

**Experiment No. 7. Design Component diagram for Hospital Management System.**

**Aim:** To understand the concept of Component diagram.

**Theory:**

A Component diagram depicts how components are wired together to form larger components or software systems. They are used to illustrate the structure of arbitrarily complex systems. It shows the organization and dependencies among a set of components. Component diagrams can be used to model the static view of a system. Component diagrams are used during the implementation phase of an application. However, it is prepared well in advance to visualize the implementation details.

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.

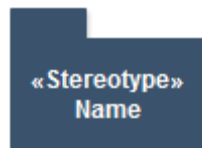
**Component Diagram Symbols:**

**Component:** Defines system behavior in terms of provided and required interfaces.



**Fig. 7.1: Component**

**Package:** It used to group elements, and to provide a namespace for the grouped elements.



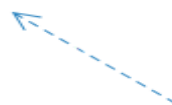
**Fig. 7.2: Package**

**Package container:** It is used to define UML elements such as classes, use cases, and Components.



**Fig. 7.3: Package container**

**Dependency:** It is a relationship in which one element depends on another element.

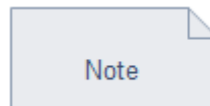
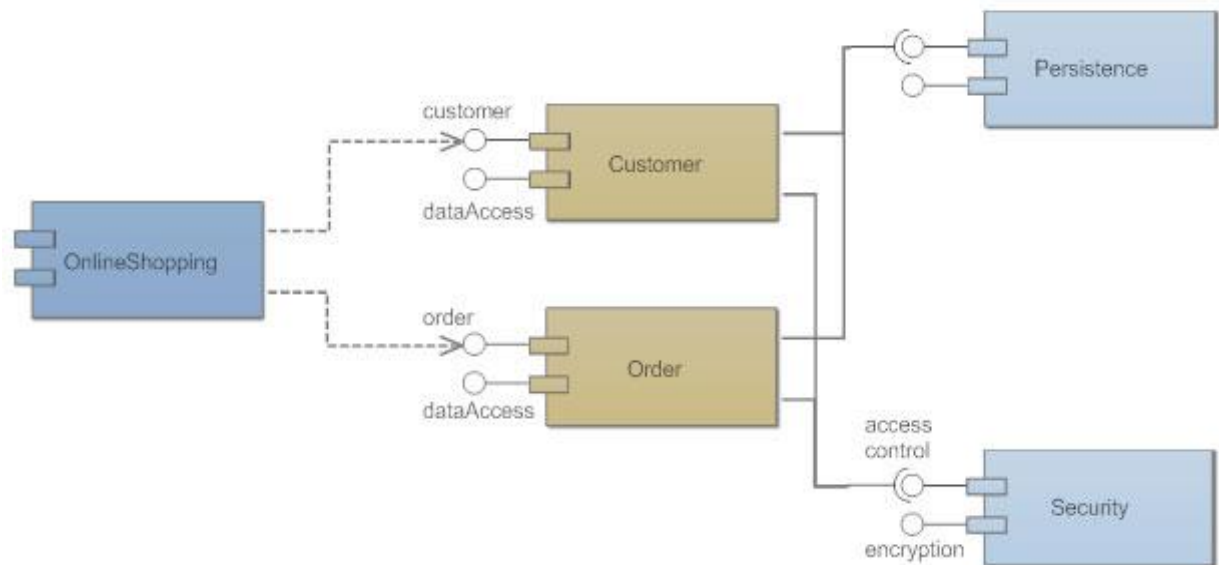


**Fig. 7.4: Dependency**

**Generalization:** It is a relationship in which one model element (the child) is based on another model element (the parent).

**Fig. 7.5: Generalization**

**Note:** It contains comments or textual information.

**Fig. 7.6: Note****Fig. 7.7: Component diagram for Online Shopping**

### Hospital Management System:

The project “Hospital Management System” is aimed to develop to maintain the day –to-day state of admission/discharge of patients, list of doctors, reports generation, and etc. It is designed to achieve the following objectives:

1. To computerize all details regarding patient details & hospital details.
2. Scheduling the appointment of patient with doctors to make it convenient for both.
3. Scheduling the services of specialized doctors, so that facilities provided by hospital are fully utilized in effective and efficient manner.
4. It should be able to handle the test reports of patients conducted in the pathology lab of the hospital.
6. The inventory should be updated automatically whenever a transaction is made.
7. The information of the patients should be kept up to date and their record should be kept in the system for historical purposes.

**Viva Question:**

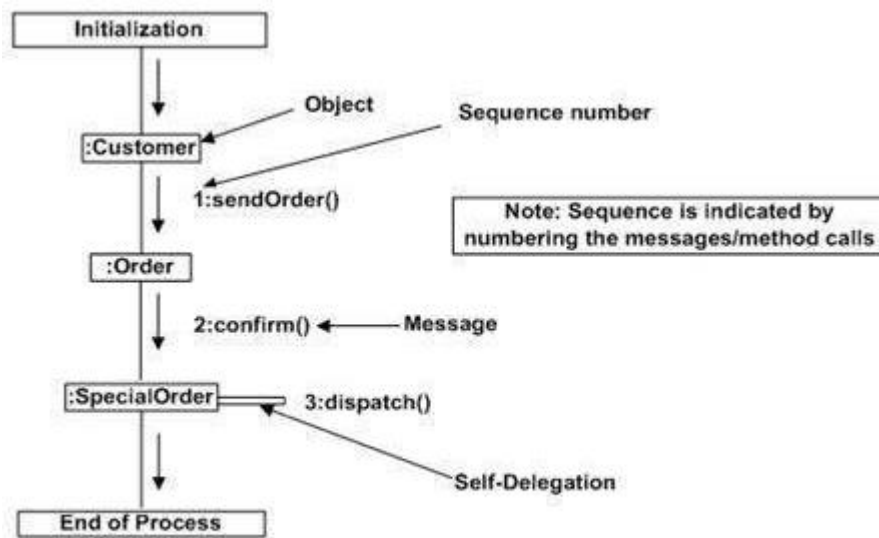
1. What is Component?
2. How to identify the Component of the system?
3. What is the main purpose of Component diagram?
4. What do you mean by dependency?
5. Explain dependencies in Component diagram?

**Experiment No. 8. Design Collaboration diagram for Automated Banking System.**

**Aim:** To understand the concept of Collaboration diagram.

**Theory:**

The Collaboration diagram is an interaction diagram. It shows the object organization of system. Collaboration diagrams show the message flow between objects in an object oriented application, and also imply the basic associations (relationships) between classes. In the Collaboration diagram, the method call sequence is indicated by some numbering technique. The number indicates how the methods are called one after another. Method calls are similar to sequence diagram. However, difference being the sequence diagram does not describe the object organization, whereas the Collaboration diagram shows the object organization.



**Fig. 8.1 Collaboration diagram for Order Management System**

**Elements of a Collaboration diagram**

A Collaboration diagram consists of the following elements:

Element and its description	Symbol
<b>Object:</b> The objects interacting with each other in the system. Depicted by a rectangle with the name of the object in it, preceded by a colon and underlined.	<u>ObjectName</u>
<b>Relation/Association:</b> A link connecting the associated objects. Qualifiers can be placed on either end of the association to depict cardinality.	0 * <u>                    </u> 1 *
<b>Messages:</b> An arrow pointing from the commencing object to the destination object shows the interaction between the objects. The number represents the order/sequence of this interaction.	1:Function() →

**Viva Question:**

1. What is reverse engineering?
2. What is the purpose of Collaboration diagram?
3. How to represent object in Collaboration diagram?
4. What do you mean by assembly connector?
5. What is the difference between Sequence diagram and Collaboration diagram?

**Experiment No. 9. Study about the Software Cost Estimation Model.**

**Aim:** To understand the concept of Software Cost Estimation Model.

**Theory:**

Cost estimation can be defined as the approximate judgments of the costs for project. Cost estimation will never be an exact science because there are too many variables involved in the calculation for cost estimation, such as human, technical, environment and political.

Cost estimation is usually measured in terms of effort. The effort is the amount of time for one person to work for a certain period of time. It is important that the specific characteristics of the development environment are taking into account when comparing the effort of two or more projects because no two development environments are the same.

Cost estimation is an important tool that can affect the planning and budgeting of a project because there are a finite number of resources for a project. A cost estimation done at the beginning of a project will help us to determine which features can be included within the resource constraints of the project (time). Requirements can be prioritized to ensure that the most important features are included in the product.

**Software Cost Estimation Model:****COCOMO:**

The Constructive Cost Model (COCOMO) is a procedural software cost estimation model. COCOMO is used to estimate size, effort and duration based on the cost of the software.

COCOMO predicts the effort and schedule for a software product development based on inputs relating to the size of the software and a number of cost drivers that affect productivity.

COCOMO has three different models that reflect the complexities:

- Basic Model – this model would be applied early in a projects development. It will provide a rough estimate early on that should be refined later on with one of the other models.
- Intermediate Model – this model would be used after you have more detailed requirements for a project.
- Advanced Model – when design of the project is complete you can apply this model to further refine your estimate.

Within each of these models there are also three different modes. The mode you choose will depend on your work environment, and the size and constraints of the project itself. The modes are:

- Organic – this mode is used for “relativity small software teams developing software in a highly familiar, in-house environment”.
- Embedded – operating within tight constraints where the product is strongly tied to a “complex of hardware, software, regulations and operational procedures”.
- Semi-detached – an intermediate stage somewhere in between organic and embedded. Projects are usually of moderate size of up to 300,000 lines of code.

**Equations Used**

There are two main equations that are used to calculate effort and schedule time (measured in months). They are:

Equation 1  $PM = a(KDSI)^b * EAF$

Equation 2  $TDEV = c(PM)^d$

Where:

- PM is effort in person-months
- EAF is the effort adjustment factor
- TDEV is the schedule time
- KDSI is the number of lines of code (in thousands)
- a, b, c, and d are all constants based on the mode you are using (refer to Table 1)

**Table Error! Bookmark not defined. – List of Constants Based on Mode**

Model	a	b	c	D
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

**Viva Question:**

1. What is COCOMO?
2. What do you mean by LOC?
3. How to calculate software cost?
4. Which basic metrics considered for cost estimation?
5. Why cost estimation is necessary in software development process?