

Assignment #1:

Functional Requirements:

The requirements that specifies what the system should do are known as functional requirements. It should include functions performed by specific screens, outlines of work-flows performed by the system, and other business or compliance requirements the system should provide.

It should include:

- o Description of data to be entered into the system.
- o Description of operations performed by each screen.
- o Description of work-flows performed by the system.
- o Description of system reports or other output.
- o Who can enter the data into the system

Non-functional requirements:

It is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. It defines the constraints on how the system will do so. They are often called "quality attributes" of a system.

Non-functional requirements are applied to the whole system rather than each function. Examples of non-functional requirements are usability, reliability, performance, supportability, security, etc.

In a real-estate system for a realtor, the functional and non-functional requirements are:

- Functional requirements:

- Login and registration facility
- Addition of new brokers.
- Addition of new property by admin
- Property search.
- Enquiry form.

- Non-functional requirements:

- Reliability
- Security of the system
- Service availability.
- User supportability.

Domain modeling:

A domain model contains conceptual class, association between conceptual classes, and attributes of a conceptual class.

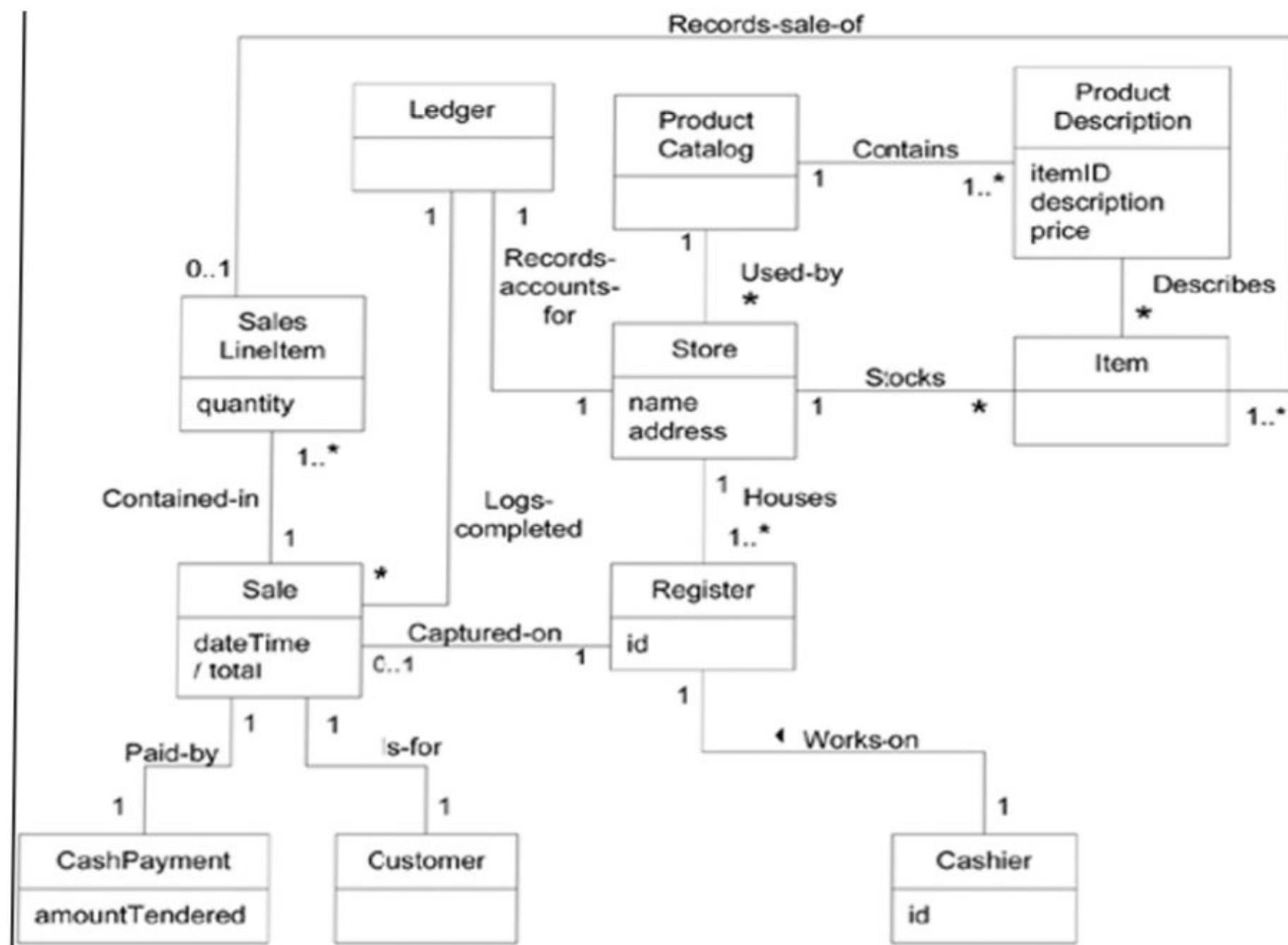
It is illustrated with a set of class diagram without the operations.

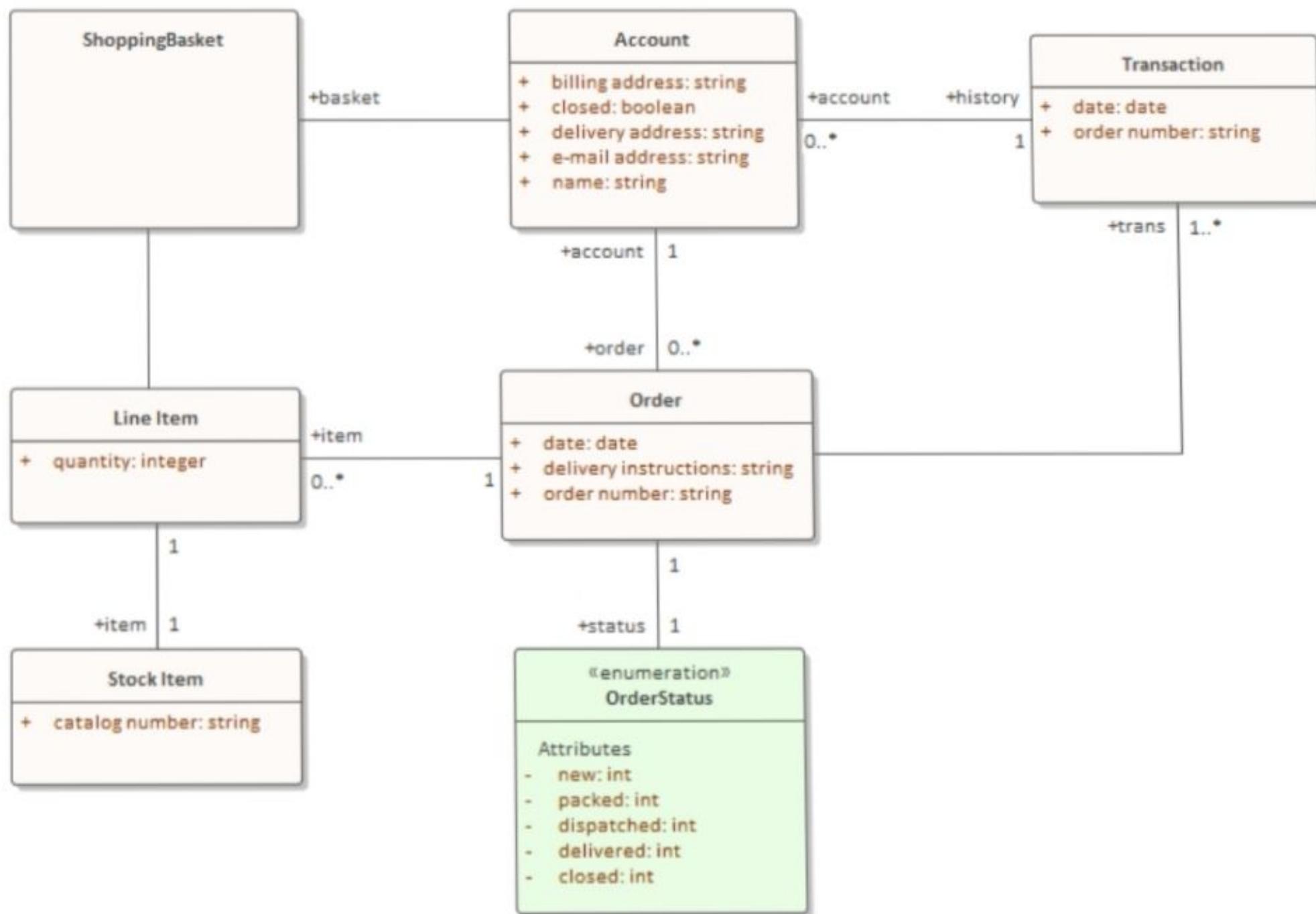
Guidelines for Domain Modeling:

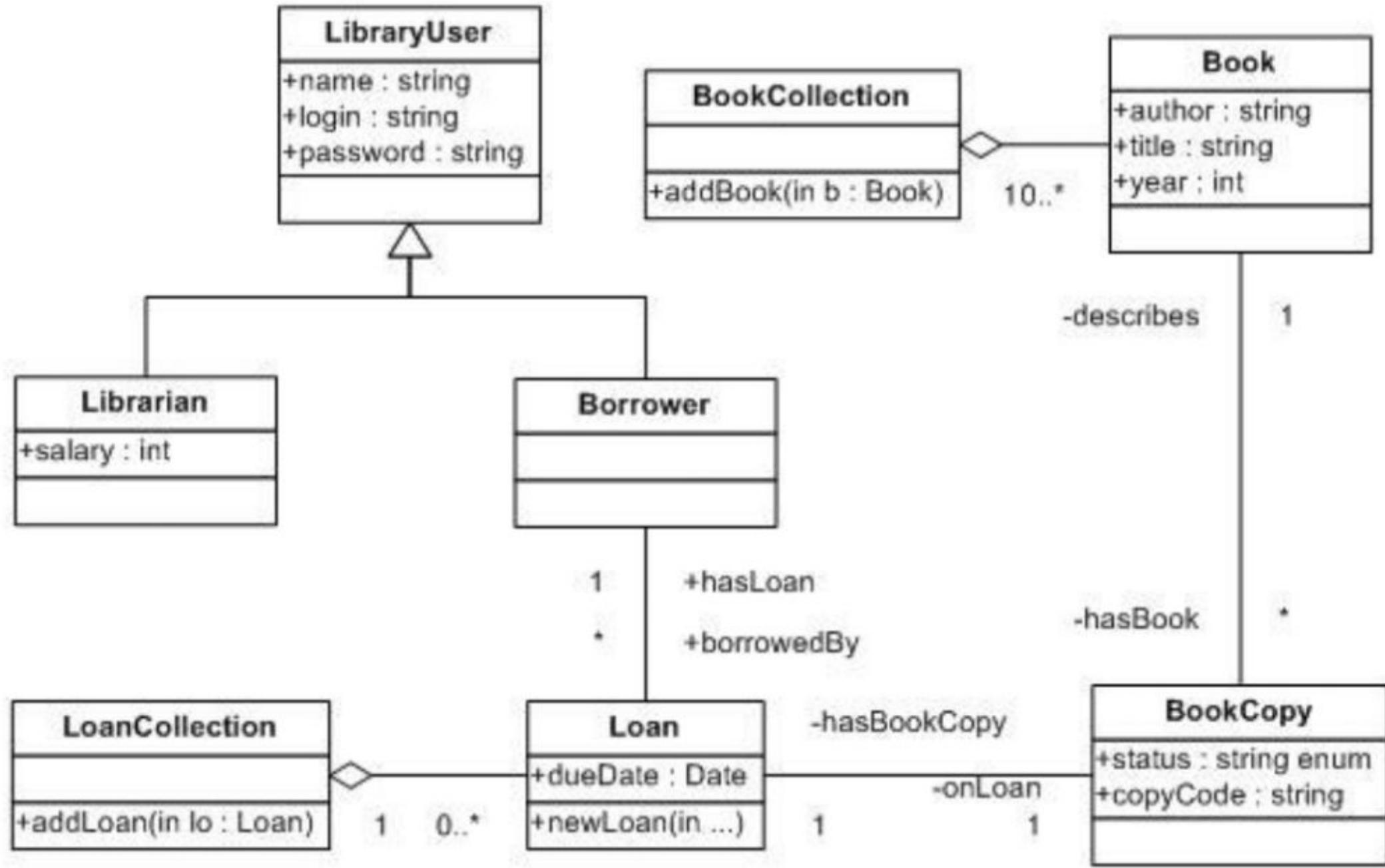
- o List the candidate conceptual classes by using any of the 3 process:- By reusing or modifying the existing model if one exists
 - Using a category list
 - Identifying noun phrases in use-cases.
- o Draw them in domain model using UML.
- o Add the associations necessary to record relationships.
- o Add the attributes necessary to fulfill the information.

Example:

Domain model for POS after adding attributes





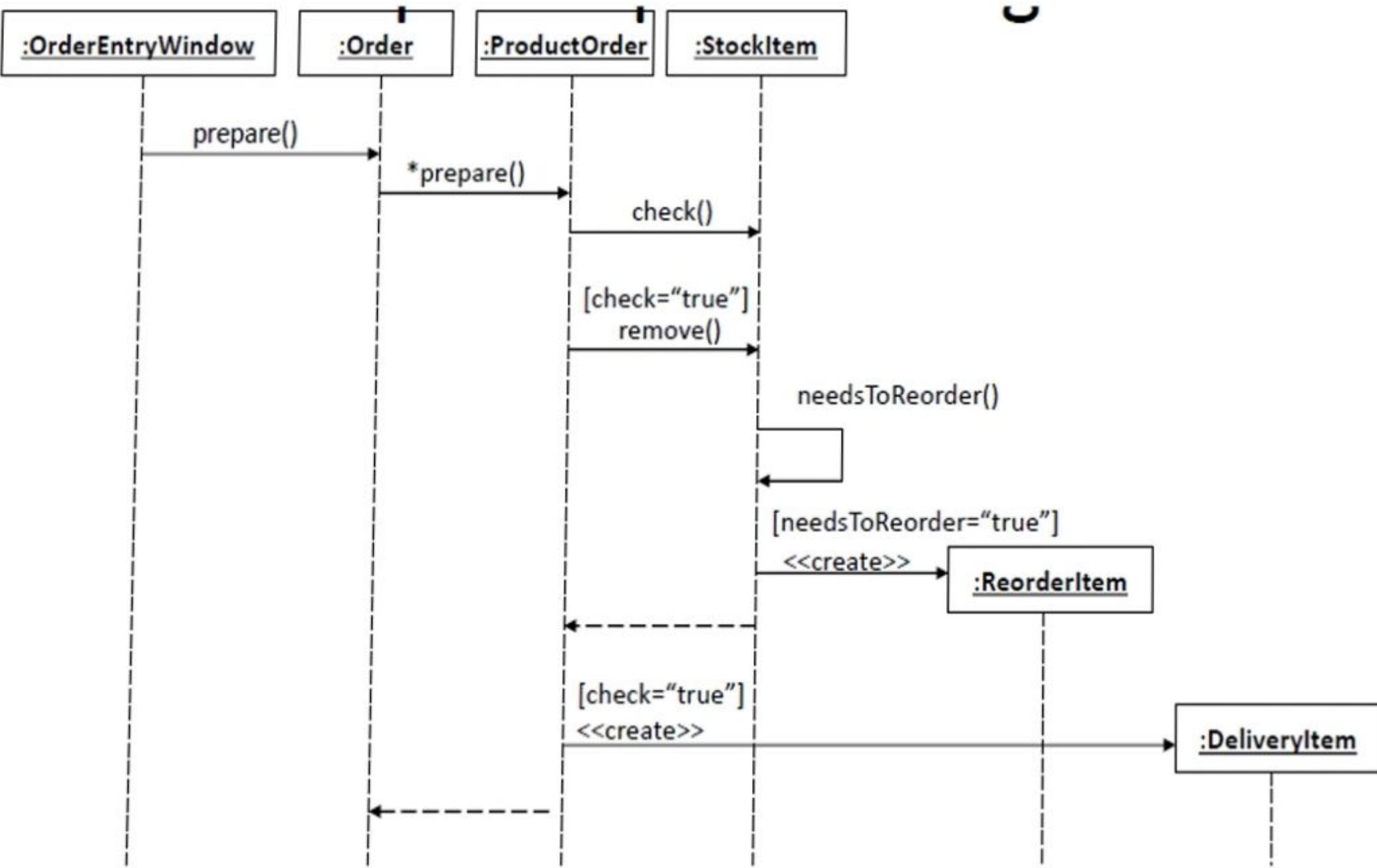


Sequence Diagram

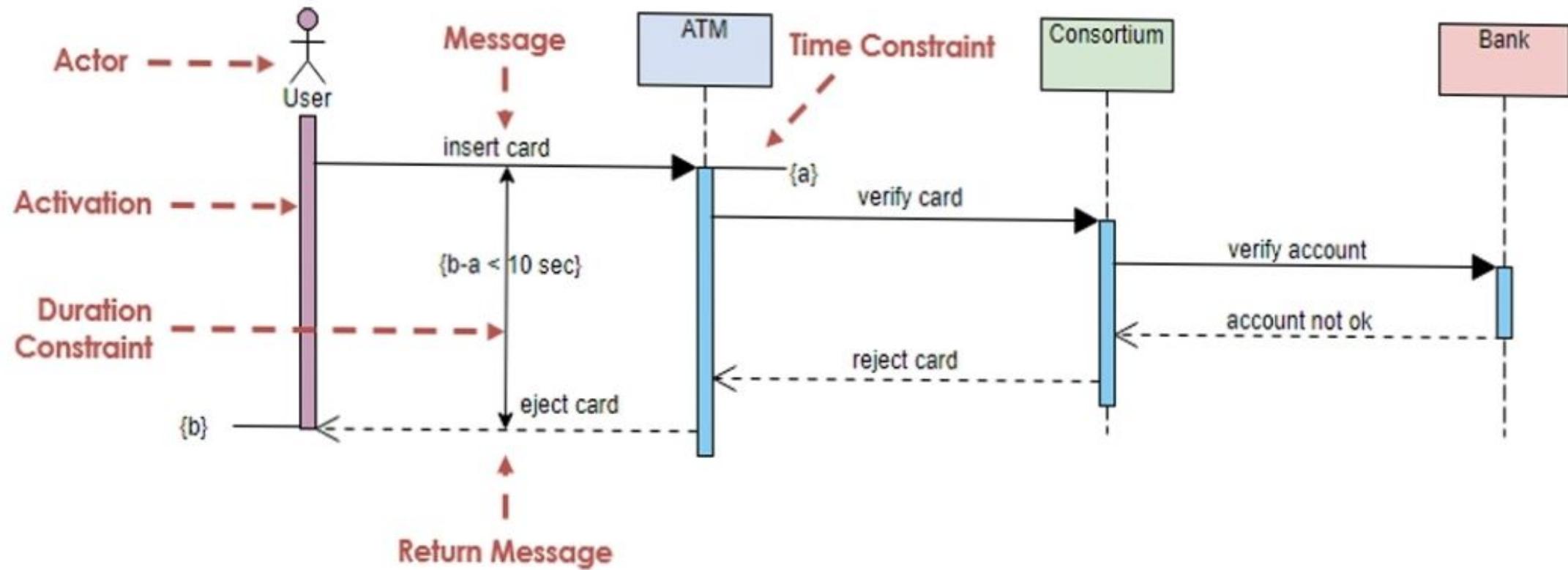
- It is an interaction diagram that shows the set of objects and messages sent and received between those objects.
- It emphasizes on time ordering of messages between objects.

Notations: Example:

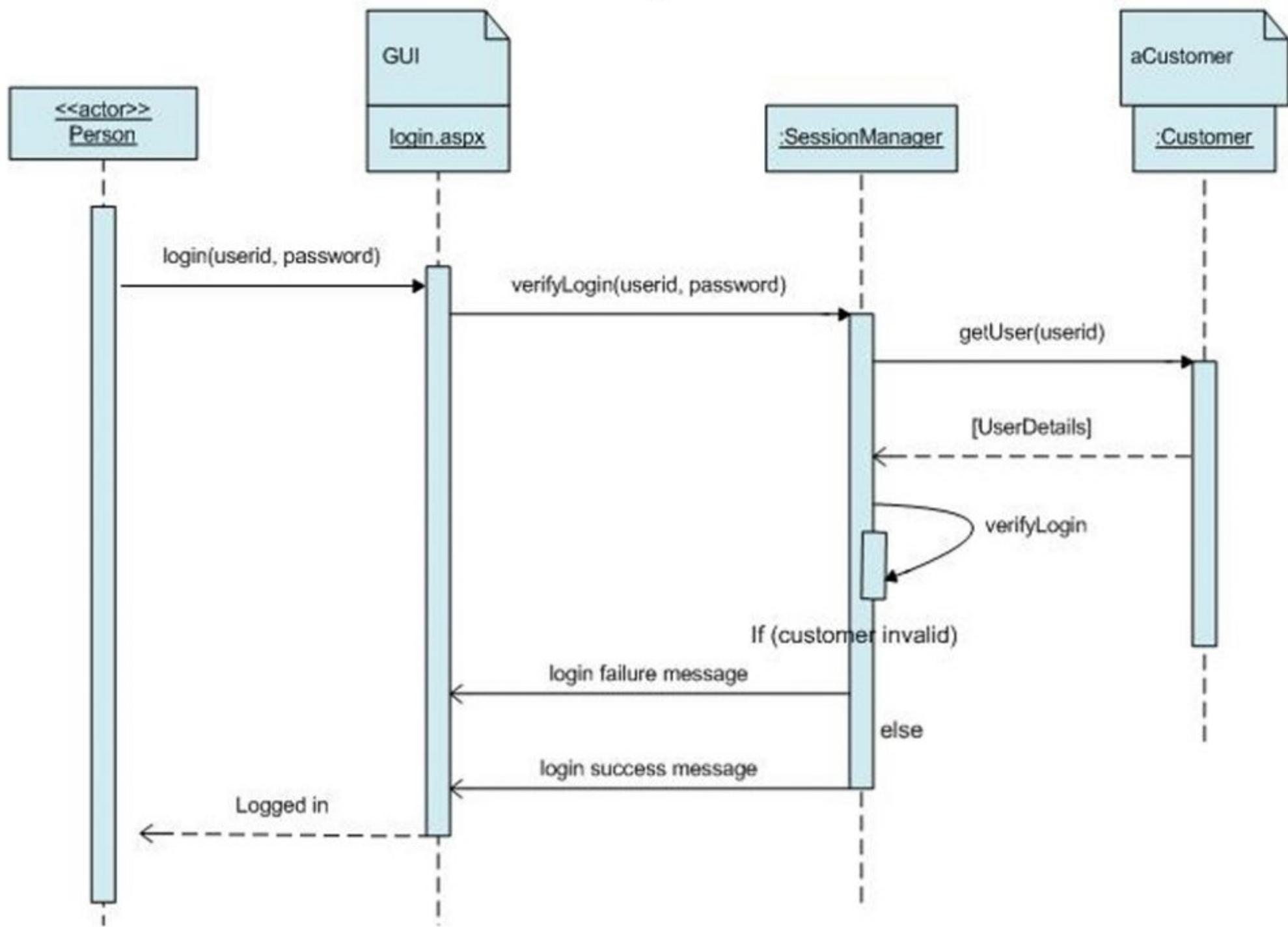
Order processing



ATM (Invalid Pin)



Login



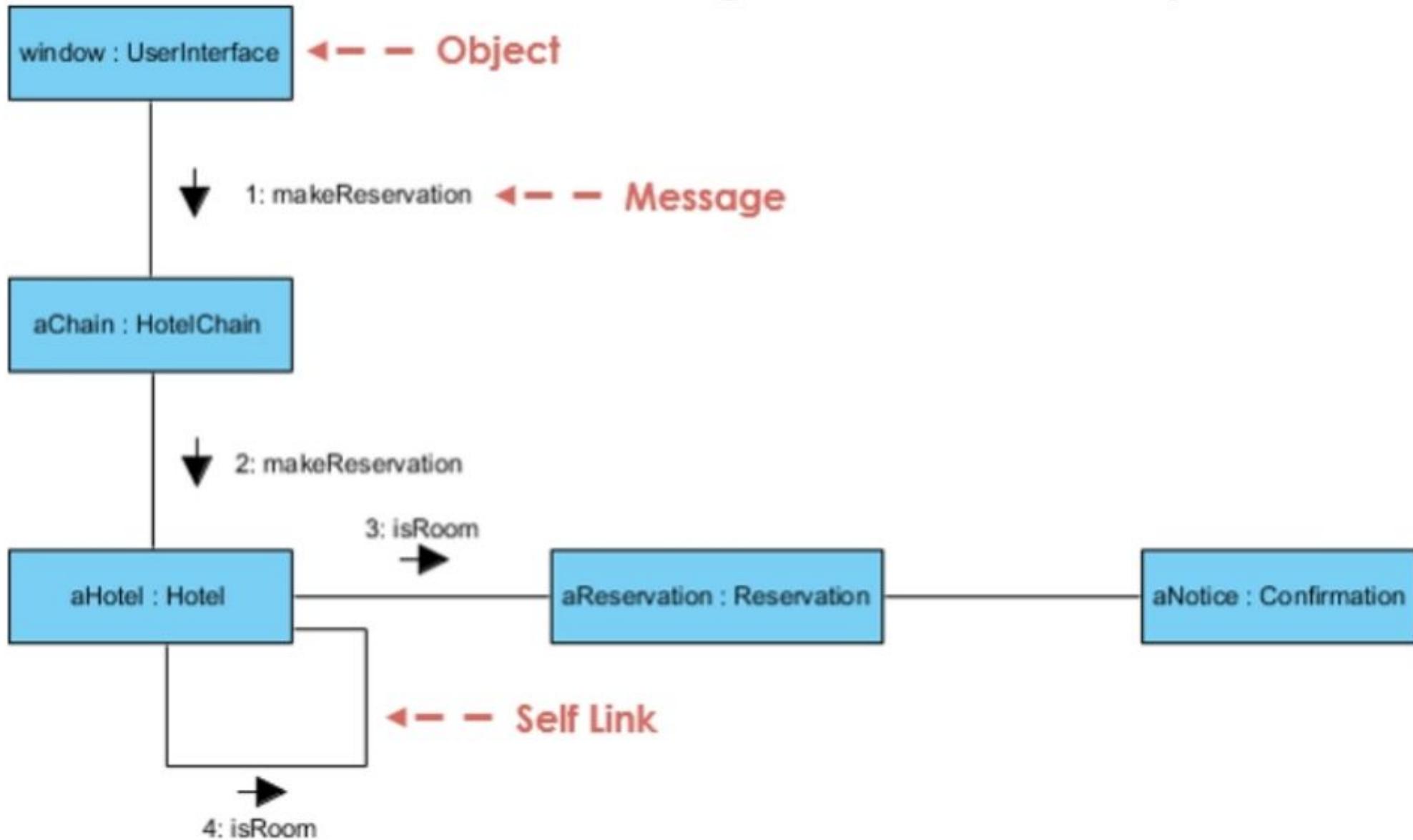
Communication diagram: (Collaboration diagram)

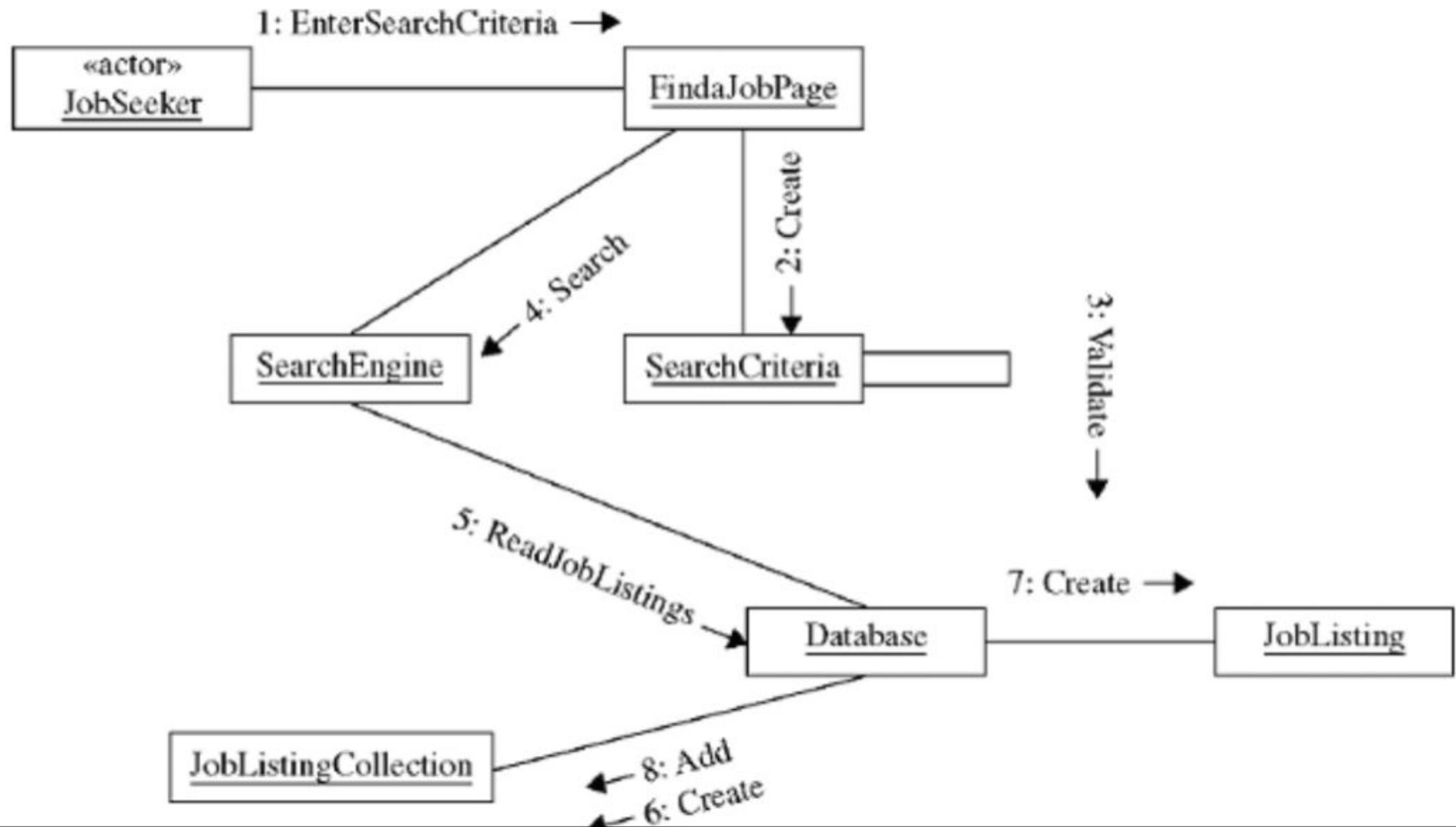
They represent a particular sequence of messages exchanged between a number of objects.

Along with sequence diagrams, they are used to by designers to define and clarify a particular flow of events of a use case.

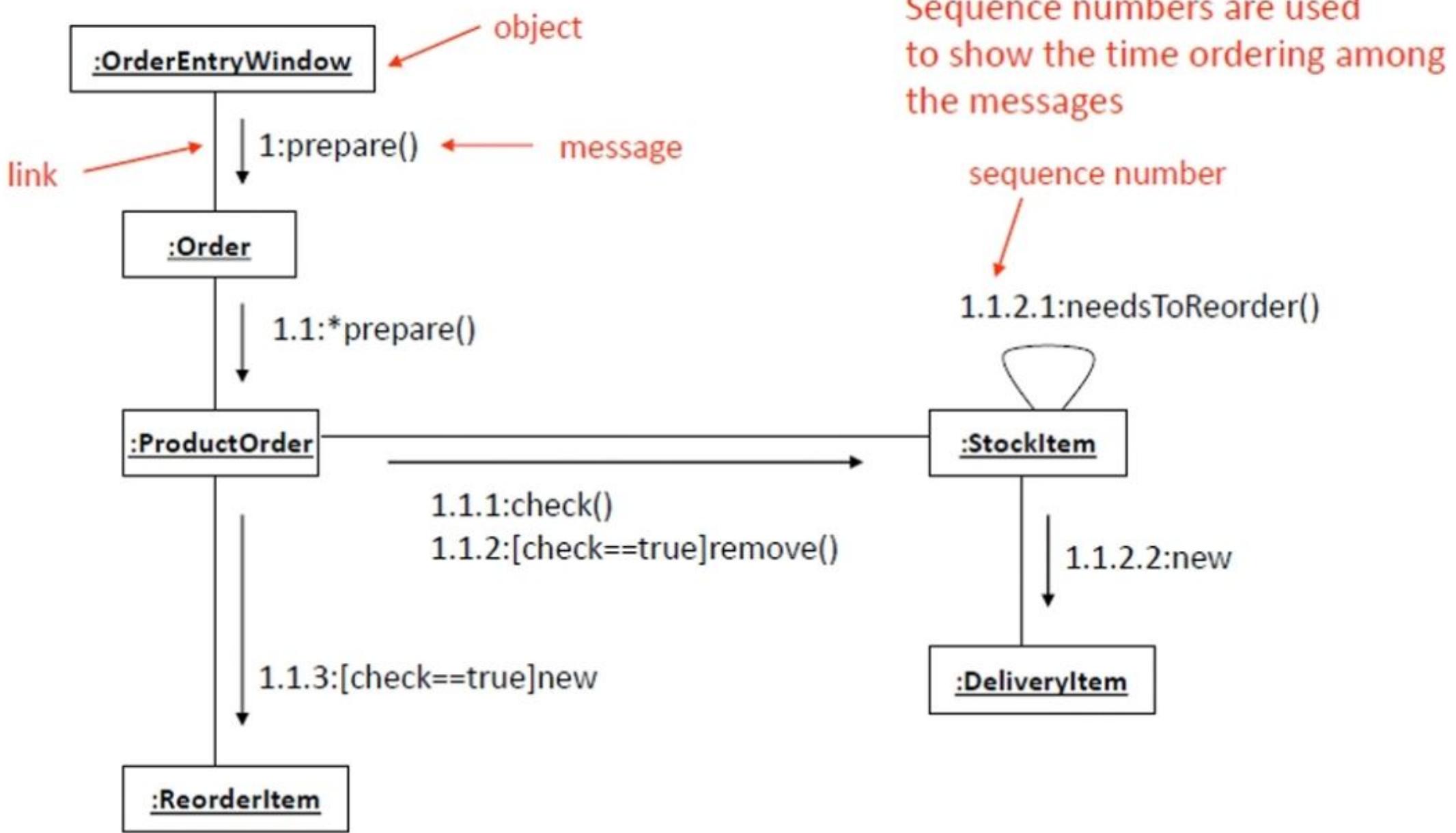
Example:

Collaboration Diagram Example





Collaboration (Communication) Diagrams

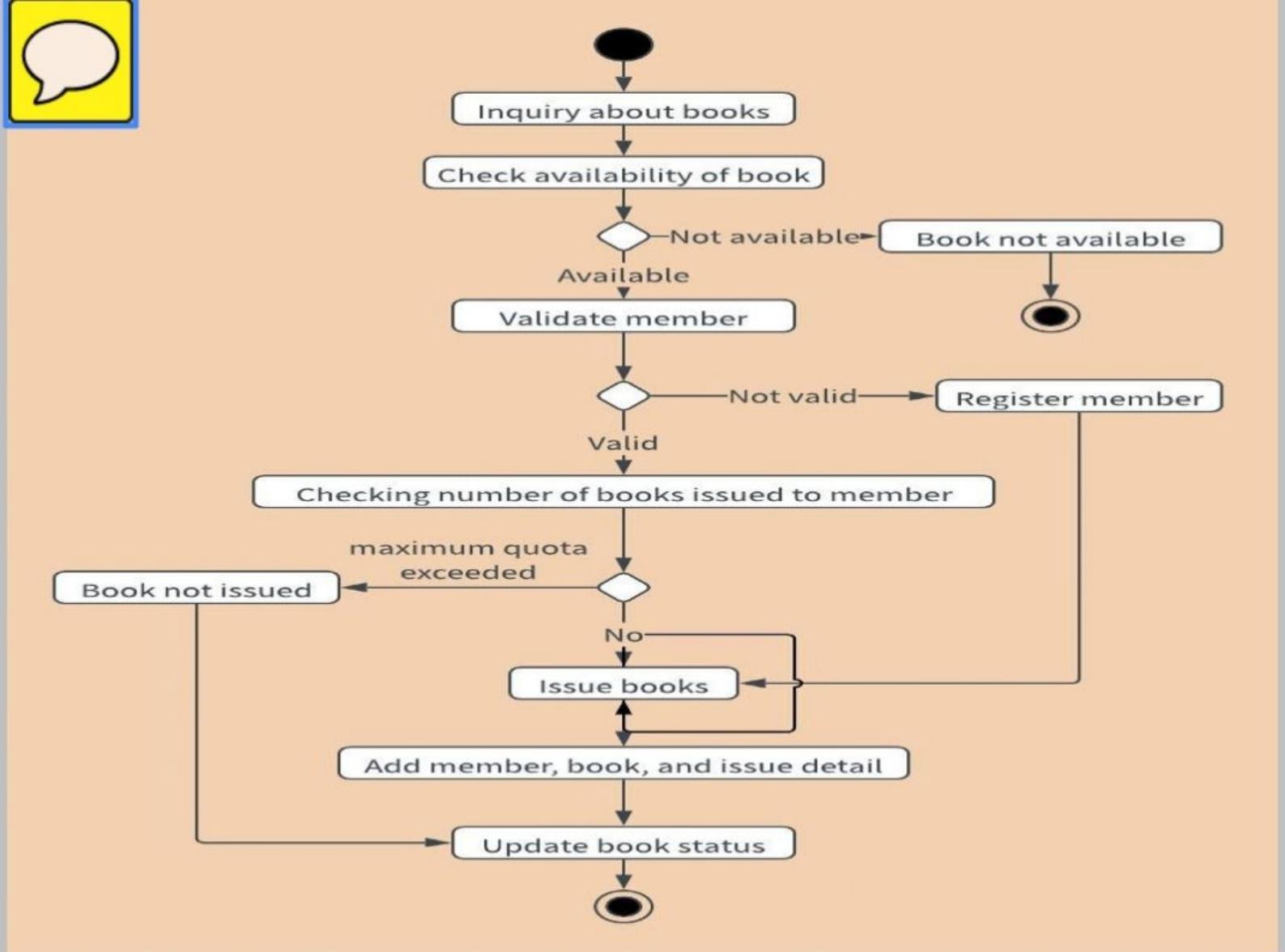


Activity diagram:

It visually represents a series of actions or flow of control in a system similar to a flowchart. Activities modeled can be sequential and concurrent. Every activity diagram must have.

- Initial state
- Final state.
- Series of actions performed by the system between two above states.

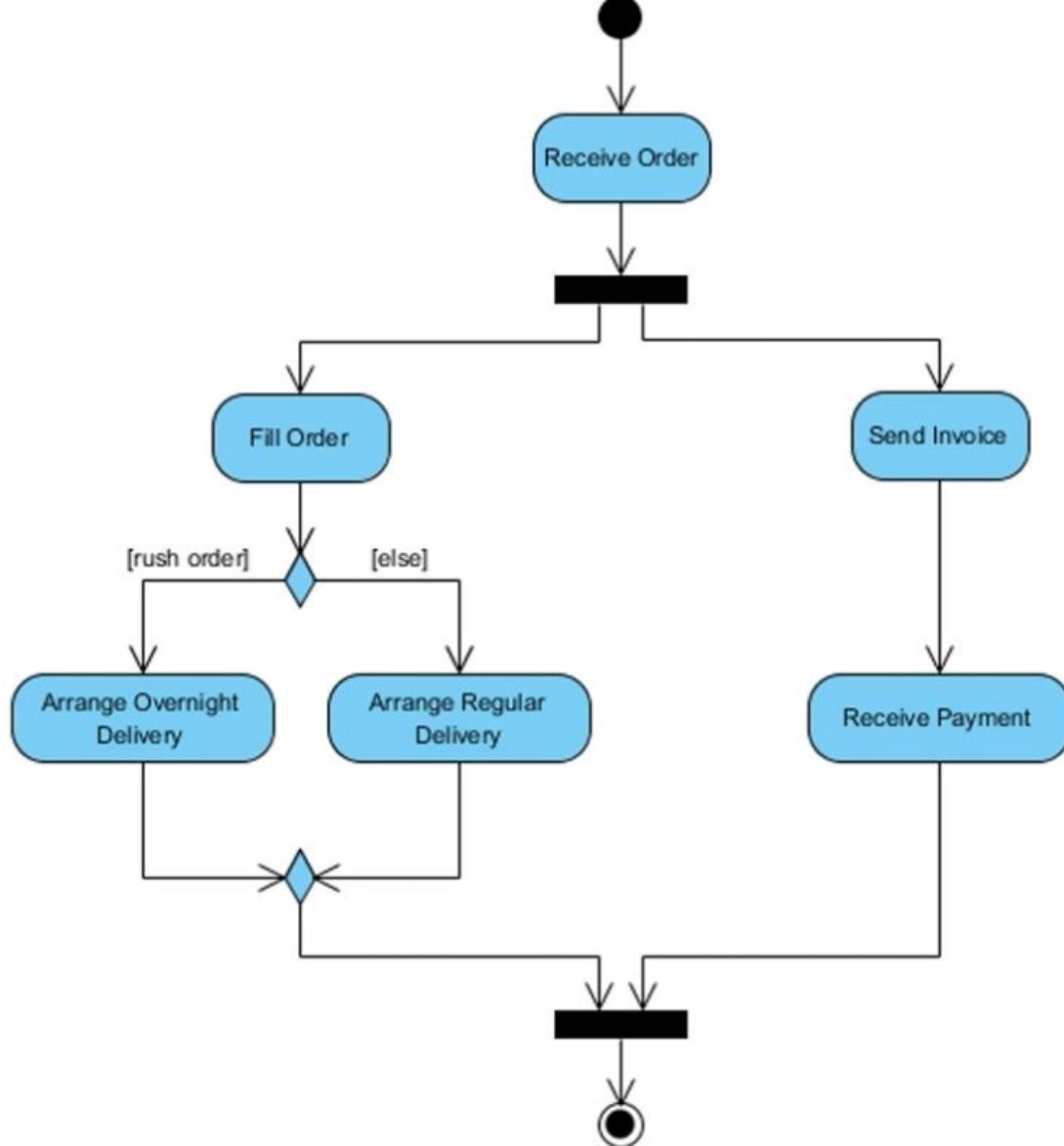
Example:



Presenter 11 Jun 2019



Library Management system



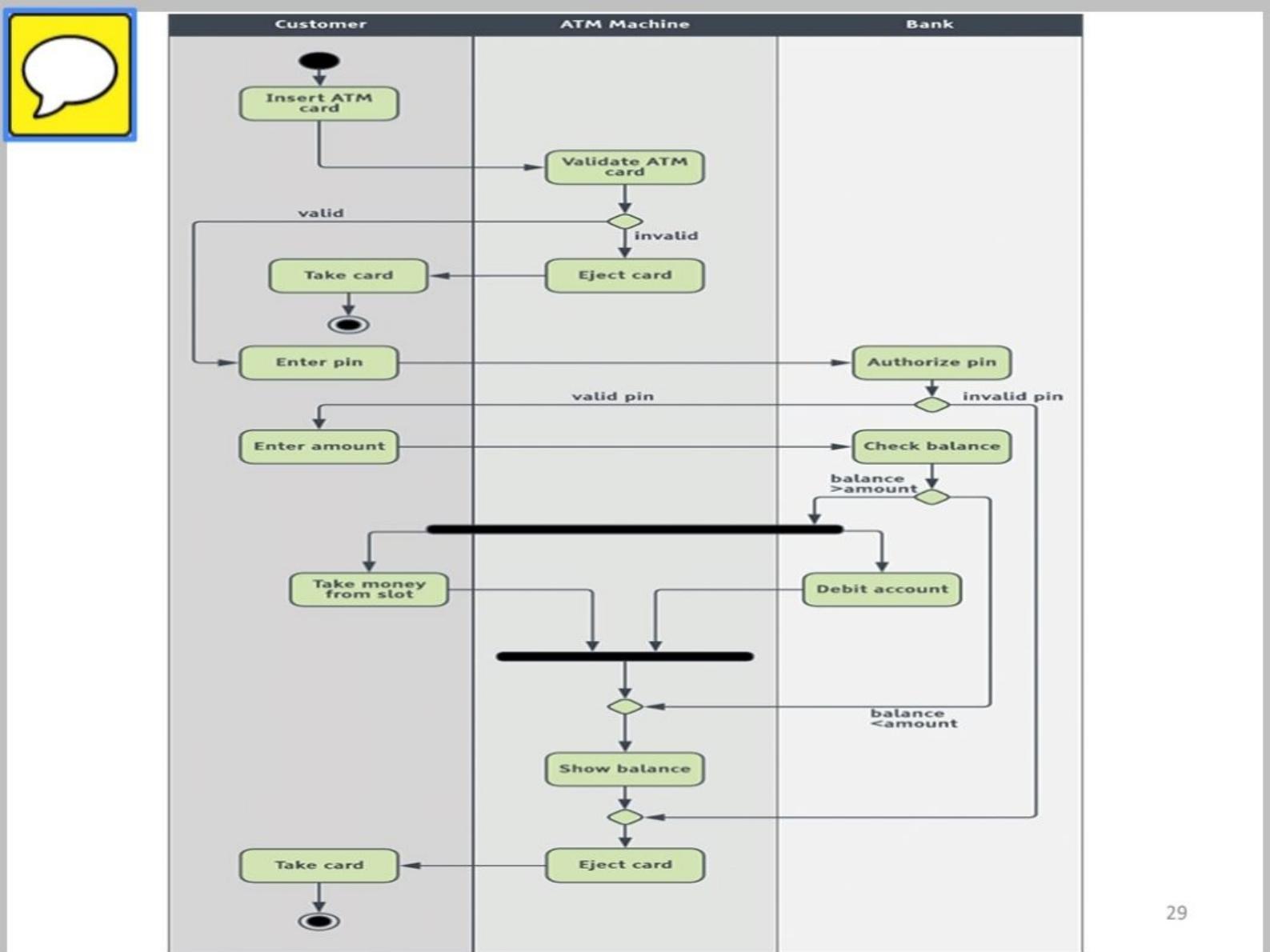
31



Presenter 11 Jun 2019



Activity diagram of order processing



29



Presenter 11 Jun 2019



ATM example with swimlane