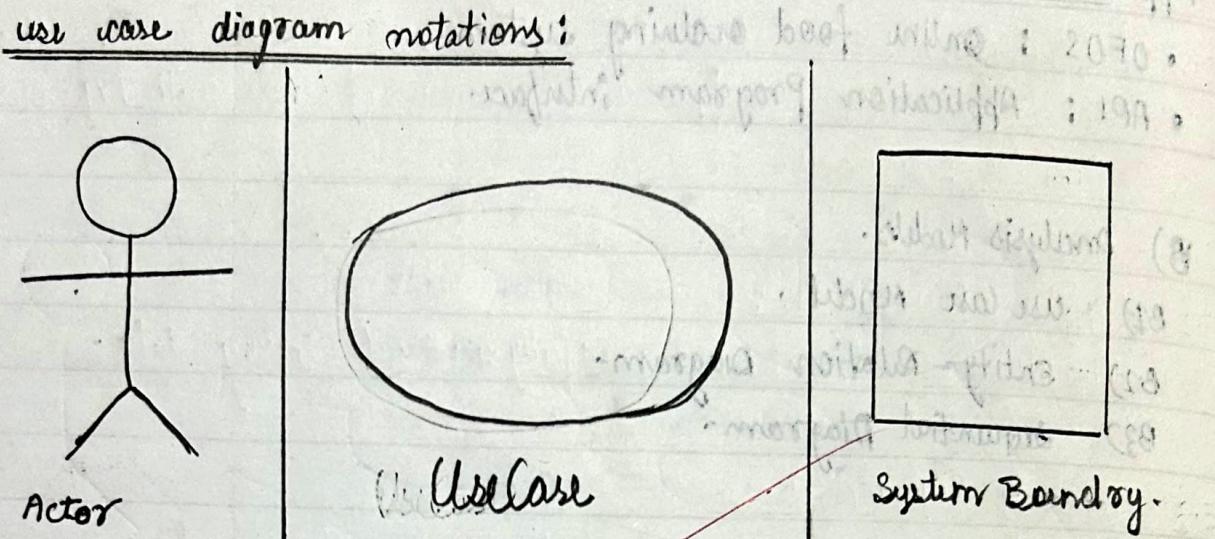


Index

Sl. No.	Name of the Experiment	Page No.	Date of Experiment	Date of Submission	Remarks
1.	Design Requirement analysis for Payroll management system.	1	18/07/2025	25/7/2025	1
2.	Software Requirement Specification for online food ordering system (OFOS)	4	25/7/2025	01/08/2025	Y
3.	Drawing the user's view analysis - use use case diagram for online food ordering system	10	26/08/2025		
4.	Draw & study the activity diagram of order management system	12	11/9/2025		Rec, 21/9/2025
5.	Create & analyse the class diagram on library management system	14			1
6.	To Draw & Study the state chart diagram for lib. management system.	16			Y
7.	Draw the Object Diagram for library management system.	18.			
8.	Draw and analyse the sequential diagram of library management system	20.			Recvd 10/10/2025
9.	Unit testing on calculator program using Junit testing framework	22			Y
10.	Perform integration testing on calculator program using Junit.	25			Recvd 31/10/2025

Inde x



User Case Diagram | Unified Modeling Language (UML)

~~• Simplified notation from UML 2.0
• off the standard notation rules~~

- Drawing the user's view analysis - use case diagram for online food ordering system.

What is a Use Case Diagram?

Use case diagram refers to it as a Behaviour model or diagram.

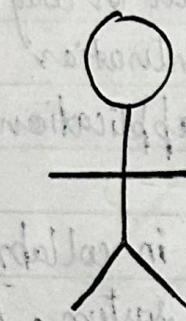
- It simply describes & displays the relationship or interaction b/w the user & or customer & providers of the application services - on the system.
- It describes diff. actions that a system performs in collaboration to achieve something with one or more users of the system. A use-case diagram is used a lot nowadays to manage system.

~~# Use Case Diagram Notations~~

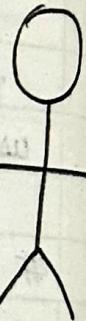
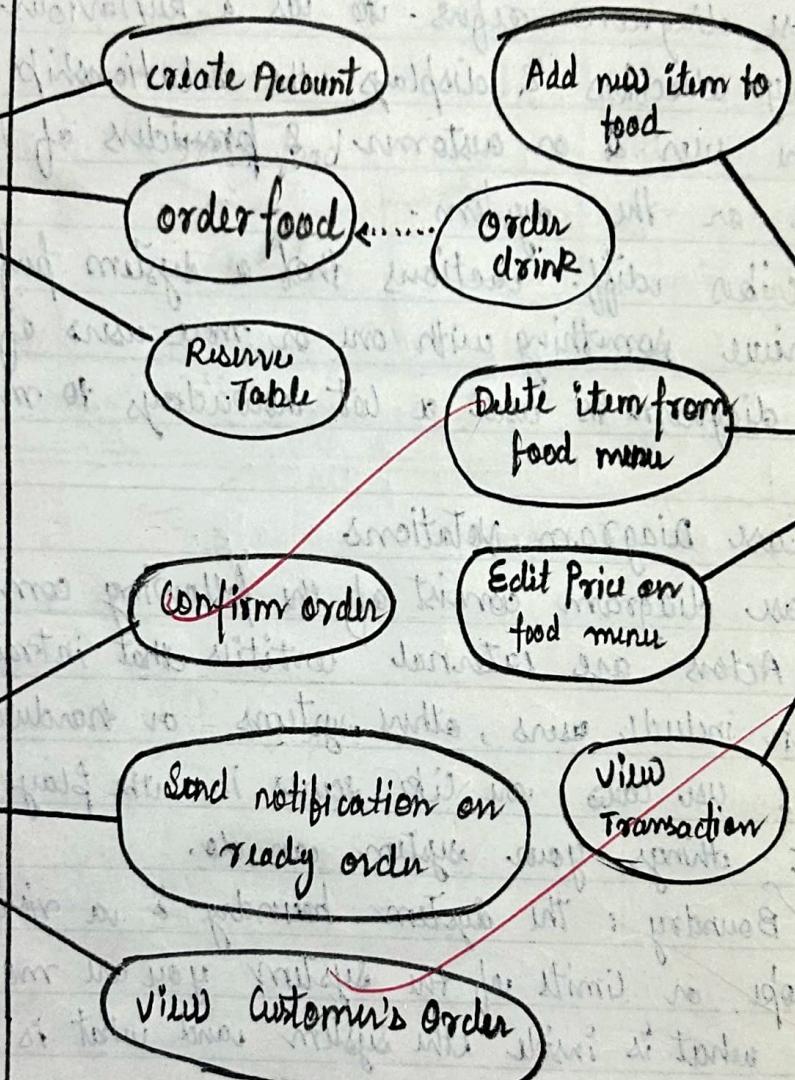
Use case diagram consist of the following components:

1. Actor : Actors are external entities that interact with a system. These can include users, other systems or hardware devices.
2. Use case : Use cases are like scenes in the play. They represent specific things your system can do.
- 3) System Boundary : The system boundary is a visual representation of the scope or limits of the system you are modelling. It defines what is inside the system and what is outside.

User tasks of Online food Ordering System



Customer



Administrator



Chef

- use case diagram of online food ordering system.
The diagram represents the interactions of customers, chefs & admin in online food ordering system.

1. Actors:-

- customers : end user who interact with the system to order food.
- chef : Responsible for handling order & related operations.
- Administrator : manages restaurant backend system incl. Main menu item, price & transaction details.

2. Use Cases :

- Create account.
- Order food.
- Order drink.
- ~~Confirm order.~~
- ~~View customer's order.~~
- Send notification on Ready order.
- Add new item to food menu.
- ~~Remove item from food menu.~~
- Edit Price on food menu.
- View transactions.

3. System Boundary :

The system boundary encompasses all the use cases listed above which collectively defines restaurant's online food ordering system. The actors only interacts only through system interface to perform their respective operations / functions.

Deeksha Verma

Teacher's Signature : _____

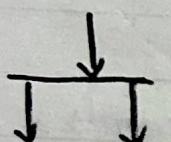
Activity diagram notations:

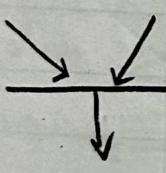
● : Start Node.

○ : End Node.

 : Action Node.

 : Decision Node.

 : Fork.

 : Merge / Join.

→ : Control flow.

■ To know & study the activity diagram of an system order management system.

■ Activity Diagram: It is used to illustrate the flow of control in a system and refers to the steps involved in the execution of a task.

It is a type of behavioural diagram and we can depict both sequential processing & concurrent procession of activities using an activity diagram i.e. an activity ~~flow~~ focuses on the condition of flow and the sequence in which it happens.

■ Activity diagram Notations:

- 1) Initial State:- Starting state before taking an activity takes place is represented using an activity initial stat.
- 2) Action or Activity State:- An action represented execution of an action or event that takes place is represented using an activity.
- 3) Action flow or flow control:- It refers to waypoints and edges. They are used to show the transition from one activity state to another.
- 4) Decision Node:- When we need to make a decision before deciding the flow of control, we use the decision node.
- 5) Guard:- It refers to a statement written next to a decision made on an arrow sometimes with in square brackets.
- 6) Fork:- fork nodes are used to support concurrent activities.
- 7) Join:- Join nodes are used to support concurrent activities converging into ones.
- 8) Merge or merge events:- Scenarios arise when activities which are not being executed concurrently have to be merged.

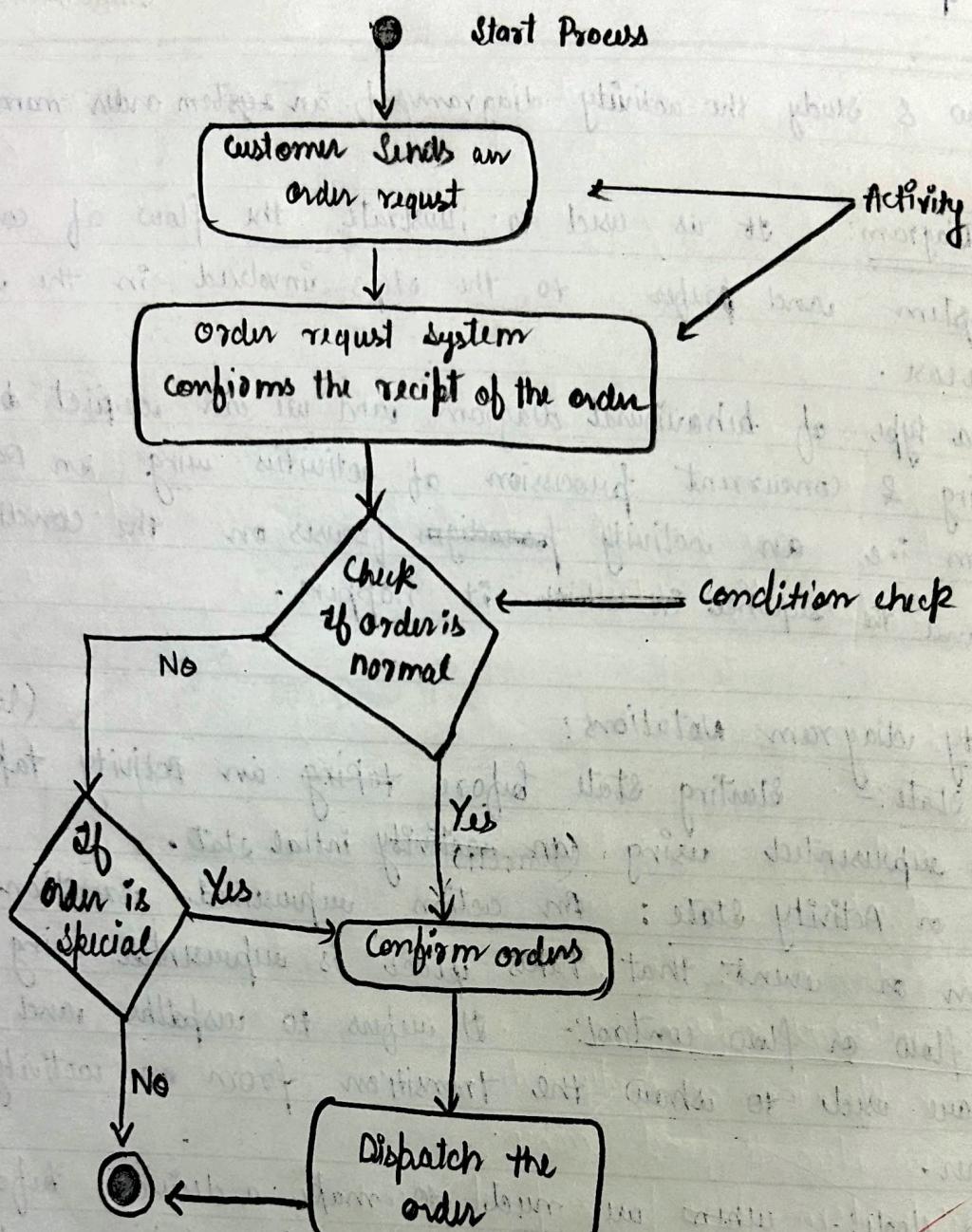


Fig : Activity diagram of an order management system

- 9) Guardians: used for grouping related activities in one columns.
- 10) Time events: an event that stops the flow for a time.
- 11) Final state: The state which the system reaches when a particular process or activity ends is known as final state or end state.

Activity diagram:

- 1) Start process:- Process begins with the customers sending an order request.
- 2) Order request system confirms receipt : The system acknowledges that it has received the order request from the customers.
- 3) Check if the orders is normal or not normal :
- A decision node checks the type of orders.
 - If the orders is normal process continues to confirm.
- 4) Check if the order is special :
- Another decision making mode checks whether the order is special.
 - If yes, then it also goes to the confirm the orders.
 - If no, the process ends (Termination Node).
- 5) ~~Confirm the order~~ :-
- Once the order type is validated , the system confirms the orders.
- 6) Dispatch the order:-
- After confirmation the order is dispatched to the customers.
- 7) Termination of the process:
- The activity ends after dispatching the order.

Tech 2017

- Aim : Create & Analyze the class diagram for ~~online food ordering system~~ library management system.

- Introduction:

Class diagram is a static structure diagram in UML that describes the classes of a system, their attributes, operations (methods), and the relationships among the classes.

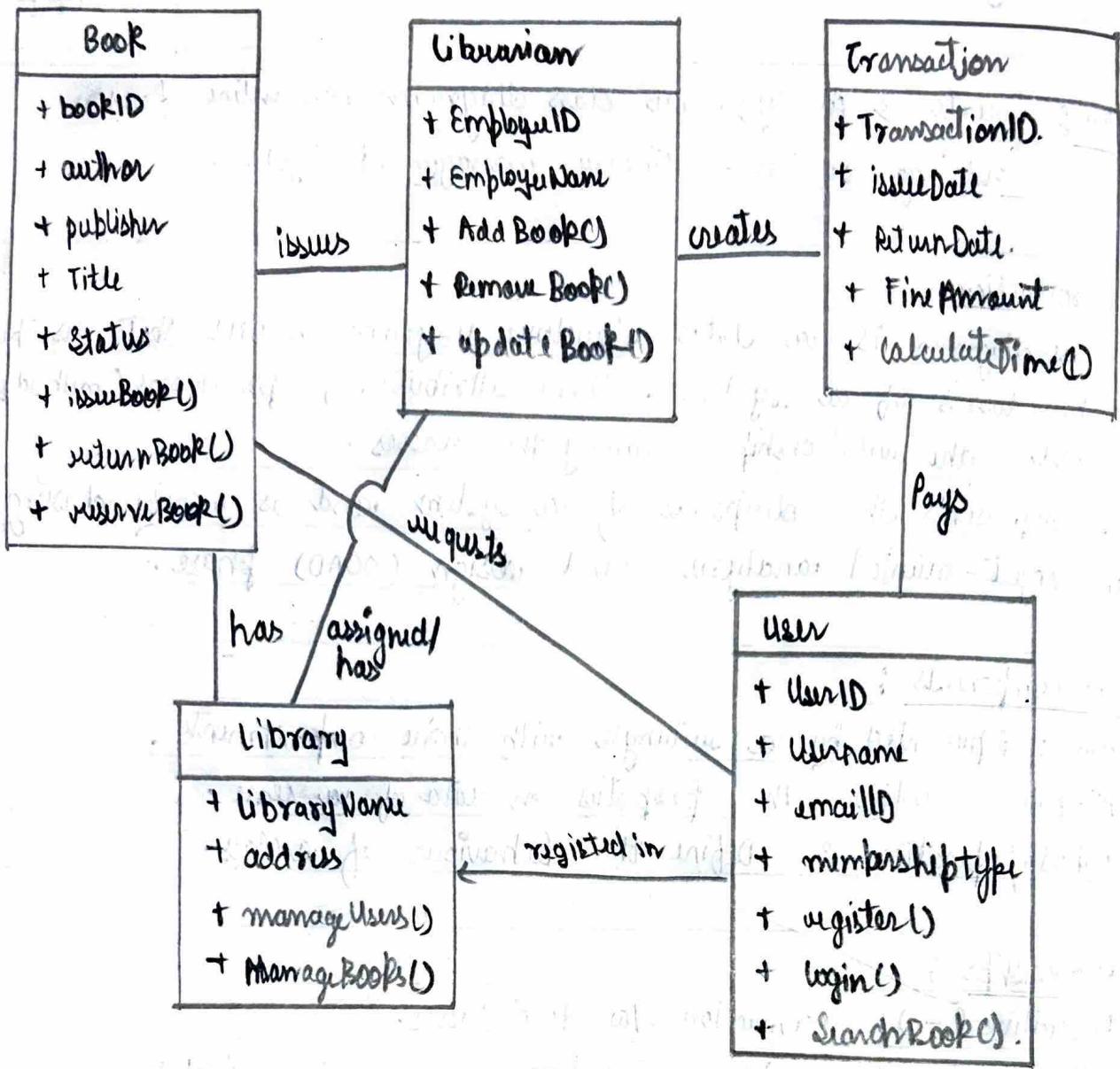
It represents the blueprints of a system and is widely used during the object-oriented analysis and design (OOAD) phase.

- Key components :

- 1) Class : Represented by a rectangle with three compartments.
- 2) Attributes : Defines the properties or data of a class.
- 3) Methods/Operations : Define the behaviour of a class.

- Relationships :

- 1) Association (-) : connection b/w two classes.
- 2) Multiplicity (1...*, 0...1) : defines how many objects participates
- 3) Aggregation (◆) : "Has a" relationship.
- 4) Composition (◆) : Strong form of aggregation.
- 5) Inheritance (▷) : A child class inherits from a parent class.



Class Diagram for Library Management System

Class diagram : library management system.

• Classes & their details:-

1) User :

- Attributes : userId, name, emailId, membershipType.
- Methods : register(), login(), searchBook().

2) Book :

- Attribute : bookId, title, publisher, author, status.
- Methods : issueBook(), returnBook(), reserveBook().

3) Librarian :

- Attributes :- employeeID, EmployeeName.
- Methods : AddBook(), RemoveBook(), updateBook().

4) Transaction

- Attributes : TransactionID, issueDate, returnDate, fineAmount.
- Methods : calculateFine();

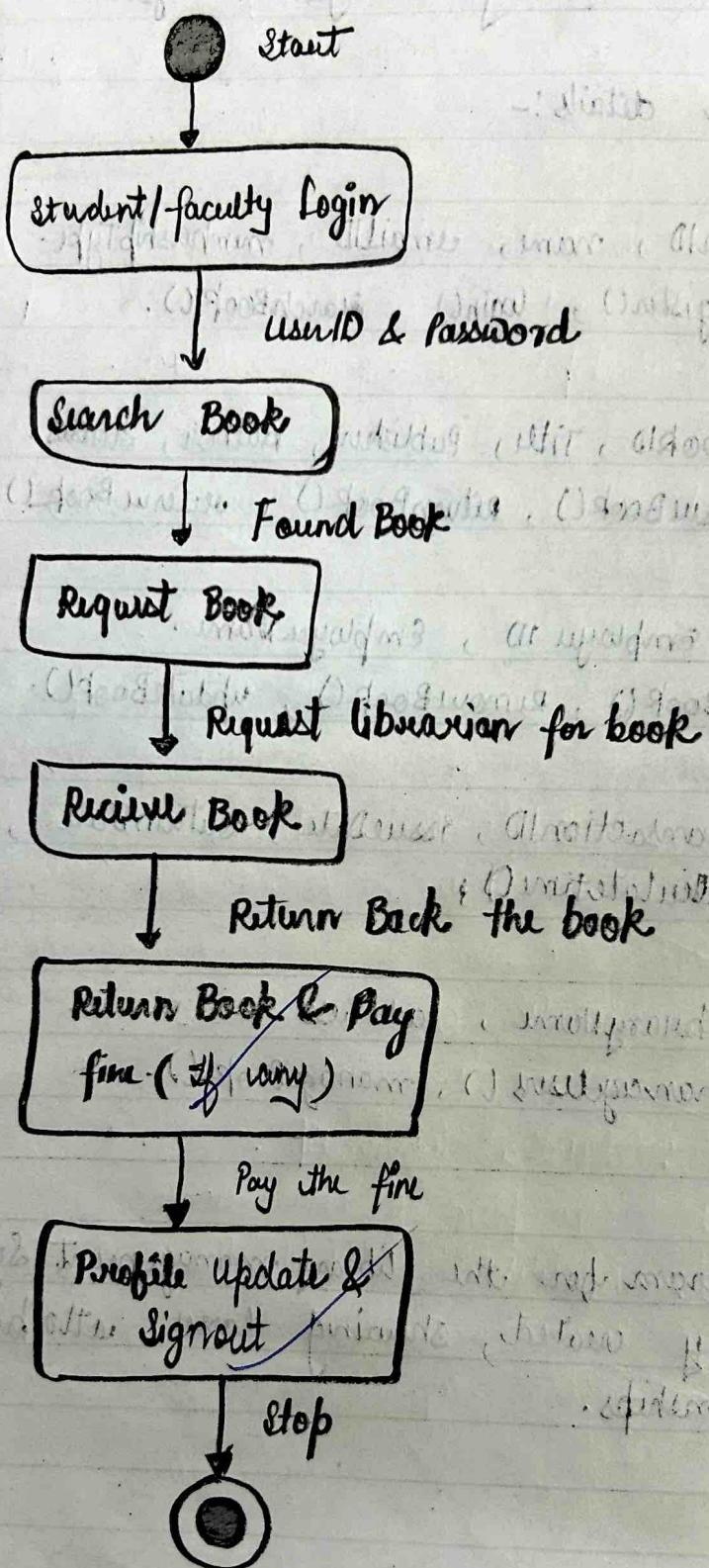
5) Library :

- Attributes : libraryName, address.
- Methods : manageUsers(), manageBooks().

• Result :-

The class diagram for the library management system was successfully created, showing classes, attributes, operations and relationships.

State Chart Diagram - Library Management System



Aim: To draw & study the static chart diagram for library management system.

theory:

A static chart diagram (also known as a static machine diagram) is a type of behavioral diagram in the unified modeling language (UML). It is primarily used to describe the different states of an object and the transitions b/w these states depending on events, actions or conditions.

unlike an activity diagram, which focuses on the workflow of the entire system or process, a state chart diagram focuses on a single object and shows how that object behaves and changes its state throughout its lifecycle.

~~Key components of Static chart diagram:-~~

1) State: Represented by rounded rectangle.

Ex: Available, issued, reserved.

2) Initial State: Represented by a filled black circle (●). It denotes the start.

3) Final State: Represented by a bullseye symbol (○). It denotes end of the lifecycle.

4) Transition: A directed arrow connecting one state to another, triggered by an event/action.

5) Event: A stimulus that triggers a state change.

Ex: issueBook(), returnBook().

6) Action: The activity performed due to transition.

7) Decision Point: A diamond shape used when the system must choose multiple transition based on conditions.

Procedure:

1. Identify the object for which the state diagram is to be created.
Ex: In this case the object is Book.
2. List all possible states the object can have during its lifecycle.
Ex: Available, Issued, Reserved, Returned, Unavailable
3. Determine the transition between states.
 - A book can move from Available \rightarrow Issued when borrowed.
 - A book can move from Available \rightarrow Reserved, when reserved.
similarly many transitions.
4. Add initial & final state:
 - The book starts at the initial state.
 - It may end in a final state when marked unavailable.
5. Draw diagram using UML notation.

Result:

The state chart diagram for the Library Management System was successfully designed. It represents the diff. states of a book object such as Available, Reserved, Issued, Returned & Unavailable along with transitions like issue, return, reserve, cancel and remove.

Aim: Draw the object diagram for library management system.

- Introduction:

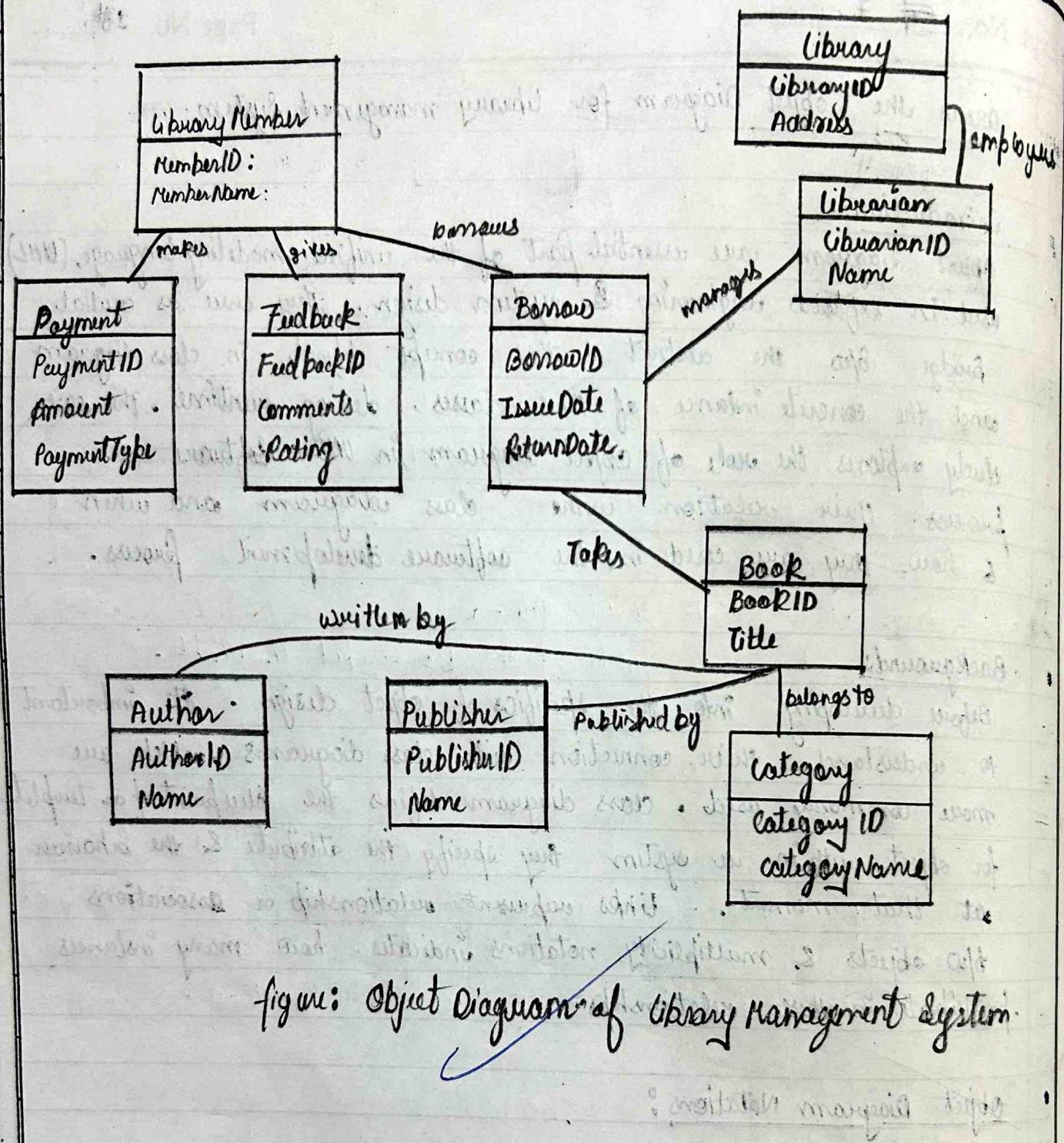
Object diagrams are essential part of the unified modeling language (UML) used in software engineering & system design. They serve as crucial bridge b/w the abstract & the concept defined in class diagram and the concrete instance of those classes during runtime. This case study explores the role of object diagram in UML. ~~Software dev. process~~ their relation with class diagram and when & how they can be used in the software development process.

- Background:

Before developing into the specifics of object design. It's important to understand their connection with class diagrams, which are more commonly used. A class diagram defines the blueprint or template for objects within a system. They specify the attributes & the behaviour at that moment. Links represent relationships or associations b/w objects & multiplicity notations indicate how many instances participate in these relationships.

- Object Diagram Notations:

- aggregation.
- Dependency Relationship.
- composition.
- link.
- Object Ben.
- Association.



library management system object diagram Analysis.

■ Object boxes :

- 1) library numbers.
- 2) Payment.
- 3) Feedback
- 4) library
- 5) librarian.
- 6) book
- 7) Author
- 8) Publisher
- 9) category.

■ Associations :

- 1) LibraryMember ↔ Payment. [number makes an payment]
- 2) LibraryMember ↔ feedback.
- 3) Librarian ↔ library.
- 4) Borrow ↔ Book.
- 5) Book ↔ Author.
- 6) Book ↔ Publisher.
- 7) Book ↔ category.

~~7/10/15/15~~

Aim : To draw and analyse the sequential diagram of library management system.

Theory:

Sequential Diagram is an interaction diagram that details how operations are carried out. They capture the interaction b/w objects in the context of call collaboration. Sequential diagram uses time focuses & show how they order the interactions visually by using the vertical axis of diagram to represent time when messages are sent & when.

Purpose:

- Models high level interactions b/w active objects in a system.
- Model the interaction b/w object instances within a collaboration that realizes a use case.
- Model the interaction b/w objects within a collaboration that realizes an operation.
- Either Model Generic interaction or specific instance of interaction.

Notations:

- | | |
|-----------------------|-----------------------|
| 1) Actors | 9) Destroy message. |
| 2) lifeline. | 10) Duration message. |
| 3) Activations. | 11) Note comment. |
| 4) Call Message. | |
| 5) Return Message. | |
| 6) Self Message. | |
| 7) Recursive Message. | |
| 8) Create Message. | |

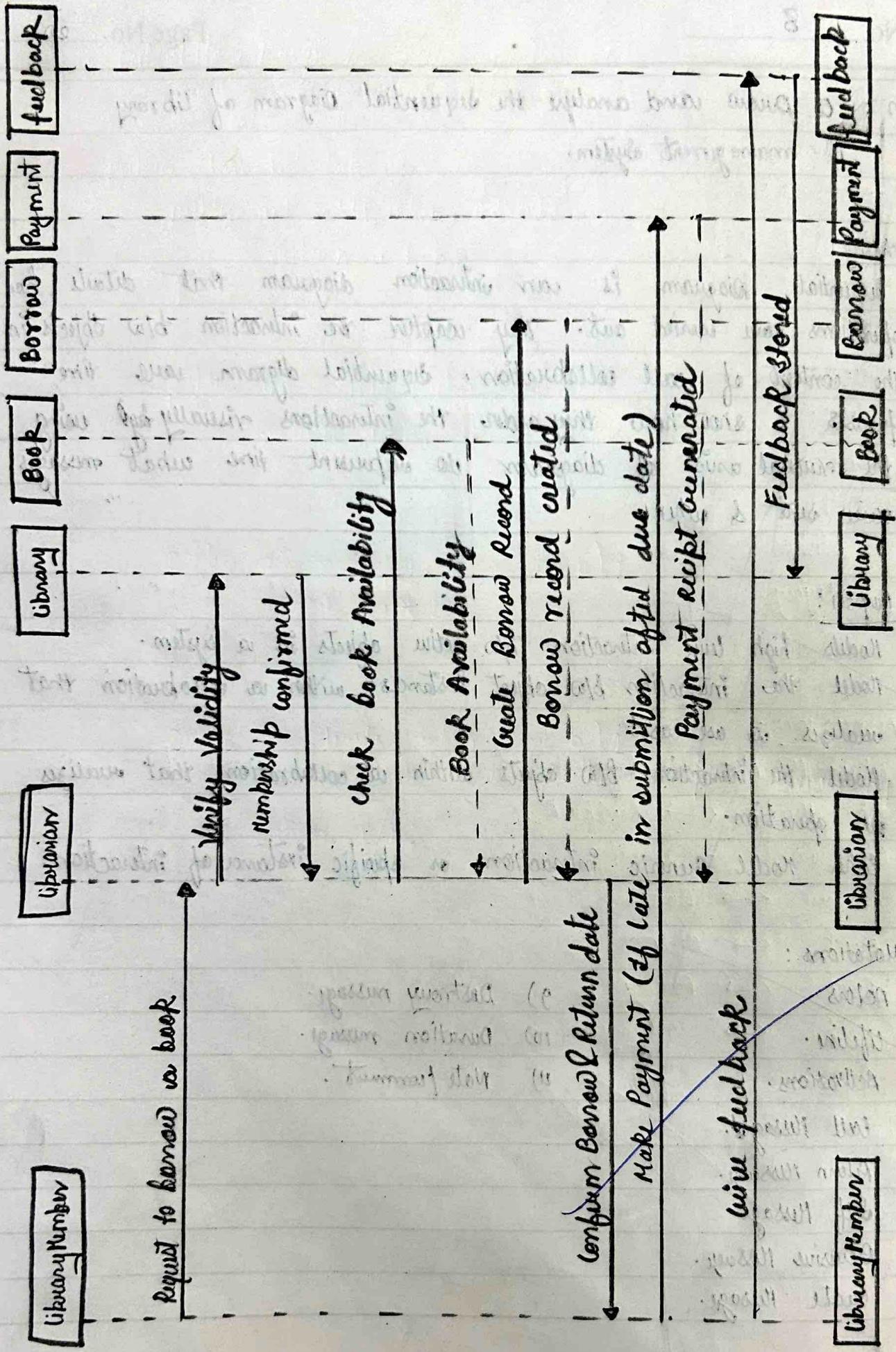


figure :- sequence Diagram - library management system

■ Sequential Diagram of library management system.

g) Actor :
library Member .

■ Participants .

- Librarian .
- Borrow .
- Library .
- Payment .
- Book .
- feedback .

■ Sequence flow :

- 1) Initialization : When library member request to borrow book .
- 2) Validation : Librarian validates the member account with the library .
- 3) Activity check : Librarian ensures book is available .
- 4) Transaction creation : Borrow record is generated .
- 5) Notification : The Member gets notified about due date .
- 6) Payment & feedback : The Member makes a payment if necessary & provides feedback about the experience .

~~Year 2010~~