

## Experiment No. 8

Aim:	To illustrate the concept of Deployment model in UML.
Objective:	To understand & implement a deployment diagram in UML that shows the architecture of a software system in terms of hardware (nodes) and the software (artifacts) deployed on them.

## Experiment No. 8

(52)

Aim:	To illustrate the concept of Deployment model in UML.
Objective:	To understand and implement a deployment diagram in UML that shows the architecture of a software system in terms of hardware (nodes) and the software (artifacts) deployed on them.
About:	A deployment diagram in UML models the physical deployment of artifacts (software) on nodes (hardware). It is used to visualize the distribution of components across different pieces of infrastructure & is especially useful for modeling client-server, web-based or distribution systems.
Key Elements:	<p>i) Nodes: Represent physical devices or execution environments (eg: server DB, client device).</p> <p>ii) Artifacts: Represent deployable pieces of software (eg: executables, libraries, WAR files).</p> <p>iii) Dependencies: Indicate communication or usage relationships between nodes and artifacts.</p>



--> Deployment diagram help to :

- i> Understand the physical deployment structure
- ii> Allocate software components to hardware
- iii> Support performance analysis, load balancing & infrastructure planning.

Example

Scenario : Web application deployment model :

A> Deployment Components :

- i> Client Node : Represents the user's browser or mobile device
- ii> Web Server Node : Hosts the front-end & handles the HTTP requests.
- iii> Application Server : Contains the business logic & processes
- iv> Database Server Node : Stores application data

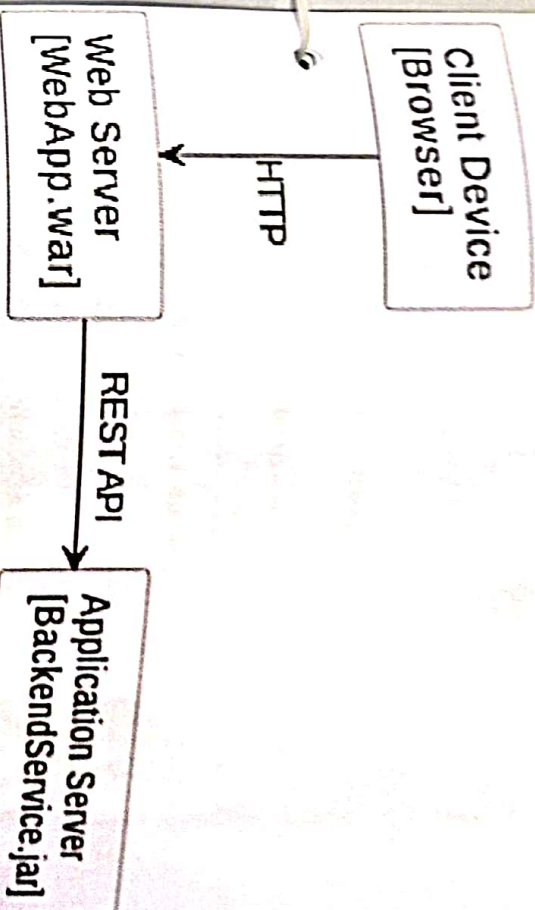
B> Artifacts :

- i> Webapp. war deployed on web server.
- ii> BackendService.jar deployed on the Application server
- iii> MySQL DB schema holds the DB Server.

c> Connections :

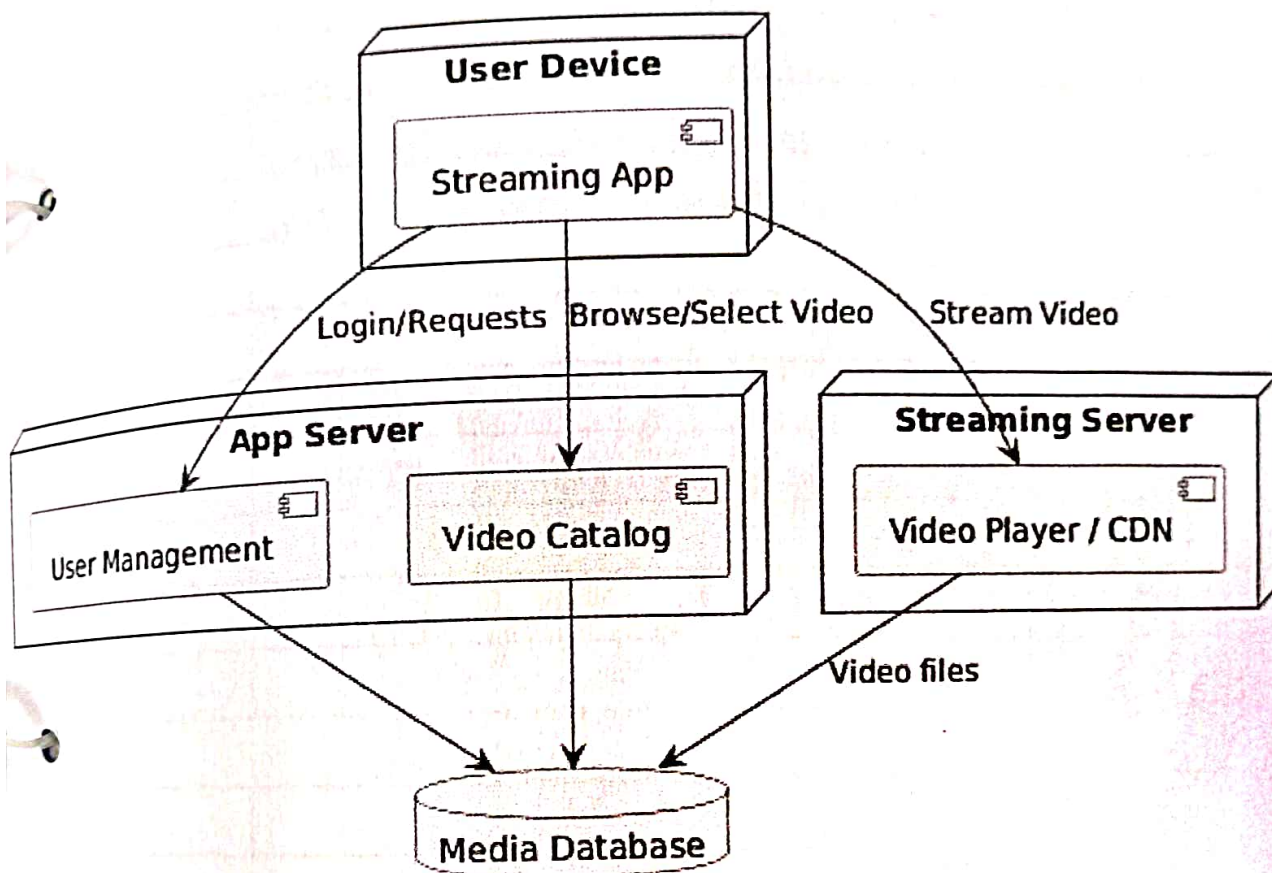
- i> Client connects to Web-Server via HTTP.

**Conclusion:** Deployment diagrams in UML are crucial for visualizing the hardware / software mapping in a system. Through this experiment, we understood how components are physically distributed & communicate within a software architecture.



\* Deployment diagram:  
deployment of a software  
(global deployment) \*





ii) Web server communicates with application server via internal API's

iii) Application server connects to DB server using JDBC.

Conclusion: Deployment diagrams in UML are crucial for visualizing the hardware / software mapping in a system. Through this experiment, we understood how components are physically distributed & communicate within a software architecture.

8/4/28  
23/4/28  
A 28/8  
3/4/28  
3/4/28