



EXPERIMENT - 8

Software Engineering Lab

Aim

To perform the implementation view diagram: Component diagram for the system.

Syeda Reeha Quasar

14114802719

4C7

EXPERIMENT – 8

Aim:

To perform the implementation view diagram: Component diagram for the system.

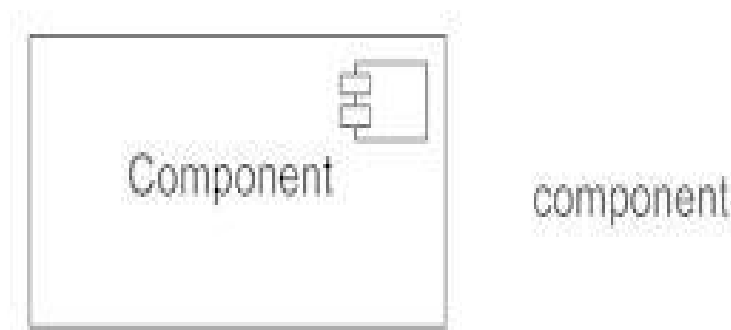
Theory:

A component diagram provides a physical view of the system. Its purpose is to show the dependencies that the software has on the other software components (e.g., software libraries) in the system. The diagram can be shown at a very high level, with just the large-grain components, or it can be shown at the component package level. [Note: The phrase component package level is a programming language-neutral way of referring to class container levels such as .NET's namespaces (e.g., System.Web.UI) or Java's packages (e.g., java.util).

Basic Component Diagram Symbols and Notations

Component

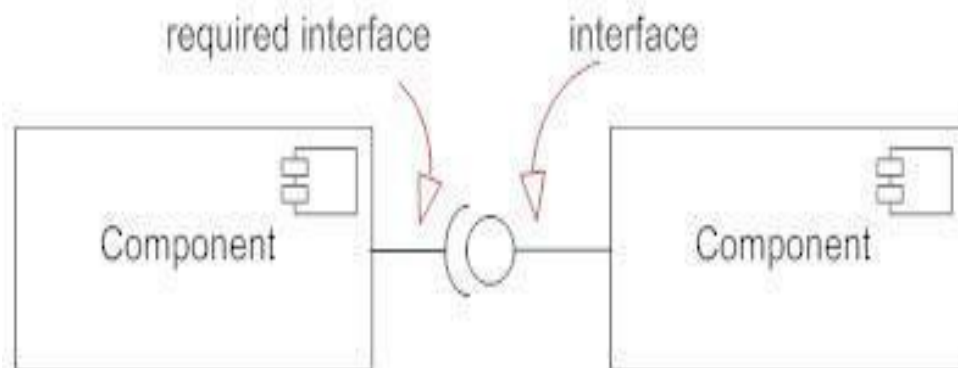
A component is a logical unit block of the system, a slightly higher abstraction than classes. It is represented as a rectangle with a smaller rectangle in the upper right corner with tabs or the word written above the name of the component to help distinguish it from a class.



Interface

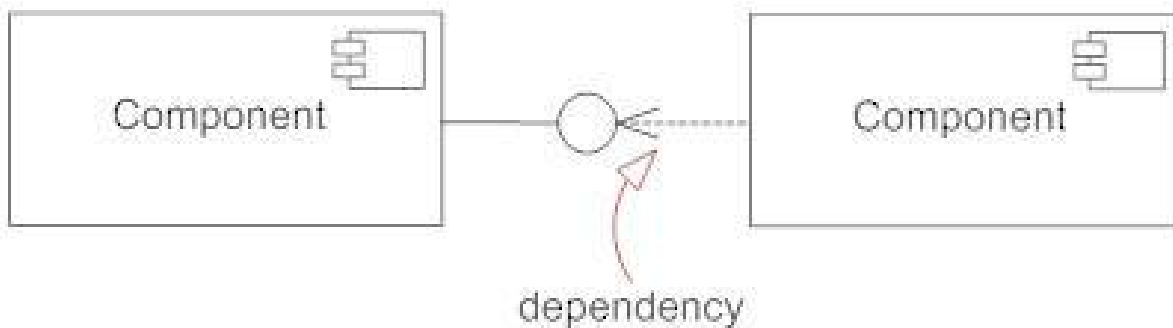
An interface (small circle or semi-circle on a stick) describes a group of operations used (required) or created (provided) by components. A full circle represents an interface

created or provided by the component. A semi-circle represents a required interface, like a person's input.



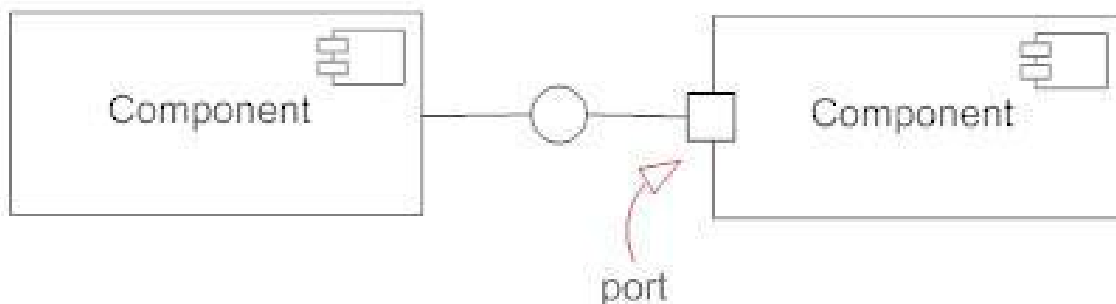
Dependencies

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.

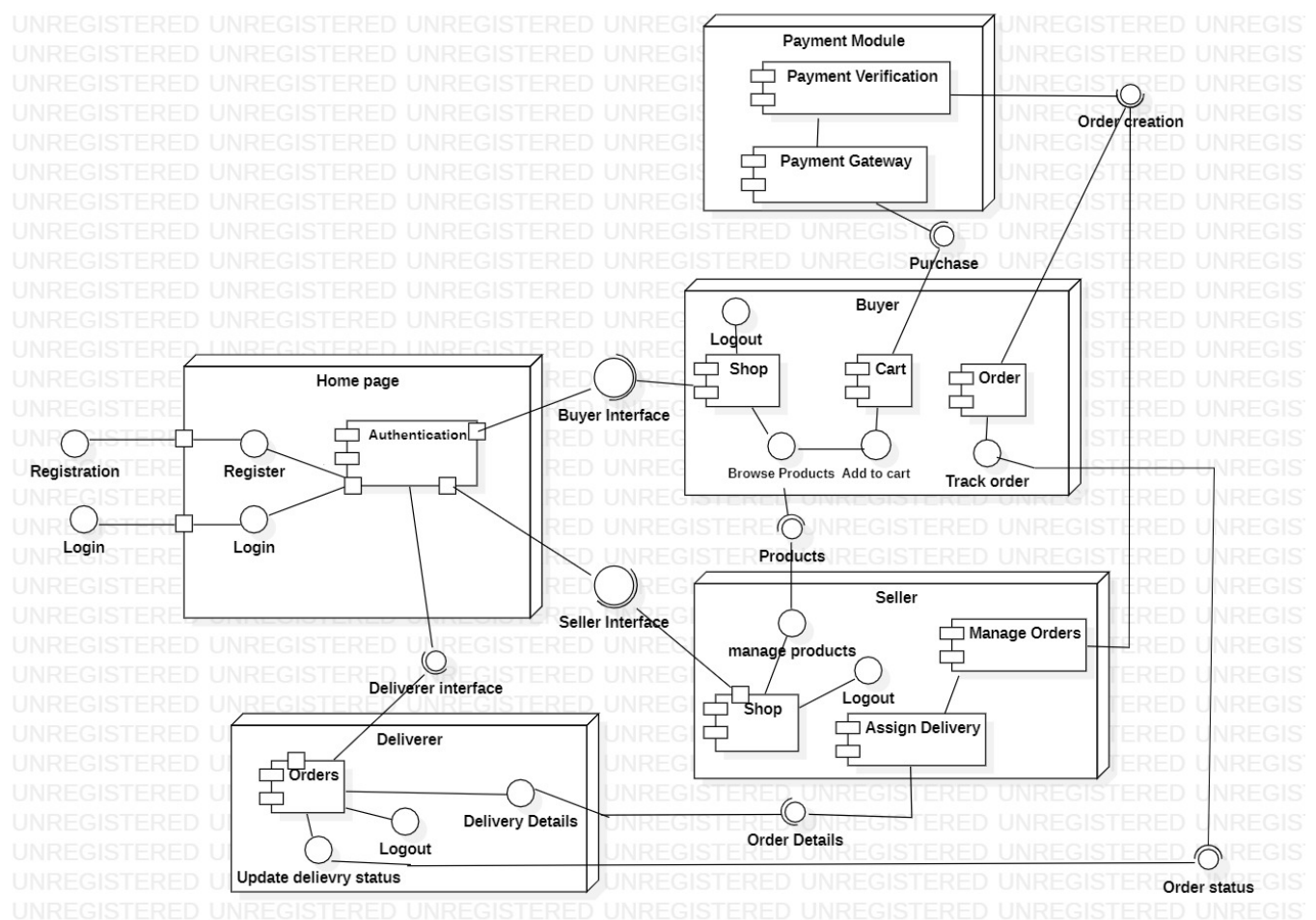


Performance Instruction:

To Draw a Component Diagram

- Take stock of everything needed to implement the planned system. For example, for a simple e-commerce system, you'll need components that describe products, orders, and customer accounts.
- Create a visual for each of the components.
- Describe the organization and relationships between components using interfaces, ports, and dependencies.

Output:



Conclusion:

Component diagram was made successfully by following above steps.

Viva Questions

1. Explain term component diagram?

Ans.

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.

2. Component diagram explains which view of system?

Ans.

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

3. Explain steps to draw component diagram?

Ans.

- Take stock of everything needed to implement the planned system. For example, for a simple e-commerce system, you'll need components that describe products, orders, and customer accounts.
- Create a visual for each of the components.
- Describe the organization and relationships between components using interfaces, ports, and dependencies.

4. What is benefit of drawing component diagram?

Ans.

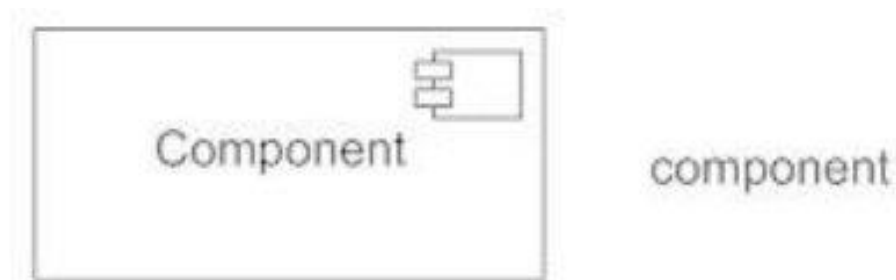
Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities.

5. Explain symbols used to draw component diagram?

Ans.

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.

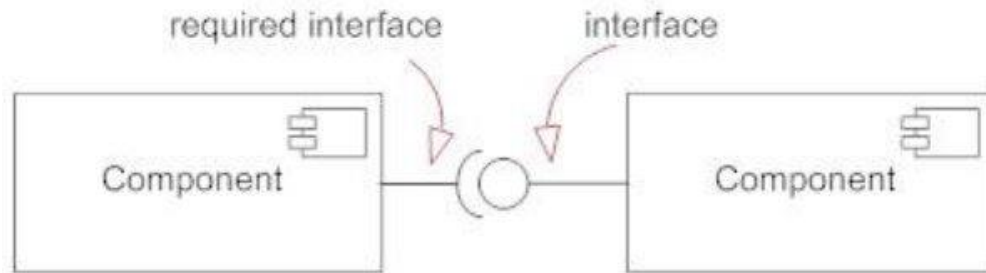


Interface

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.

**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.

