Experiment → 5

Aim → Explain class diagram with symbols/Notations.

Class diagram → The class diagram shows the building blocks of any object-oriented system. Class diagrams defind a static view of the model, or part of the model, describing what attributes and behaviour it has rather than detailing the methods for achieving operations. Class diagrams are most useful in illustrating relationships between classes and interfaces. Generalizations, aggregations and associations are all valuable in reflecting inheritance, composition or usage, and connections respectively.

Symbols of class diagram :

UML class is represented by the following figure. The diagram is divided into four parts.

- The top section is used to name the class.
- The second section is used to describe the operations performed by the class.

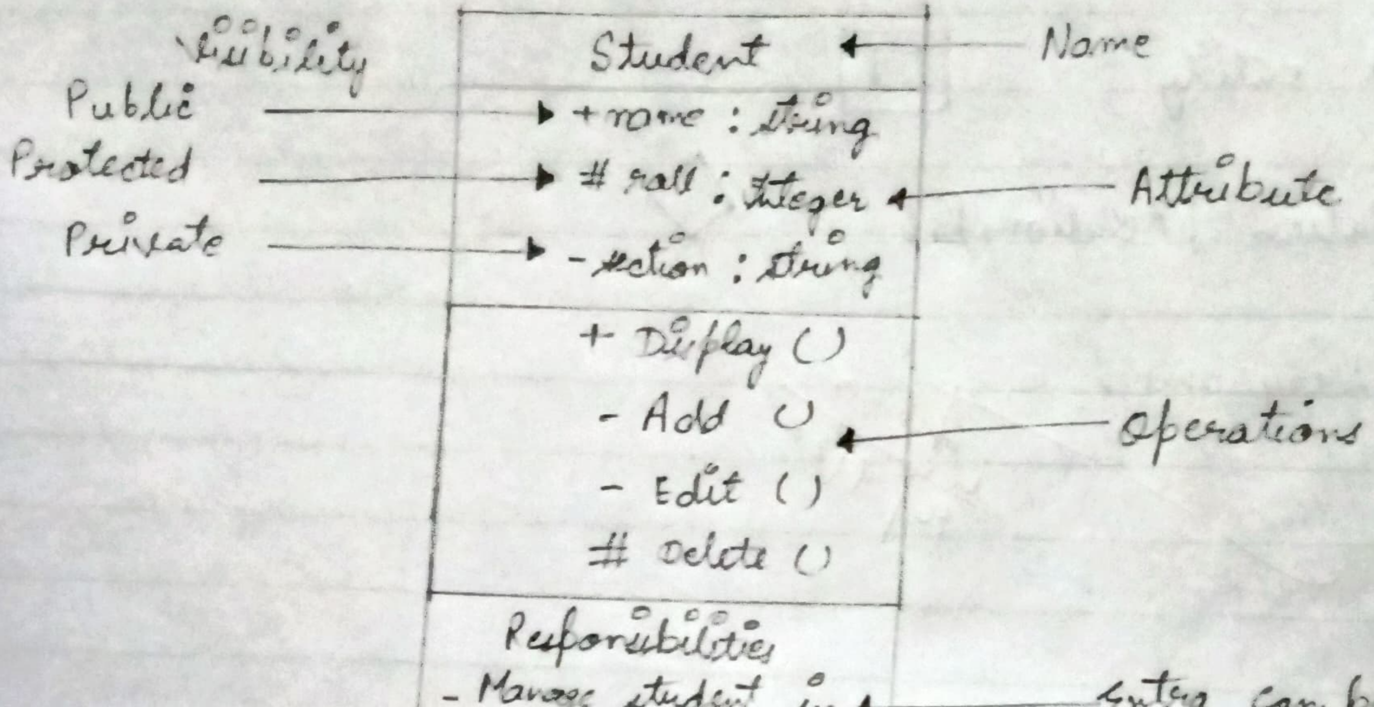
Generalization : A generalization is used to indic inheritance. Draw from the specific classifier to a general classifier, the generalize implication is that the source inherits the target's ~~of~~ characteri

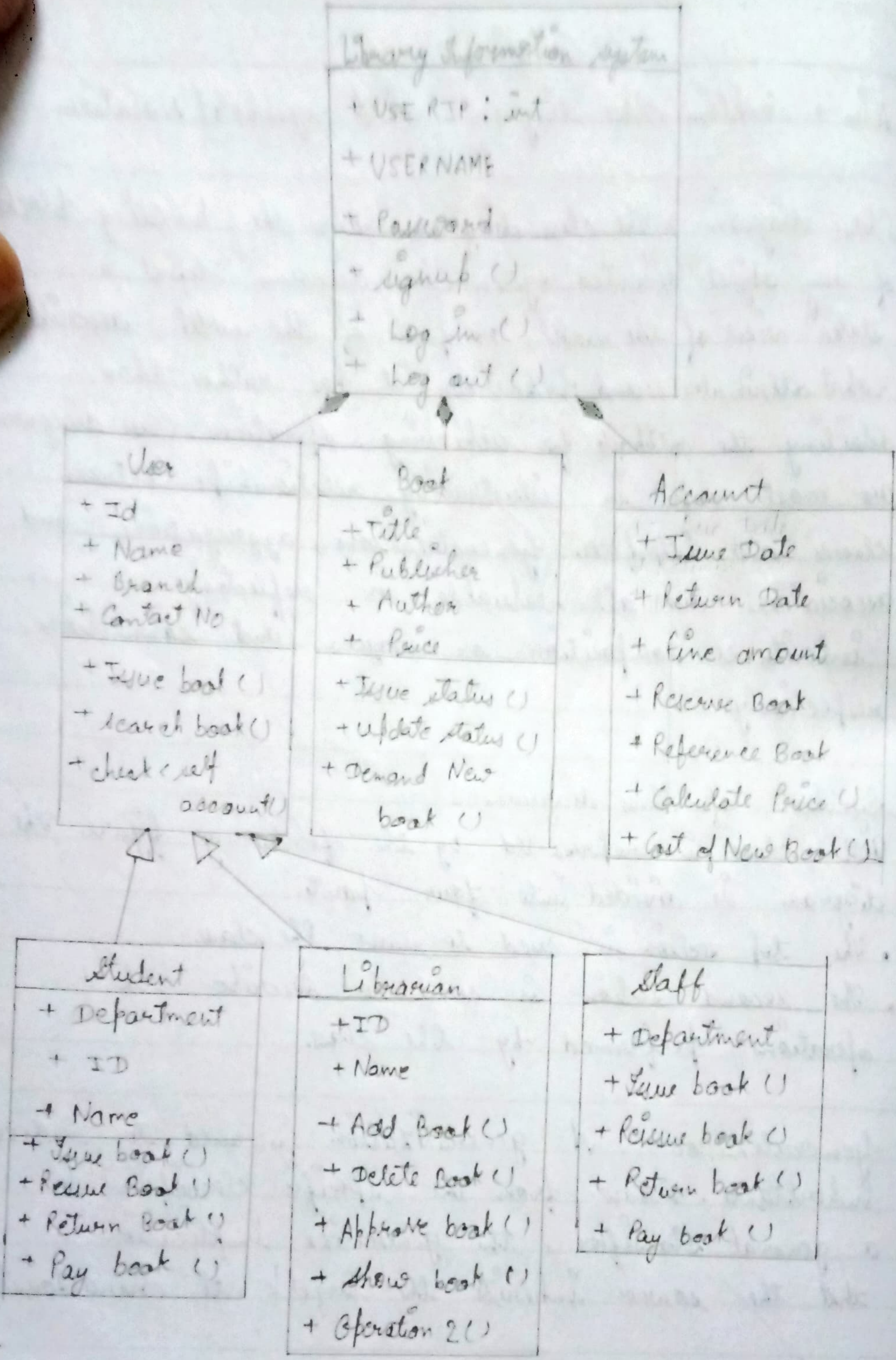
Aggregation are Composition are subsets of association meaning they are specific case of association. In both aggregation and composition object of one class "owns" objects of another class But there is a subtle difference :

Aggregation → Implies a relationship where the child can exist independently of the parent.
Example → Class (parent) and student (child). Delete the class and the students still exist.

Composition → Implies a relationship where the child cannot exist independent of the parent.
Example → House (parent) and Room (child). Rooms ~~doe~~ don't exist seperate to a house.

CLASS


Visibility
Public
Protected
Private
Student
Name
+ name : string
# roll : Integer
- section : string
Attribute
+ Display ( )
- Add ( )
- Edit ( )
# Delete ( )
Operations
Responsibilities
- Manage student in
Entry Component


Library Information system
+ USERID : int
+ USERNAME
+ Password
+ signup ()
+ Log in ()
+ Log out ()
User
+ Id
+ Name
+ Branch
+ Contact No
+ Issue book ()
+ search book()
+ check self account()
Book
+ Title
+ Publisher
+ Author
+ Price
+ Issue status ()
+ Update status ()
+ Demand New book ()
Account
+ Issue Date
+ Return Date
+ fine amount
+ Reserve Book
+ Reference Book
+ Calculate Price ()
+ Cost of New Book ()
Student
+ Department
+ ID
+ Name
+ Issue book ()
+ Reissue Book ()
+ Return Book ()
+ Pay book ()
Librarian
+ ID
+ Name
+ Add Book ()
+ Delete Book ()
+ Approve book ()
+ Show book ()
+ Operation 2 ()
Staff
+ Department
+ Issue book ()
+ Reissue book ()
+ Return book ()
+ Pay book ()

Experiment → 6

Aim :- (A) Explain Deployement diagram with symbols.

Deployement diagram is one of the types of UML diagram which is used to define the hardware requirements for the particular project product to execute the software : basically, it maps the software design requirement to the physical system which executes the software design and visualize how software interacts with hardware.

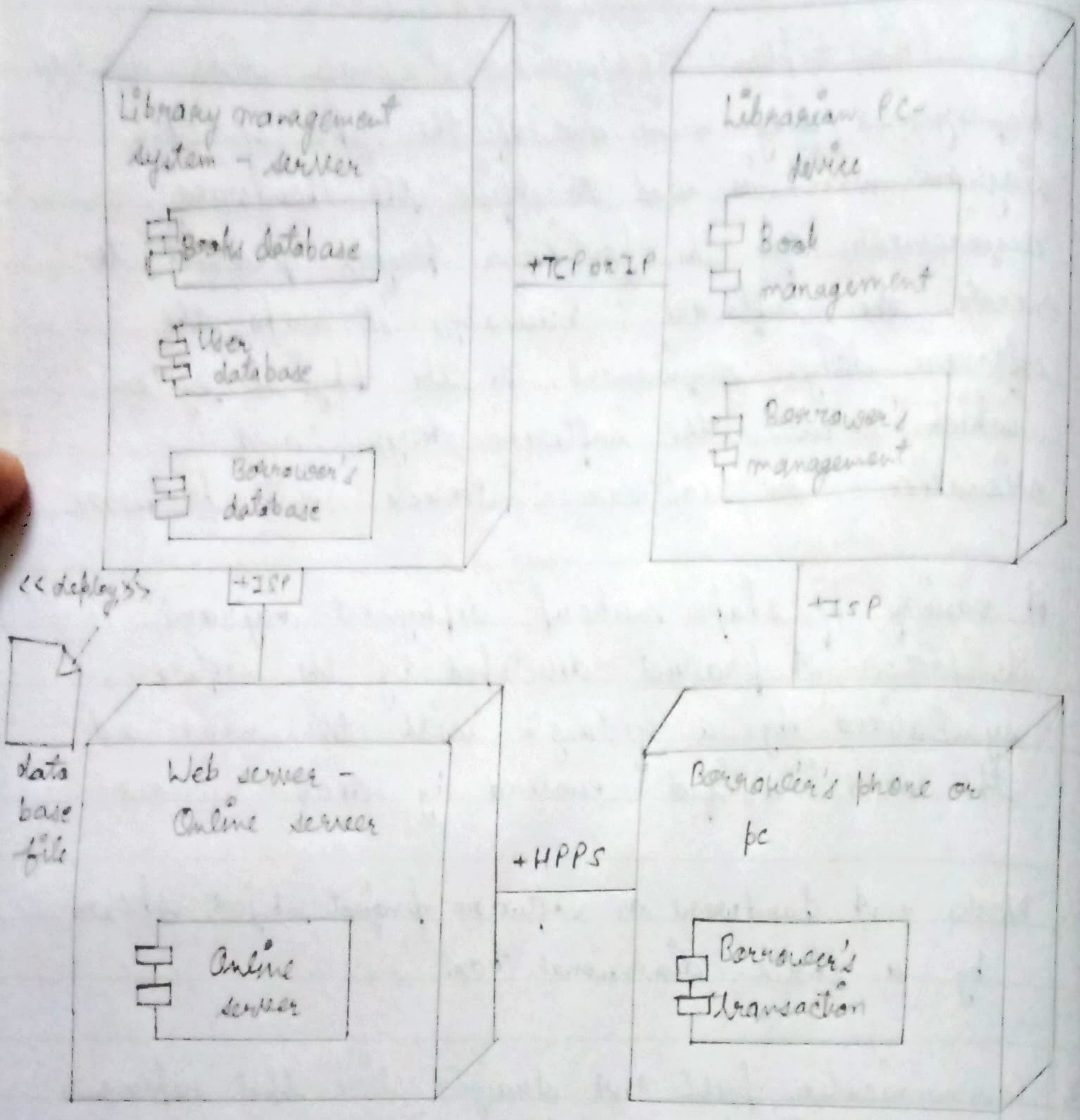
A variety of shapes makeup deployment diagrams:

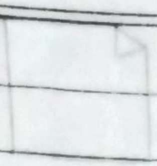
1. Artifact → A product developed by the software, symbolized by a rectangle with the name and the word "artifact" enclosed by double arrows.

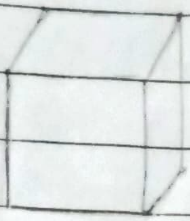
2. Nodes → A hardware or software ~~project~~ object, shown by a three-dimensional box.

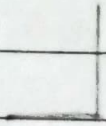
3. Communication path → A straight line that represents communication between two device nodes.

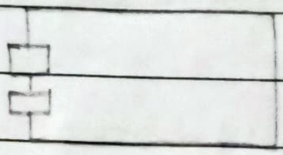
4. Component → A rectangle with two tabs that indicates a Hardware / software element.


Library management system - server
Books database
User database
Borrower's database
+TCP or IP
Librarian PC - device
Book management
Borrower's management
+ISP
<<deploy>>
data base file
+ISP
Web server - Online server
Online server
+HPPS
Borrower's phone or pc
Borrower's transaction

1. Artifact → 

2. Node → 

3. Communication path → 

4. Component → 

(b) Draw a Deployment diagram.

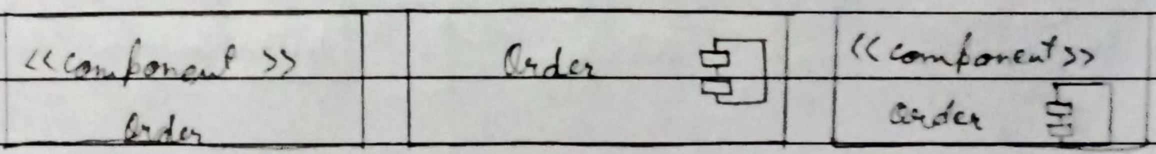
←

Experiment → 7

(A) Explain component Diagram with symbols / notations
Component represents a modular part of a system →
A component defines its behaviour in terms of provided and required interfaces. Package is used to group elements, and to improve a namespace for the group elements.

Notation / symbols of component diagram → A component represents a modular part of a system that encapsulates its contents and whose manifestation is replaceable within its environment.
A high-level, abstracted view of a component in UML can be modeled as:-

1. A rectangle with the component's name
2. A rectangle with the component icon
3. A rectangle with the stereotype text and/or icon.
4. rectangle with the stereotype text and/or icon.



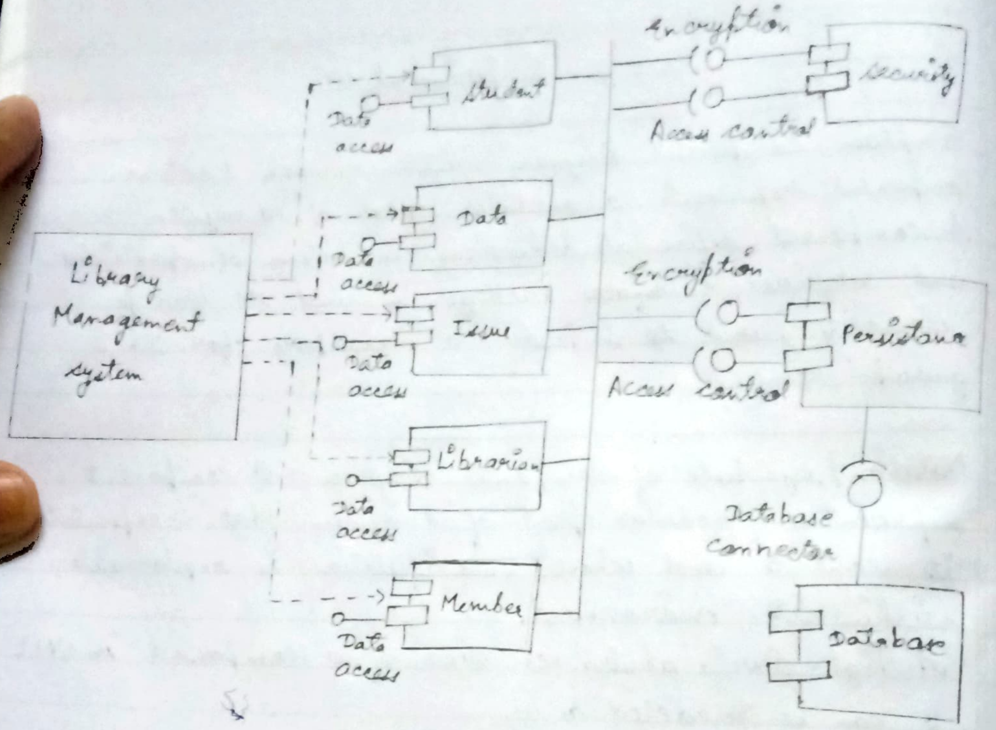
Interface : Interfaces in component diagrams show how components are wired together and interact with each other. The assembly connector allows linking the component's required interface with the provide interface of another component. This shows that one component is providing the service that the other is requiring.

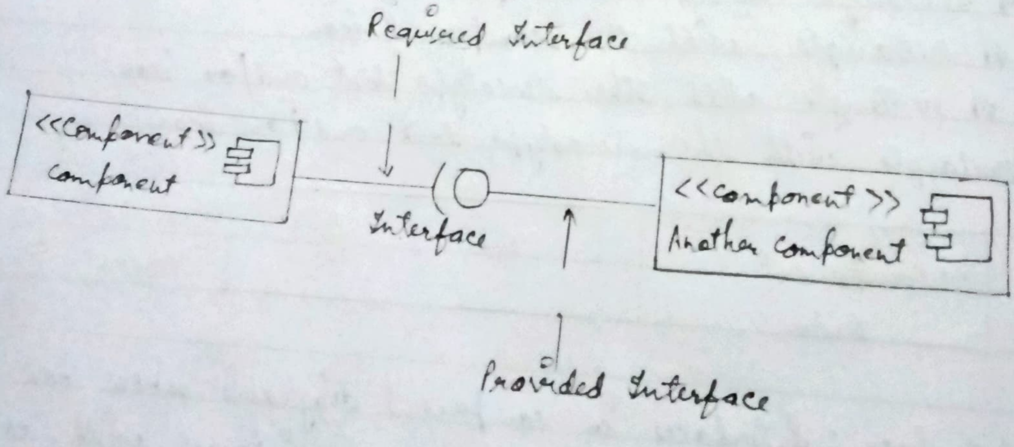
This is Component diagram of Library Management system which shows components. provided and required interfaces. Ports and relationship between the issues, student, librarian, member and Address. This type of Diagrams is used in component based Development to describe systems with service oriented Architecture. LIBRARY MANAGEMENT SYSTEM UML COMPONENT DIAGRAM describes organization and wiring of the physical component.

Provided Interface symbols with a complete circle at their end represent an interface that the component provides- the "lollipop" symbol is shorthand for a realization relationship of an interface classifier.
Required Interface symbols with only a half circle at their end (a.k.a. sockets) represents an interface that the component requires (in both cases, the interface's name is placed near the interface symbol itself).

Dependency : A dashed line that ends in an arrow, which indicates that one node or component is dependent on another.

(B) ~~A~~ Draw component diagram of Library management system -





POORNIMA

of another
is providi

This is Comp
which sh
interfaces.
Student
Diagrams
to descri
LIBRARY
describes
compon

Provided
end refer
the "
relations
Required
their e
compon
same

Depende
indicat
on

(B)* Draw
sy