Génie logiciel pour la conception d'un Système d'Information

CSC4521

Voie d'Approfondissement
Intégration et Déploiement de Systèmes d'Information
(VAP DSI)

UML diagrams for System Architecture

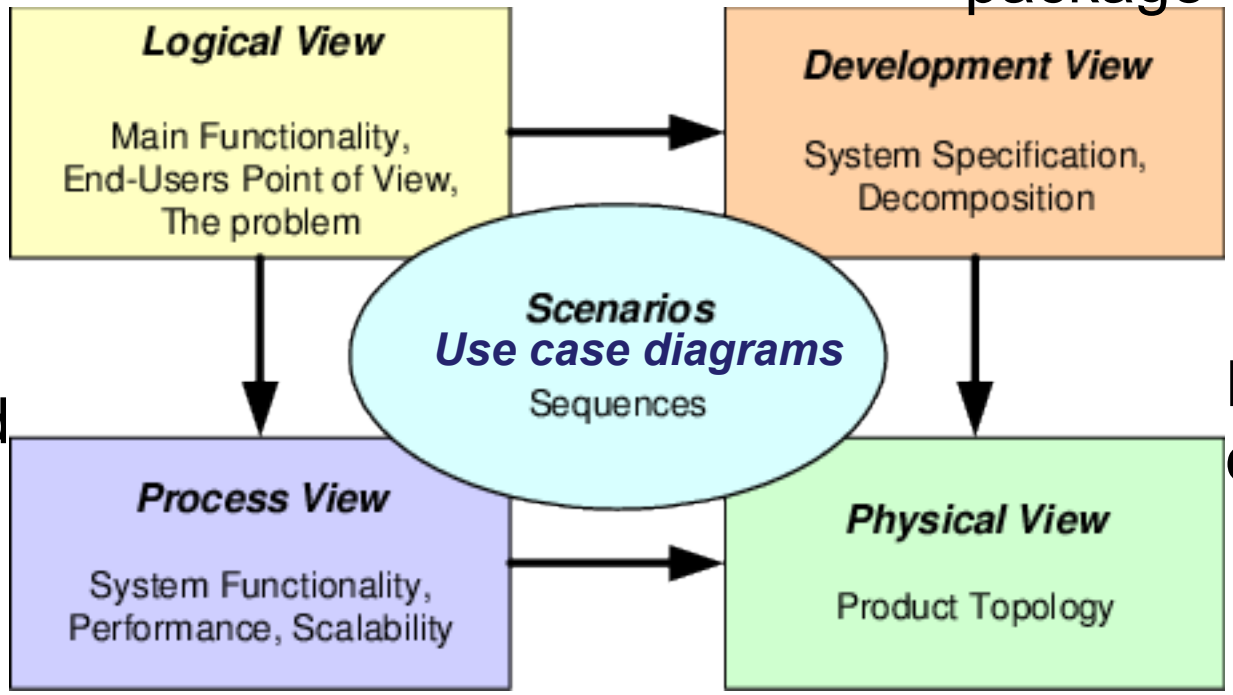
http://jpaulgibson.synology.me/~jpaulgibson/TSP/Teaching/CSC4521/

…/CSC4521/CSC4521-SystemArchitecture

The 4+1 architectural view model

Class, Sequence and State diagrams

Component and package diagrams



Activity and interaction overview diagrams

Deployment diagrams

The “4+1” view model is rather “generic”: other notations and tools can be used

Kruchten, Philippe (1995, November). Architectural Blueprints — The “4+1” View Model of Software Architecture. IEEE Software 12 (6), pp. 42-50.

UML diagrams for system architecture

- **Component Diagrams**
- **Activity Diagrams**
- **Interaction Overview Diagrams**
- **Deployment Diagrams**
- **Package Diagrams**

You have already seen class, sequence and state machine diagrams?

The component diagram

In UML 2, components are considered autonomous, encapsulated units within a system or subsystem that provide one or more interfaces.

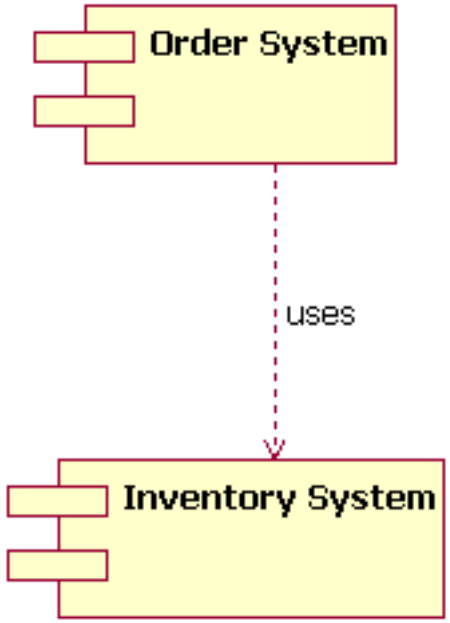
Components are larger design units that represent things that will typically be implemented using “replaceable modules”.

A component encapsulates behaviour and implements specified interfaces

Developers find the component diagram useful because it provides them with a high-level, architectural view of the system

A component contains “related” classes, used to provide a single service. Many of the classes inside are hidden and can be changed, so long as the service is still provided in the same way.

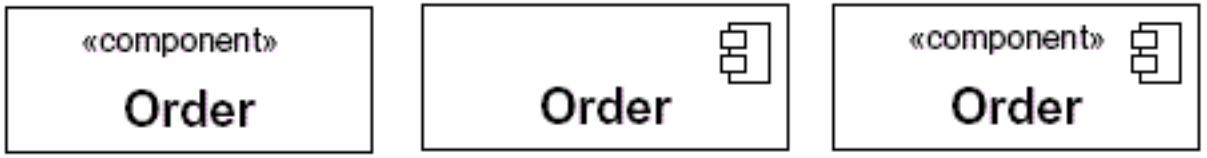
The component diagram

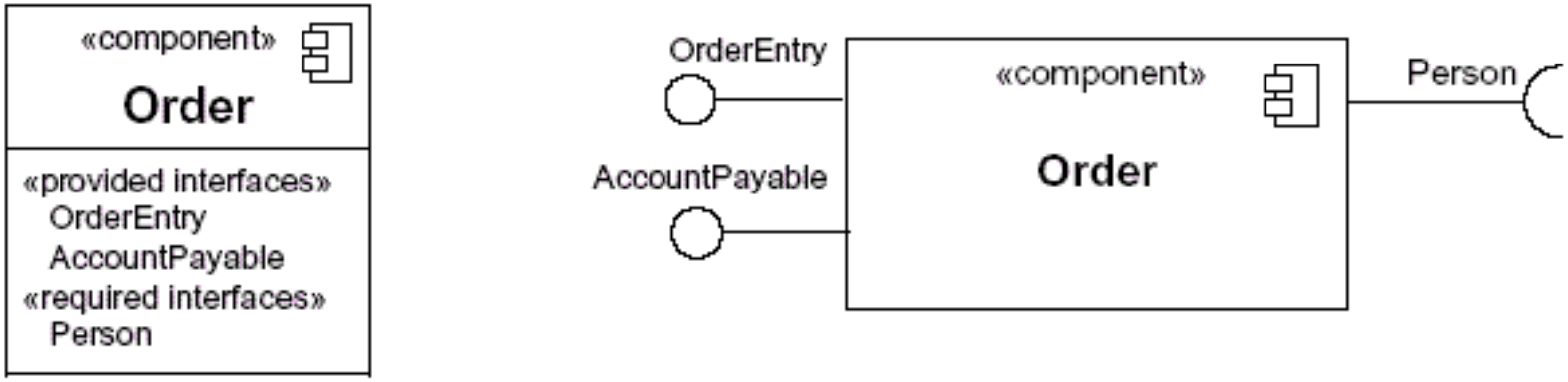


UML 1.4

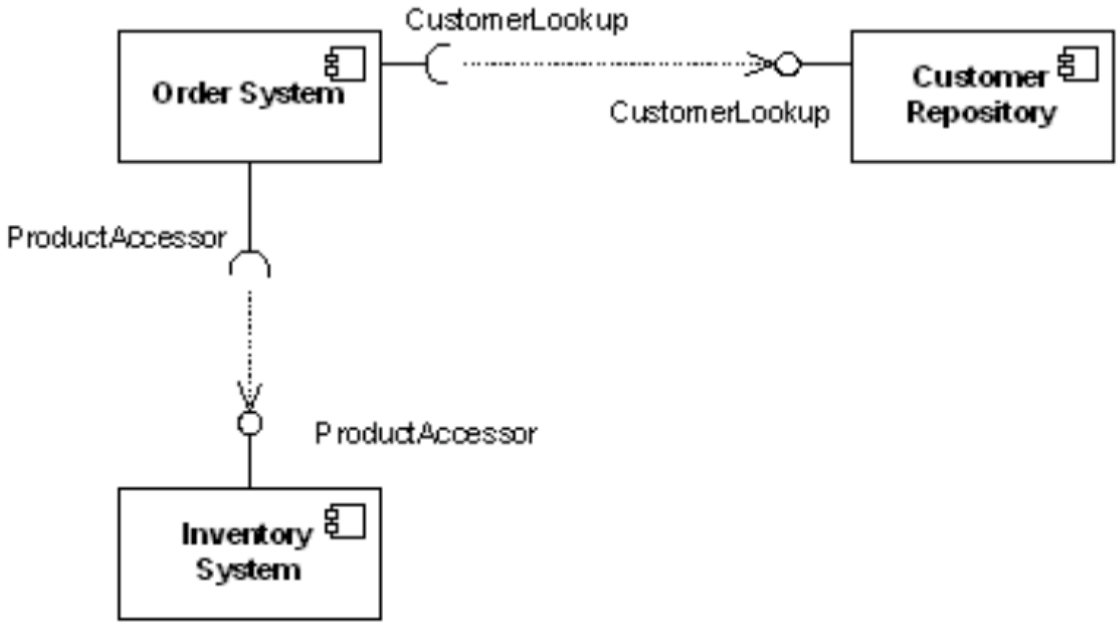
The component diagram

In UML 2 a component is merely a specialized version of the class concept.



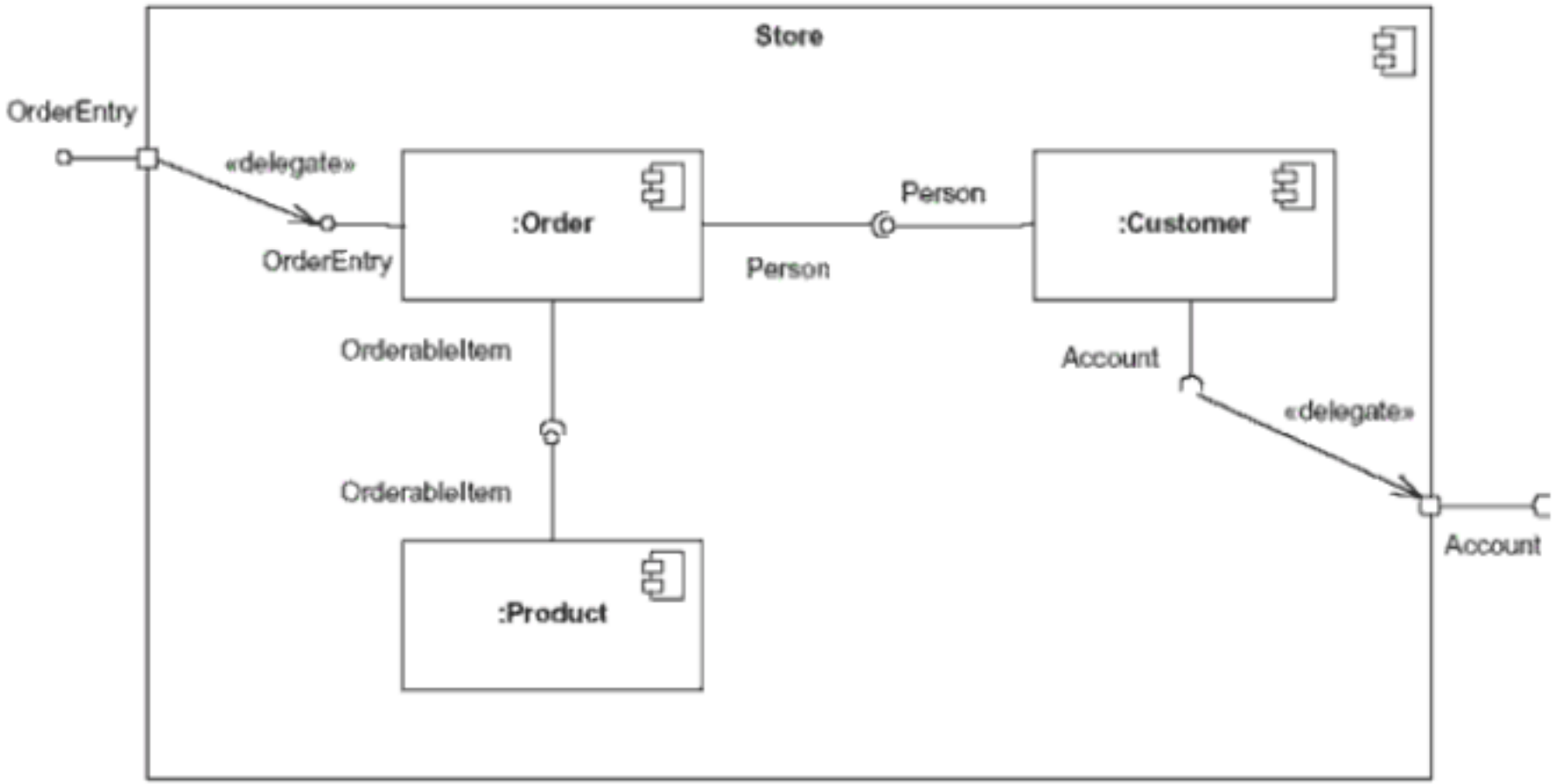


The component diagram



When showing a component's relationship with other components, the lollipop and socket notation must also include a dependency arrow (as with a class diagram).

The component diagram



Structured Components

The activity diagram

Activity diagrams are suitable for modelling the activity flow of the system.

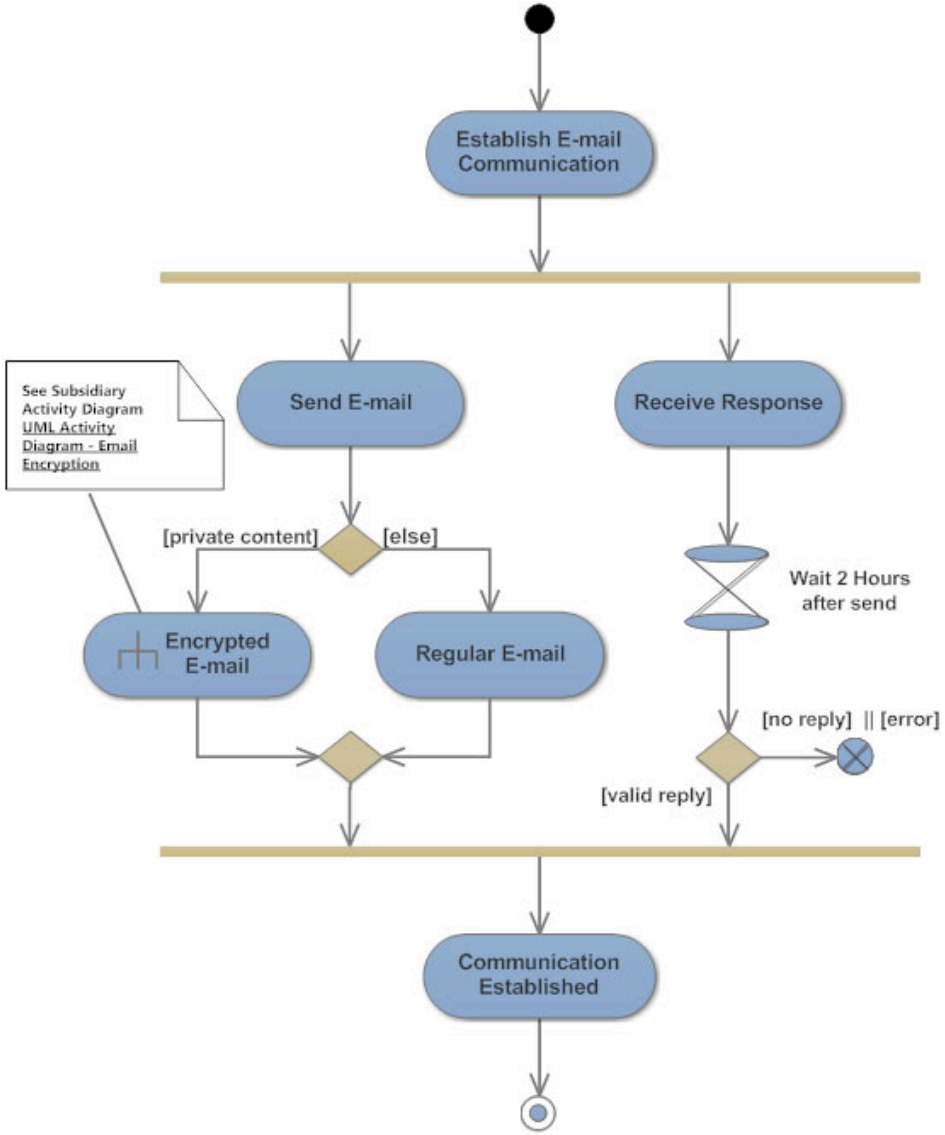
It captures the dynamic behaviour of the system

An application can have multiple systems. Activity diagram also captures these systems and describes the flow from one system to another.

This specific usage is not available in other UML diagrams.

Often used to describe steps in a use case

The activity diagram

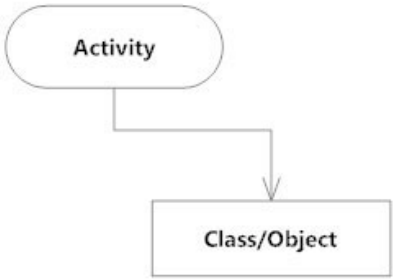


The activity diagram

Start Point/Initial State

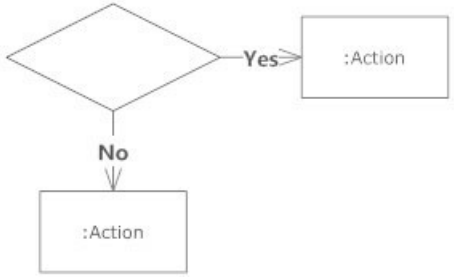
Activity

Action Flow



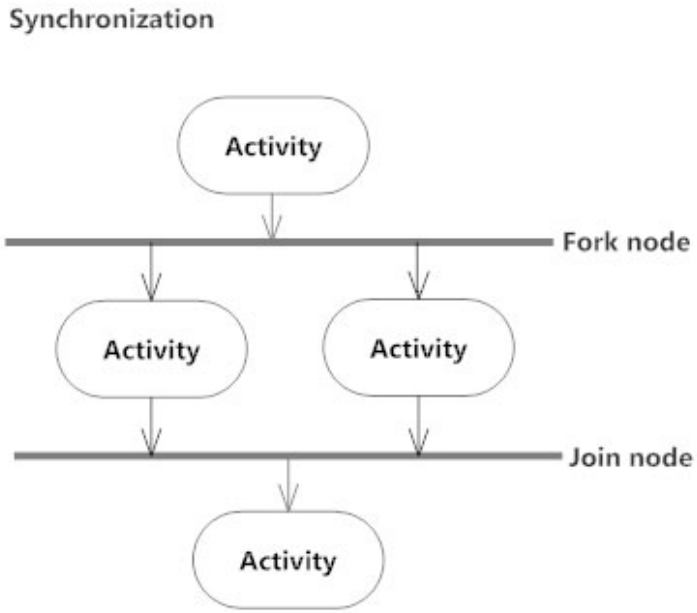
Object Flow

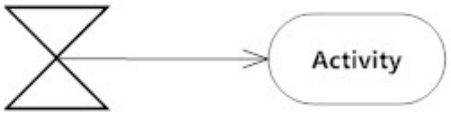
Decision Symbol

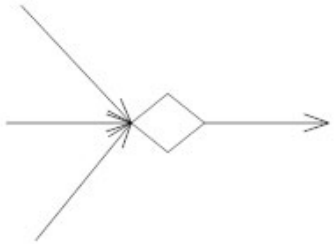


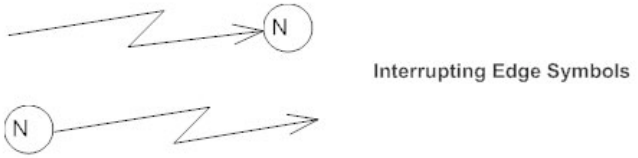
Guard Symbols

The activity diagram



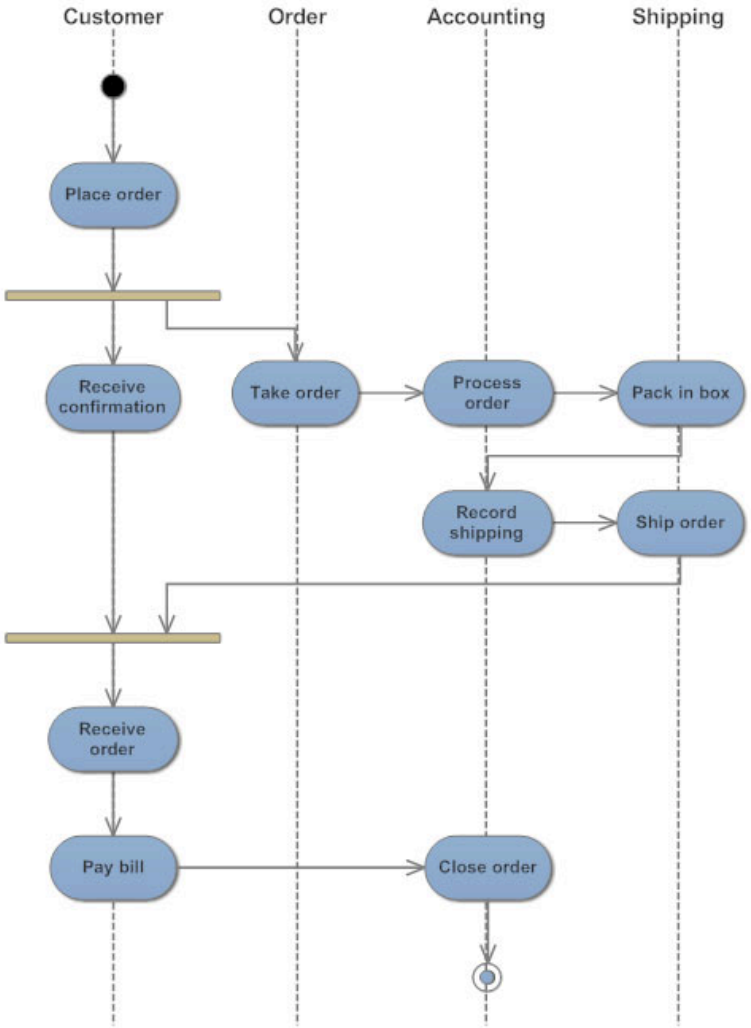






Signal sent and received

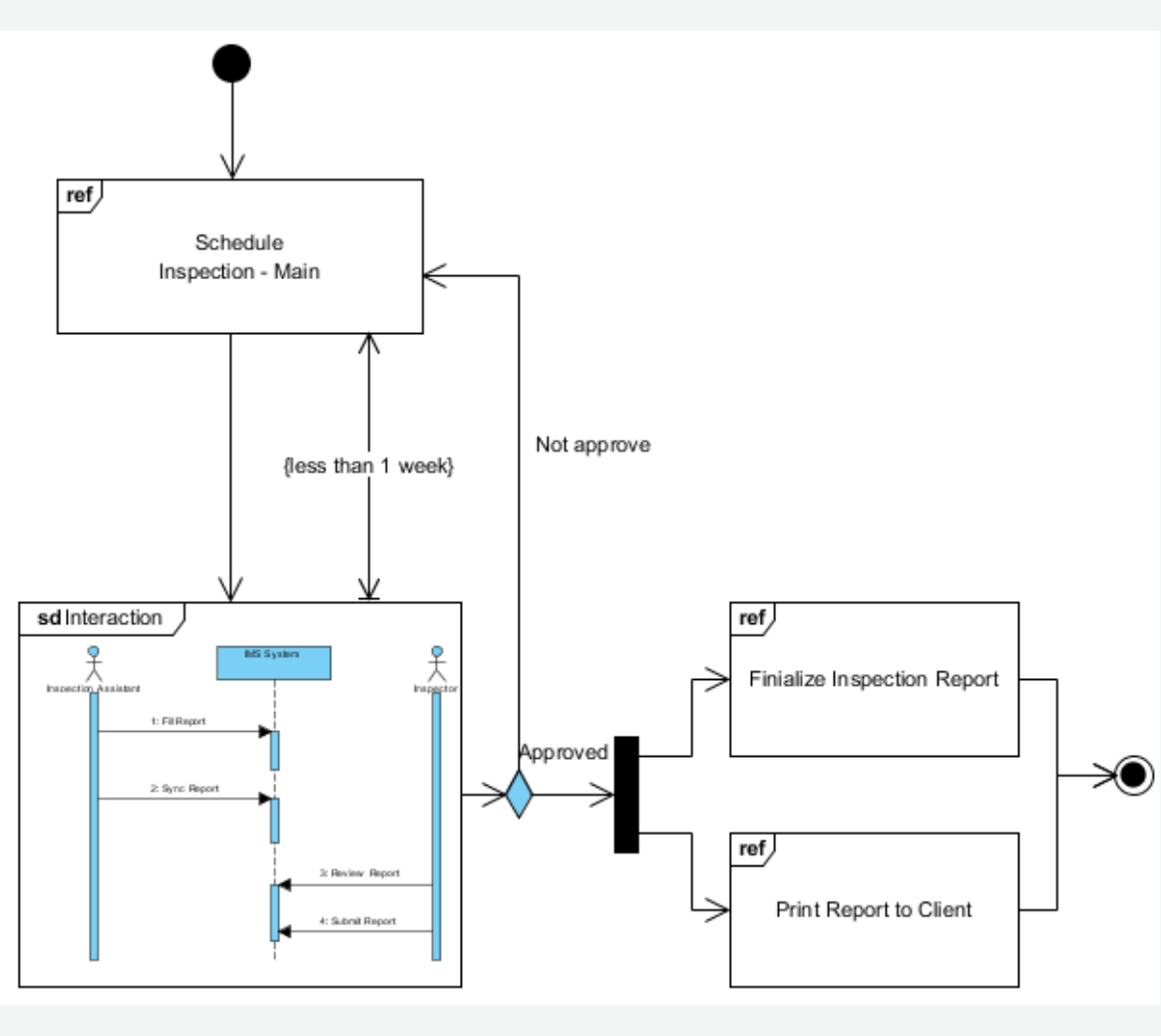
The activity diagram



Swim lanes group related activities into columns

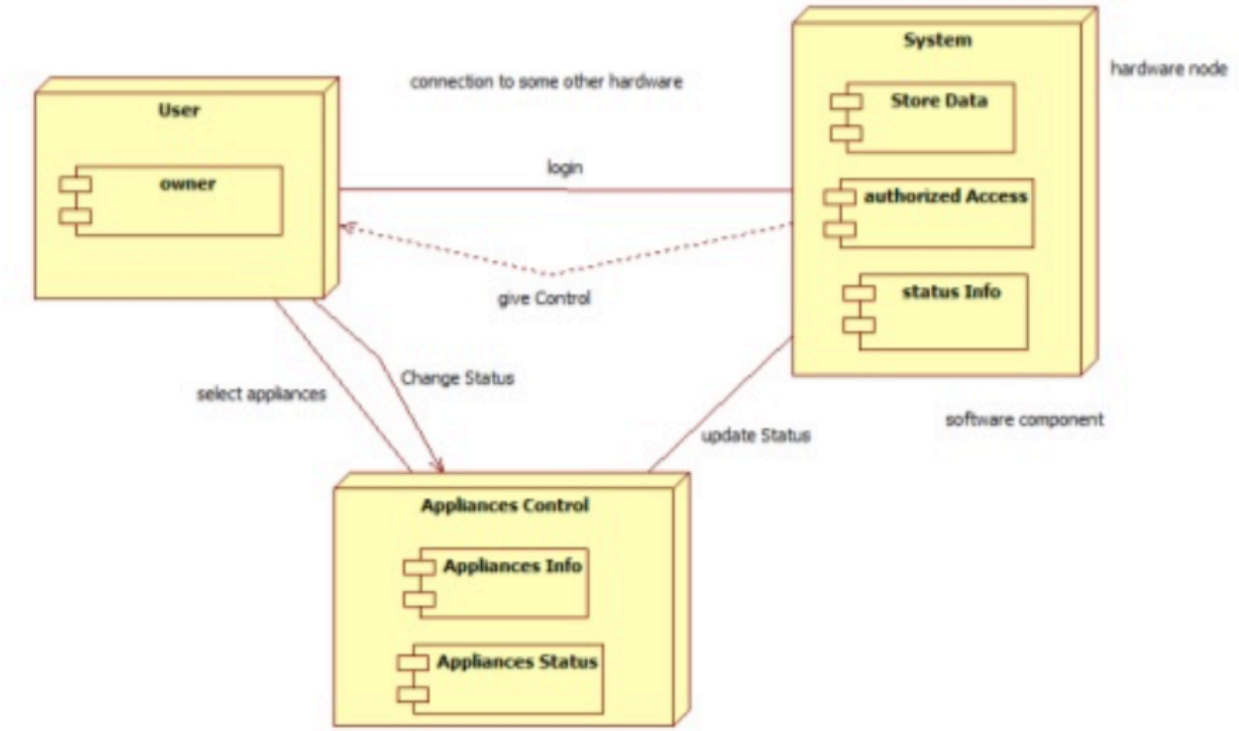
The interaction overview diagram

Interaction overview diagram is an activity diagram made of different interaction diagrams



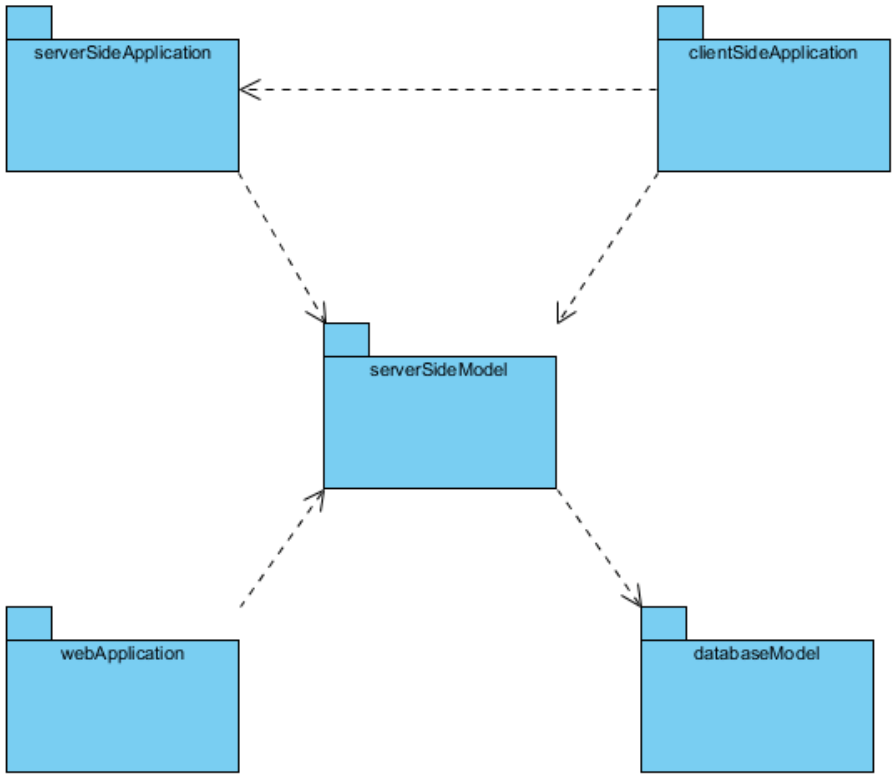
The deployment diagram

A **UML 2 deployment diagram** depicts a static view of the run-time configuration of processing nodes and the components that run on those nodes … suitable for modelling (physically) distributed aspects of a system.



The package diagram

Package diagrams are used to structure high level system/model elements.



Packages are depicted as file folders and can be used on any of the UML diagrams, although they are most common on use-case diagrams and class diagrams because these models have a tendency to grow.