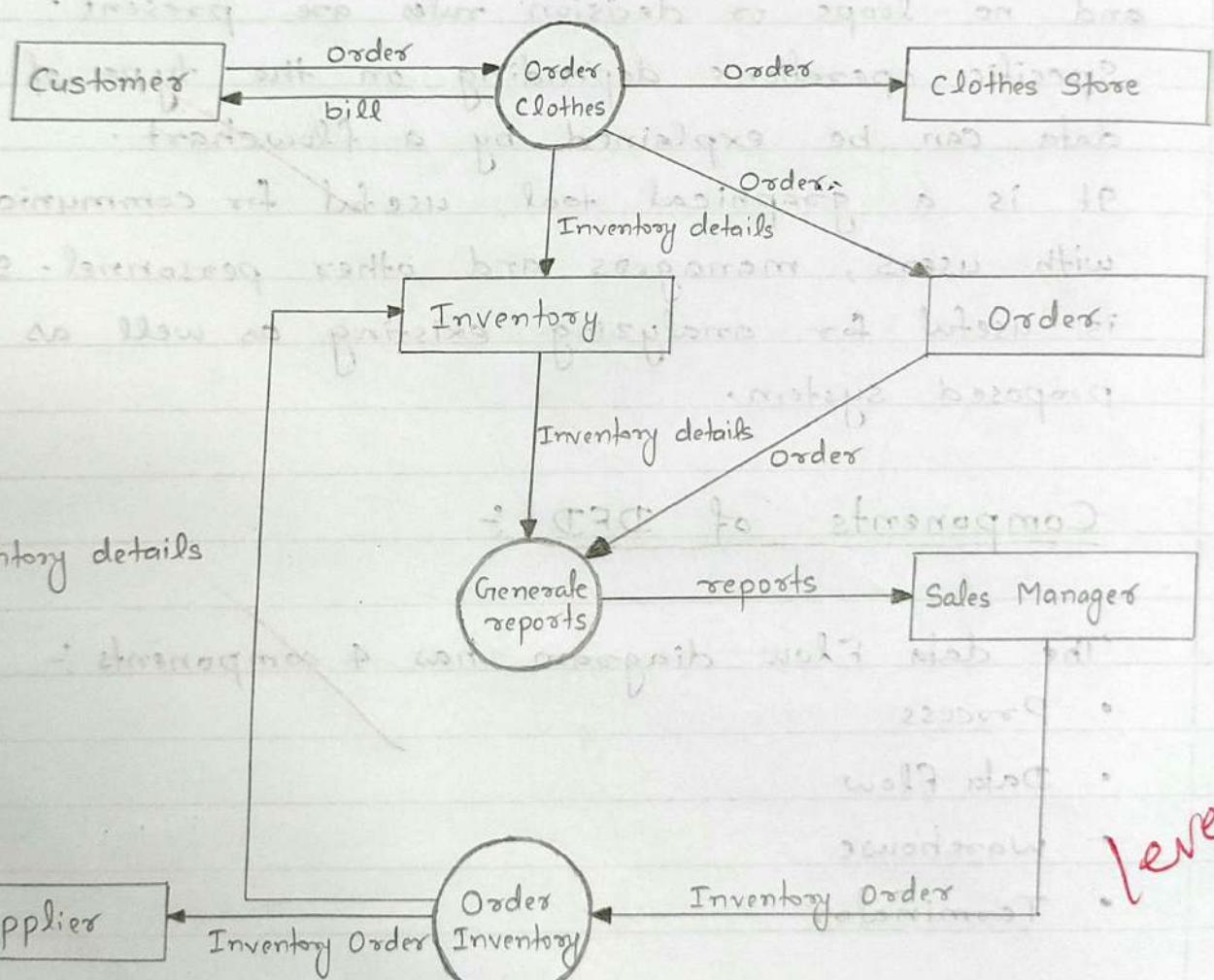
### Components of DFD :-

The data Flow diagram has 4 components :-

- Process
- Data Flow
- Warehouse
- Terminator



level - 0



level - 1

Level 1, Level 2 DFD of Clothes ordering System

1.) Draw the Level 1 and Level 2 DFD of Clothes Ordering System.

⇒ Levels of DFD :- DFD uses hierarchy to maintain transparency thus multilevel DFD's can be created.

- 0-level DFD :- It represents the entire system as a single bubble and provides an overall picture of the system.
- 1-level DFD :- It represents the main functions of the system and how they interact with each other.
- 2-level DFD :- It represents the process within each function of the system and how they interact with each other.
- 3-level DFD :- It represents the dataflow within each process and how the data is transformed and stored.

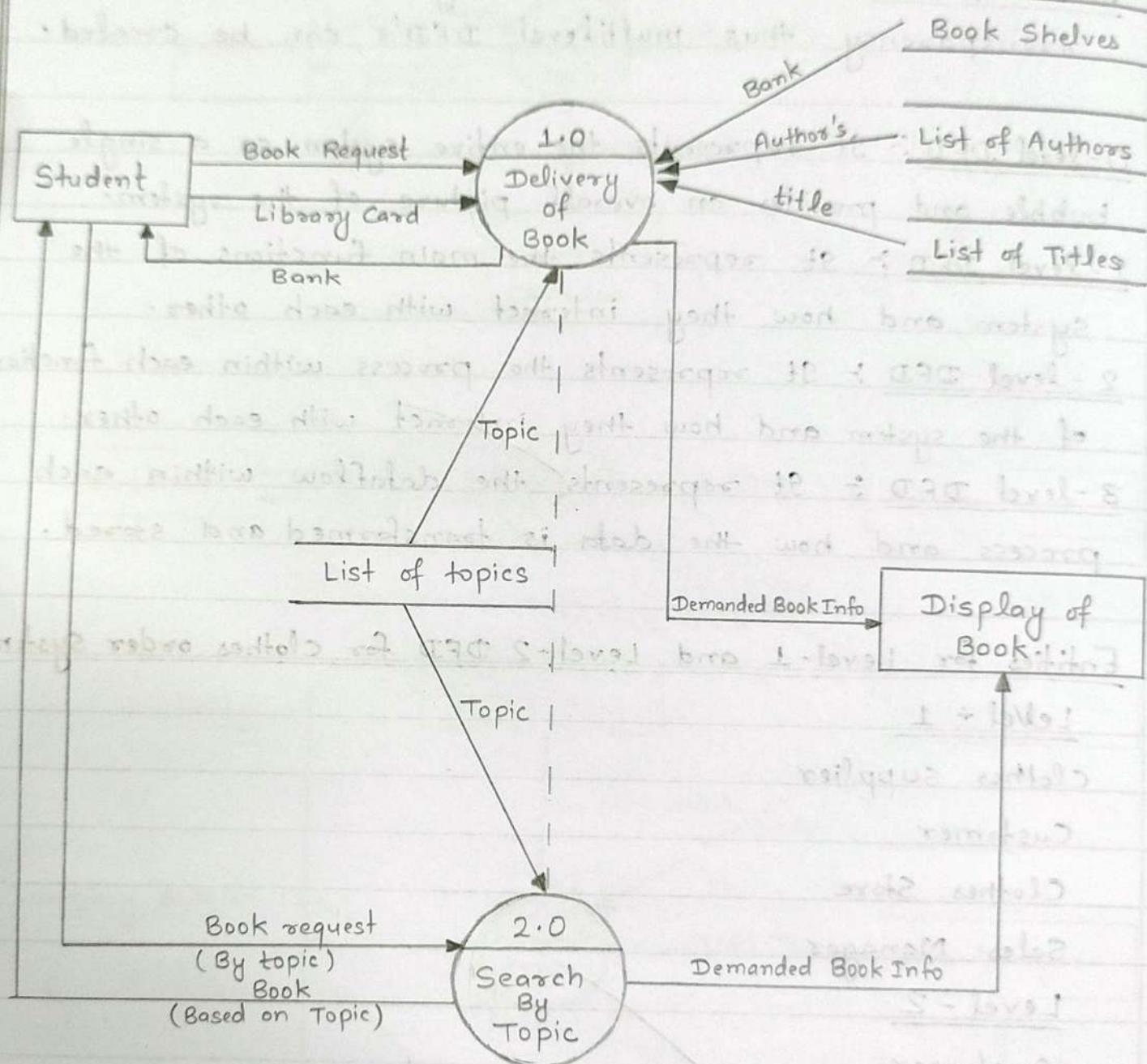
Entities for Level-1 and Level-2 DFD for clothes order system:

Level - 1

- Clothes supplier
- Customer
- Clothes Store
- Sales Manager

Level - 2

- Customer
- Supplier
- Inventory
- Clothes Store
- Order
- Sales Manager



Level 1 DFD of Library Management System

2) Draw the DFD for Library Management System :-

→ Symbols used in DFD :-

Square Box :- A square box defines source or destination of the system. It is also called entity. It is represented by rectangle.

Arrow or line :- An arrow identifies the data flow i.e it gives informations of the data that is in motion.

Circle or bubble chart :- It represents as a process that gives us information. It is also called processing box.

Open Rectangle :- An open rectangle is a data store. In this data is store either temporary or permanently.

Entities for Level 1 DFD of Library Management System :-

Delivery of Book

Search by Topic

Student

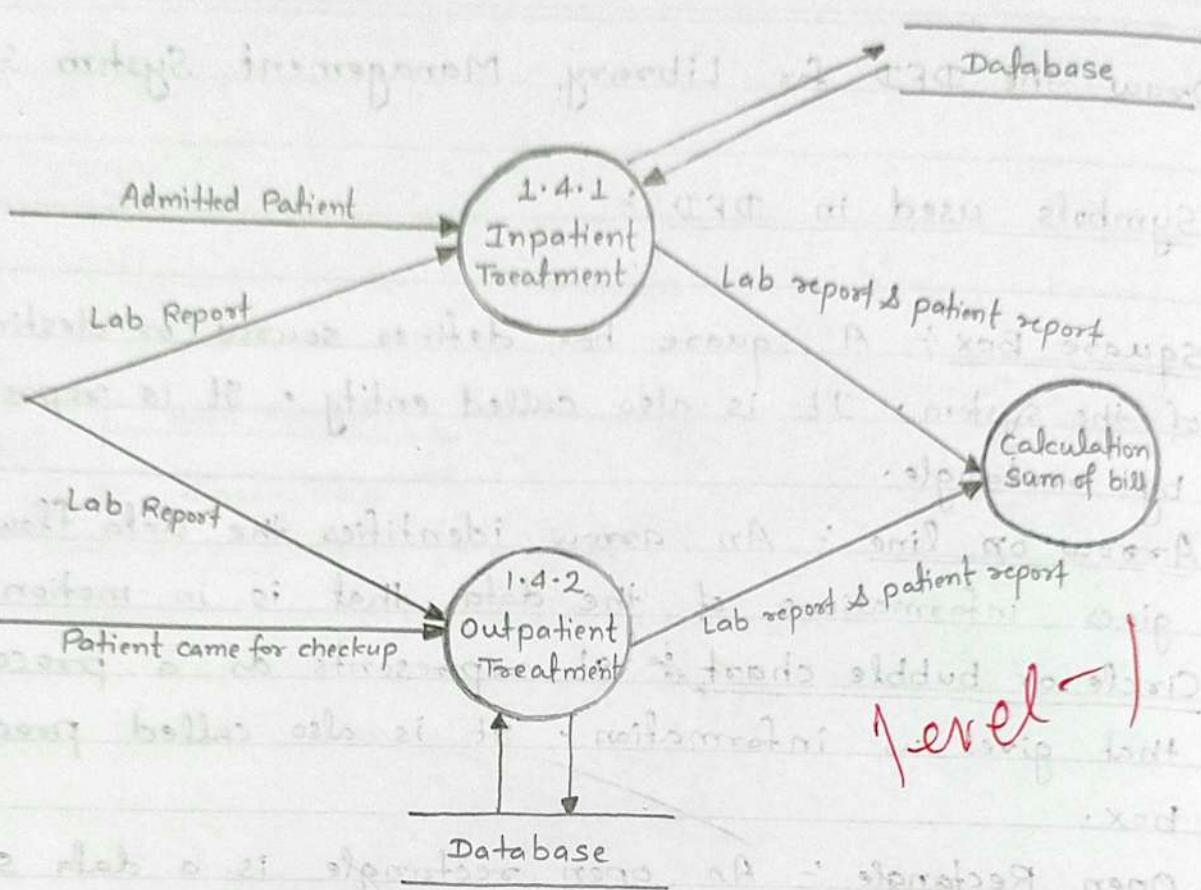
Display of Book

Book shelves

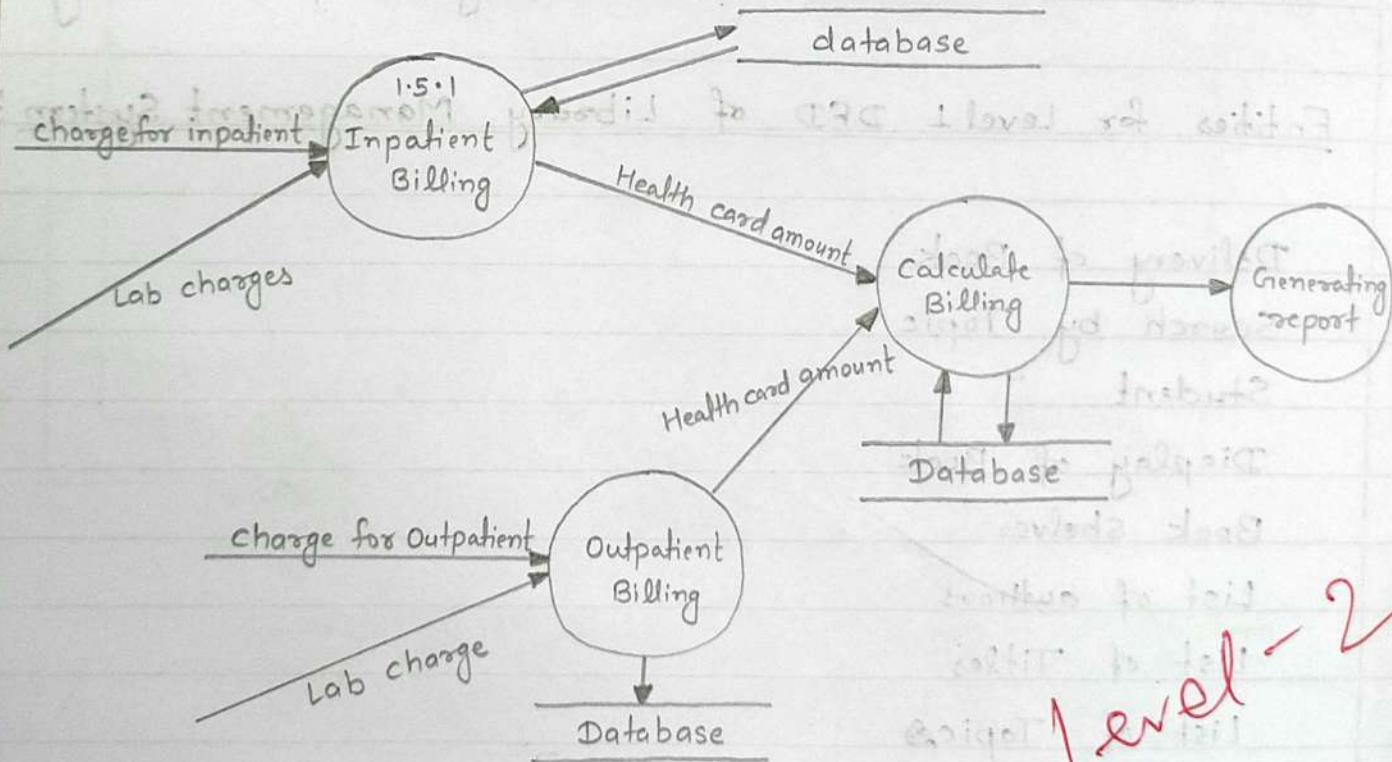
List of authors

List of Titles

List of Topics



Level - 1



Level - 2

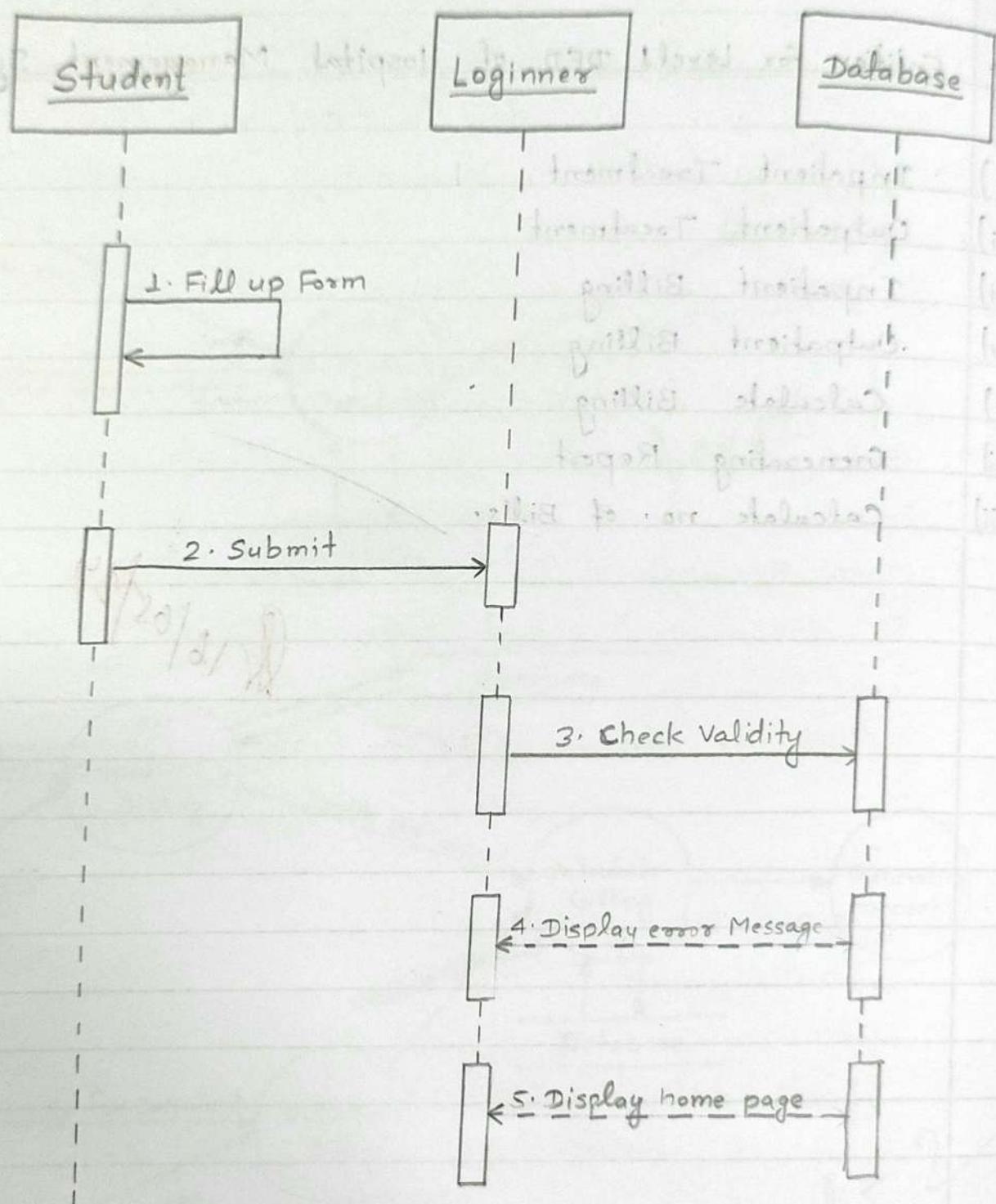
Level 1 DFD of Hospital Management System

3) Draw the DFD of Hospital Management System:

→ Entities for Level 1 DFD of Hospital Management System:

- (i) Inpatient Treatment
- (ii) Outpatient Treatment
- (iii) Inpatient Billing
- (iv) Outpatient Billing
- (v) Calculate Billing
- (vi) Generating Report
- (vii) Calculate no. of Bills.

16/05/29



## Sequence diagram for student Registration system

1.) Draw sequential Diagram for student Registration system!

Sequence Diagram are interaction diagrams that detail how operations are carried out. They capture the interaction b/w objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when. It consists of two parts :-

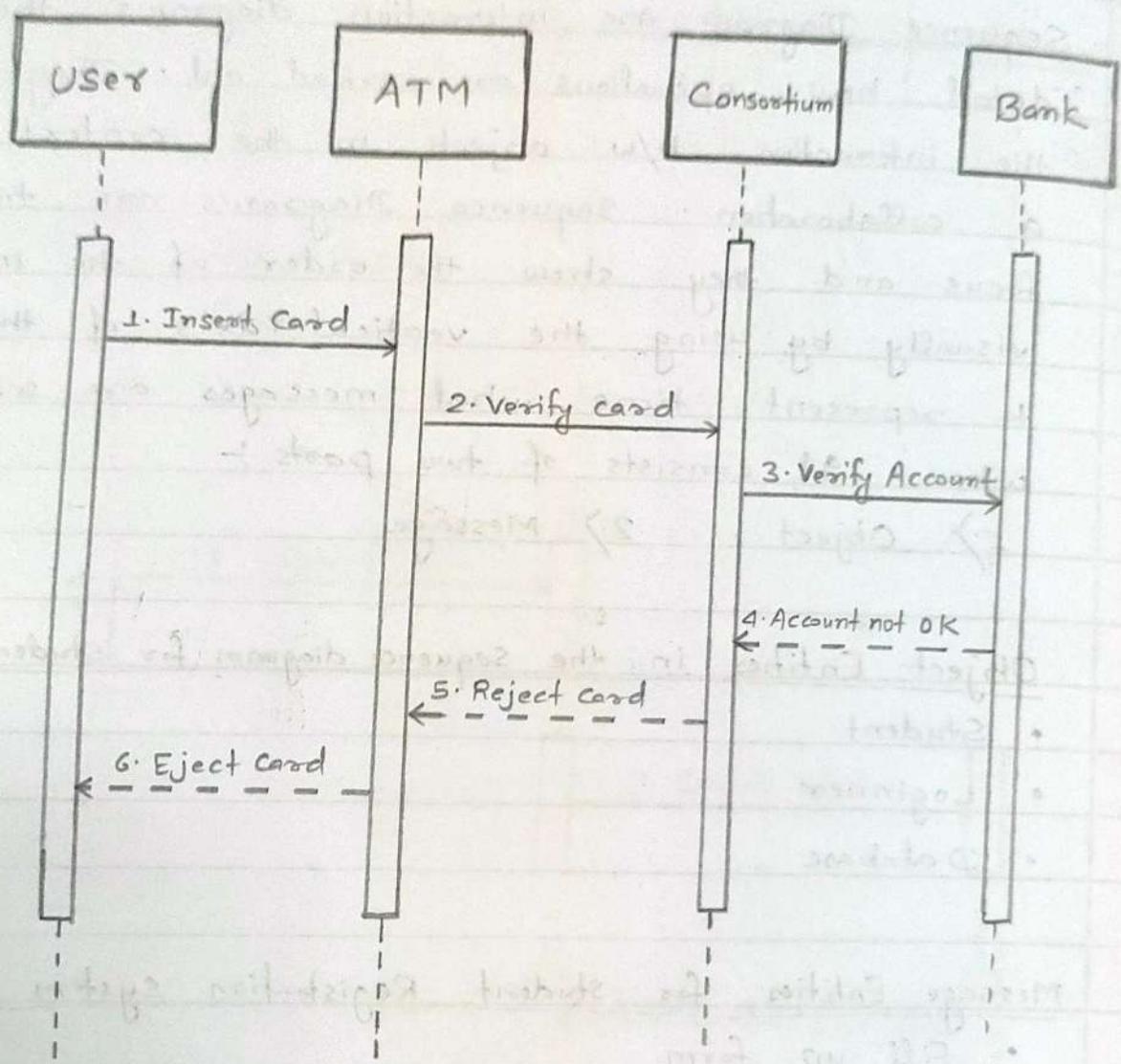
- 1.) Object
- 2.) Message

Object Entities in the Sequence diagram for student Registration System :-

- Student
- Loginner
- Database

Message Entities for student Registration system :-

- Fill up form
- Submit
- check validity
- Display error Message
- Display home page



Sequence Diagram for ATM Management System

2) Draw Sequence Diagram for ATM Management System ?

⇒ Object Entities for sequence diagram of ATM Management System :-

- User
- ATM
- Consortium
- Bank

⇒ Message Entities for sequence diagram of ATM management system :-

- Insert card
- Verify card
- Verify account
- Account not ok
- Reject card
- Eject card.