SOFTWARE ENGINEERING PRACTICAL DOCUMENT

PRACTICAL NO: 7

AIM: Study and implementation of Component Diagrams.

SOLUTION

What is a Component Diagram?

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.

A component diagram is used to break down a large object-oriented system into the smaller components, so as to make them more manageable. It models the physical view of a system such as executables, files, libraries, etc. that resides within the node.

It visualizes the relationships as well as the organization between the components present in the system. It helps in forming an executable system. A component is a single unit of the system, which is replaceable and executable. The implementation details of a component are hidden, and it necessitates an interface to execute a function. It is like a black box whose behaviour is explained by the provided and required interfaces.

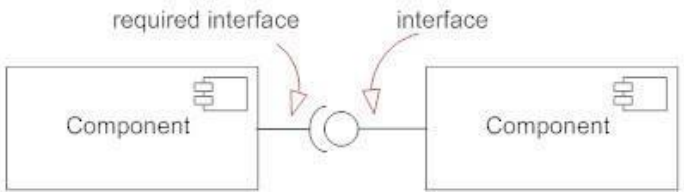
Basic Component Diagram Symbols and Notations

Component

A component is a logical unit block of the system, a slightly higher abstraction than classes. It is represented as a rectangle with a smaller rectangle in the upper right corner with tabs or the word written above the name of the component to help distinguish it from a class.

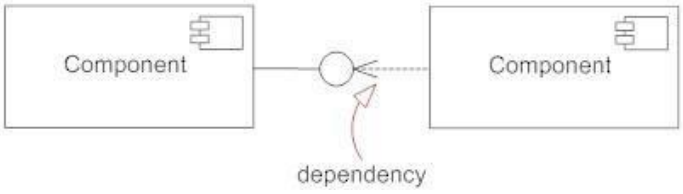
Interface

An interface (small circle or semi-circle on a stick) describes a group of operations used (required) or created (provided) by components. A full circle represents an interface created or provided by the component. A semi-circle represents a required interface, like a person's input.



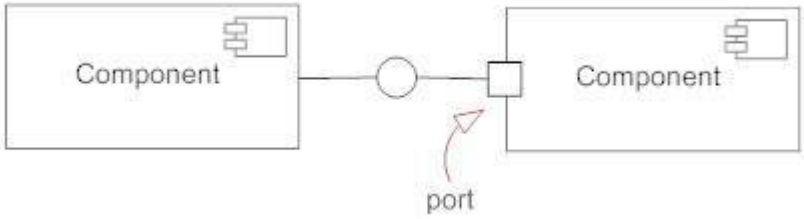
Dependencies

Draw dependencies among components using dashed arrows.



Port

Ports are represented using a square along the edge of the system or a component. A port is often used to help expose required and provided interfaces of a component.



Purpose of a Component Diagram

Since it is a special kind of a UML diagram, it holds distinct purposes. It describes all the individual components that are used to make the functionalities, but not the functionalities of the system. It visualizes the physical components inside the system. The components can be a library, packages, files, etc.

The component diagram also describes the static view of a system, which includes the organization of components at a particular instant. The collection of component diagrams represents a whole system.

The main purpose of the component diagram is enlisted below:

It envisions each component of a system.

It constructs the executable by incorporating forward and reverse engineering.

It depicts the relationships and organization of components.

Why use Component Diagram?

The component diagrams have remarkable importance. It is used to depict the functionality and behaviour of all the components present in the system, unlike other diagrams that are used to represent the architecture of the system, working of a system, or simply the system itself.

In UML, the component diagram portrays the behaviour and organization of components at any instant of time. The system cannot be visualized by any individual component, but it can be by the collection of components.

Following are some reasons for the requirement of the component diagram:

It portrays the components of a system at the runtime.

It is helpful in testing a system.

It envisions the links between several connections.

When to use a Component Diagram?

It represents various physical components of a system at runtime. It is helpful in visualizing the structure and the organization of a system. It describes how individual components can together form a single system. Following are some reasons, which tells when to use component diagram:

To divide a single system into multiple components according to the functionality.

To represent the component organization of the system.

EXAMPLE OF LIBRARY MANAGEMENT SYSTEM

The purpose of the Library Management system is to allow for storing details of a large number of books, magazines, Journals, thesis and allow for add, search, borrow, return facilities separately to administrator/Librarian, staff and students. Different privileges are given to different types of users.

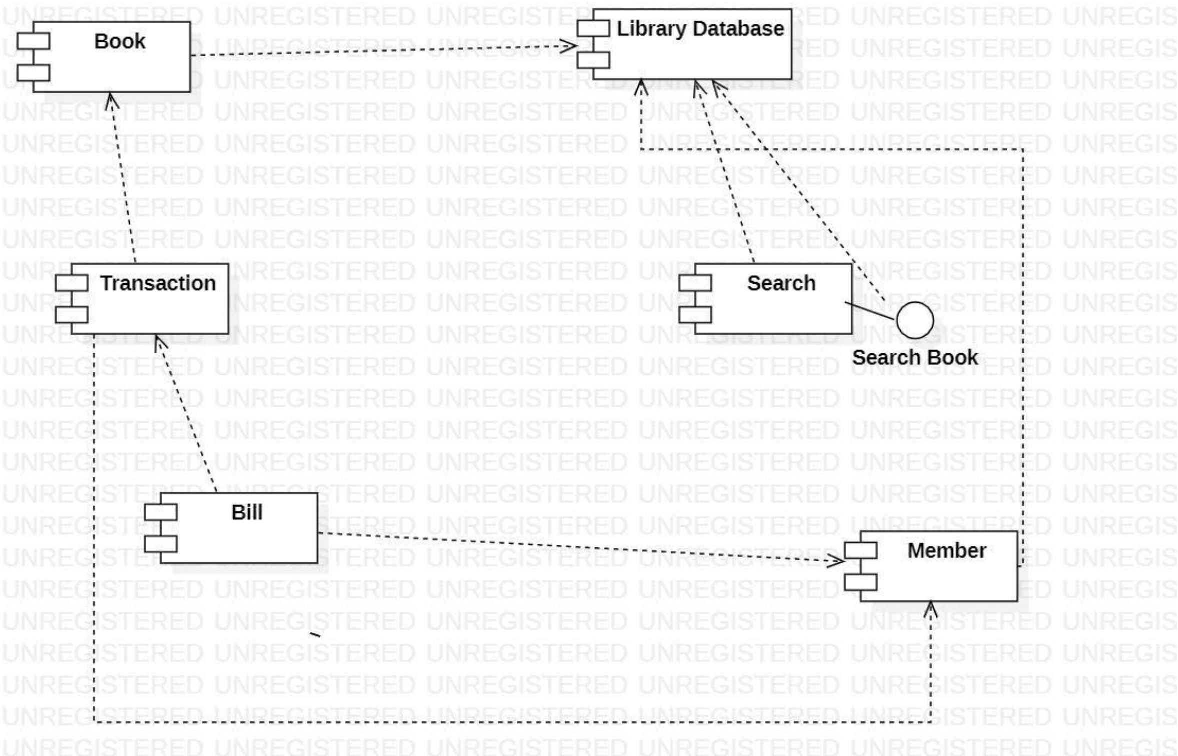
The tasks we have to do are:

1. You will have to identify the main entities (objects) for this system.

2. You will have to find out the relationships between these objects.

3. You will have to find the necessary attributes and functions that need to be associated with each object to implement the functionality mentioned above.

4. You will make a final comprehensive diagram show and all objects and their relations along with their attributes and functions**.**



Library Database
Book
Transaction
Search
Search Book
Bill
Member