Information Systems Analysis & Modeling

Lecture 6: User story and Use case

Mona Taghavi

LaSalle College
Montréal

- Identifying user stories and use cases is a key task when defining functional requirements.

What is a user story?

- A one sentence description, written from the viewpoint of the user, describing what function is needed.
- A **user story** is in the everyday language of the end user that states what a user does as part of his or her work.
- User stories are a basic concept in Agile development because they focus on simplicity, value added, and user collaboration.

Example of a user story

As a customer, I want to be able to purchase a book with a credit card, so it is convenient.

As a sales manager, I should be able to run reports at anytime, so I can make adjustments to improve our sales numbers.

User story structure

As an <actor>,
I want to be able to <function>,
so that I can <business reason>

As an <actor>,
I should be able to <function>,
so that I can <business reason>

User story structure

As an <actor>,
I want to be able to <function>,
so that I can <business reason>

As an <actor>,
I should be able to <function>,
so that I can <business reason>

As a customer,
I want to be able to purchase a book with a credit card,
so that I can purchase my books conveniently.

As a sales manager,
I should be able to run reports at anytime,
so that I can make adjustments to improve our sales numbers.

Keys to a good user story

I : Independent - each story should stand alone

N : Negotiable - keep the stories nimble until just before implementation

V : Valuable - define features and function that provide value to the business

E : Estimatable – clear enough to enable a validate estimate

S : Small – small enough to be implemented in a single iteration

T : Testable – independently verifiable

Other User story examples

As a sales rep, I should be able to search for my customers by first and last name, so that I can access their details.

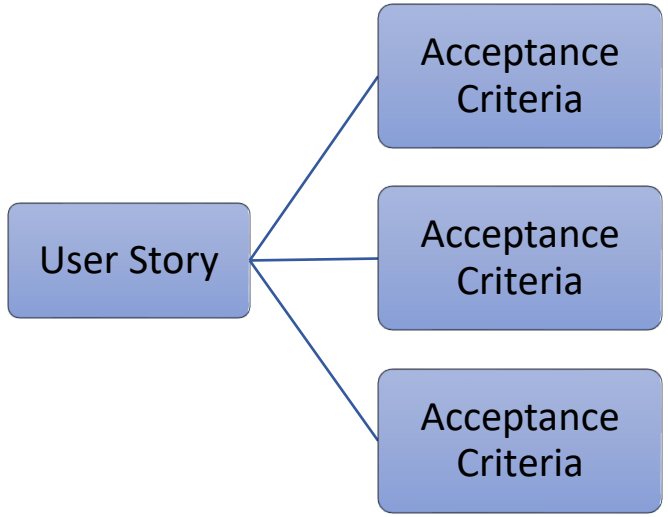
As a truck driver, I want to be able to have turn-by-turn navigation in my cab, so that I can get directions to my destination.

As a customer, I want to be able to reset my password via the internet, so that I can reset it without having call into the customer service line.

As a Human Resources professional, I want to be able to attach digital resumes to candidate profiles, so that I can access those resumes conveniently.

Acceptance Criteria

- Provides details on what the solution needs to accomplish
- Describes how the user story can be verified to be completed



Acceptance Criteria

- Provides details on what the solution needs to accomplish
- Describes how the user story can be verified to be complete

User Story	Acceptance Criteria
As a sales rep, I should be able to search for my customers by first and last name, so that I can pull up their details.	1. Sales rep can search for customers 2. Search criteria can be first name or last name 3. A search can return 0, 1, or many results 4. For one or many results, the rep can choose a customer from the list

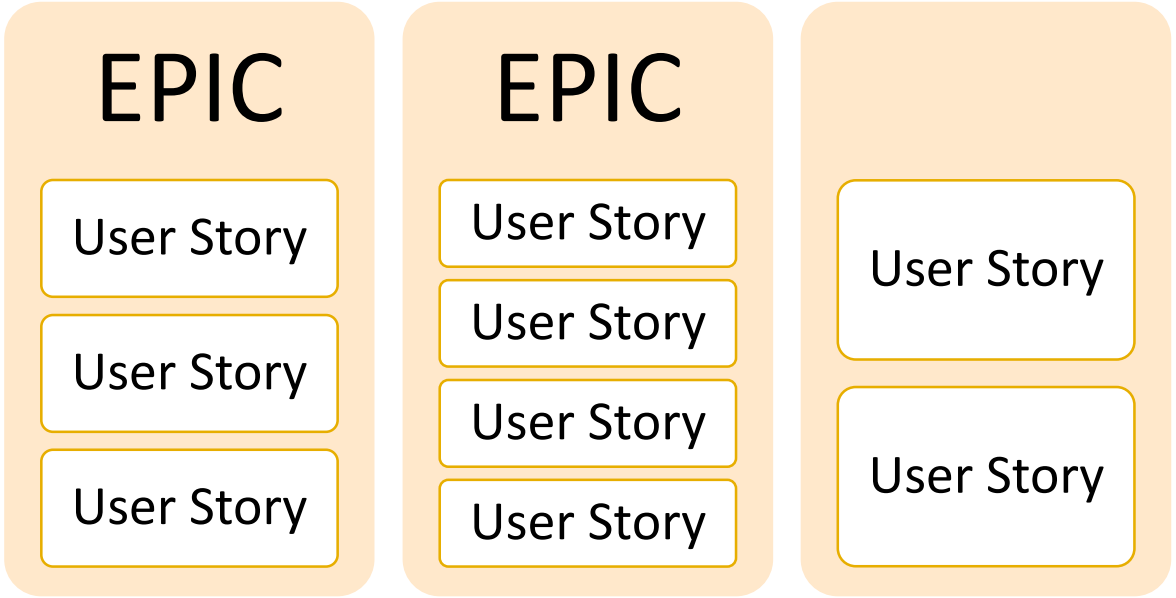
Acceptance Criteria

- Provides details on what the solution needs to accomplish
- Describes how the user story can be verified to be complete

User Story	Acceptance Criteria
As a customer, I want to be able to reset my password via the internet, so that I can reset it without having call into the customer service line.	1. Customer can indicate on the website they want to reset their password 2. Customer is asked to enter their email address 3. If the entered email address is not in the database, the customer is informed 4. If the entered email address is in the database, the customer is sent an email with a temporary password 5. Customer can log in with their username and temporary password 6. Customer is forced to change their password upon first successful login with their temporary password 7. If the temporary password is not used in 24 hours, it shall be expired and not allow the user to login with it

Introduction to epics

- Describes at a broader view what the user will get from the product
- Contain multiple user stories (usually 5-10)
- Not all user stories need to be tied to an epic



An epic example

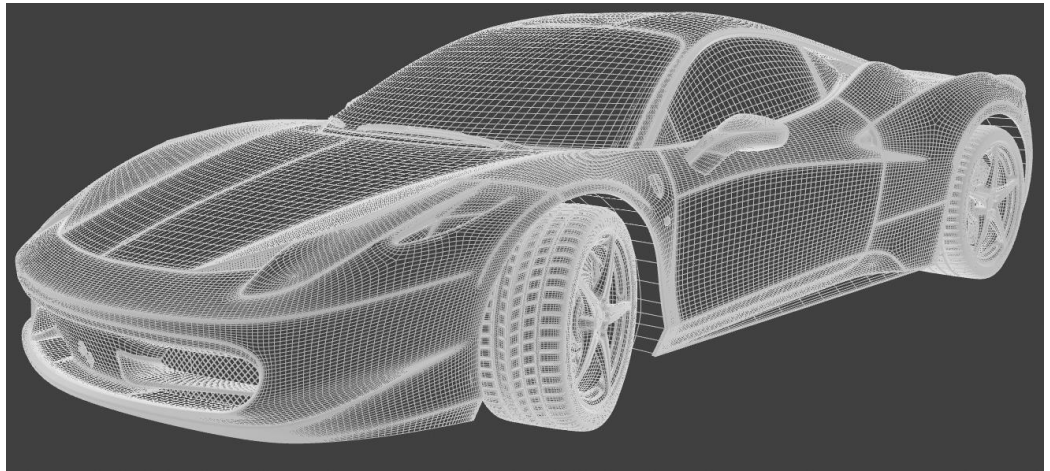
Epic: Customers should be able to self-serve on our website
User Story
As a customer, I want to be able to reset my password via the internet, so that I can reset it without having call into the customer service line.
As a customer, I want to be able to search for Frequently Asked Questions about the product, so that I can get answers to my questions quickly.
As a customer, I want to be able to see store locations, so that I can find a store location close to me

What is a model?

- The requirement models are created as part of the analysis activity *Define requirements*, although remember that the analysis activities are actually done in parallel with design and implementation and in each iteration of the project.
- A **model** is a representation of either reality or vision. Just as "a picture is worth a thousand words", most models use pictures to represent the reality or vision.

Why do we need models?

- Models are **abstractions**, which allow people to concentrate on the essentials of a system by omitting non-essential details.
- Models are **simplifications**: they help us to understand large, multi-faceted, complex systems.
- Models help us to **solve problems** without having to build concrete artifacts (e.g. we can build a model of a bridge before actually building the bridge – very helpful).
- Models provide a **common language** of description (modelling language) so that we can communicate the complexity of systems to others.



Modelling languages

- **Modelling languages** are systems of **notation** that use one or more **diagramming techniques**.
- A diagramming technique consists of a set of graphical symbols for representing **objects**, **processes** and **relationships** in information systems.
- Modelling languages and diagramming techniques are designed to be:
  - Clear and expressive enough to represent complex reality
  - Unambiguous
  - Universal
  - Easy to produce
  - Supported by drawing tools

Modelling languages

- There are a number of different modelling languages available to designers of information systems. Amongst the most commonly encountered are:
  - Unified Modelling Language (UML)
  - Structured Systems Analysis and Design Method (SSADM)
  - Business Process Modelling
  - Jackson Structured Programming

User story Vs. Use case

- User stories and Use case are similar concepts in that they focus on the goals of the user, and they show the list of functions at the appropriate level of detail. In Agile development, an *Epic is the equivalent of a Use Case*.
- But they differ in the approach taken to identify them and in the amount of detail that is captured by the analyst.
  - User stories are favored by highly Agile system development methodologies, and they are turned over to the programmer analyst much earlier than use cases are. The Agile development philosophy is to work directly with users and avoid doing too much documentation.
  - In contrast, a use case approach traditionally meant analysts complete much documentation for each use case, focusing on detailed steps carried out by the user and the system. In practice, use cases can also be very brief for Agile development.

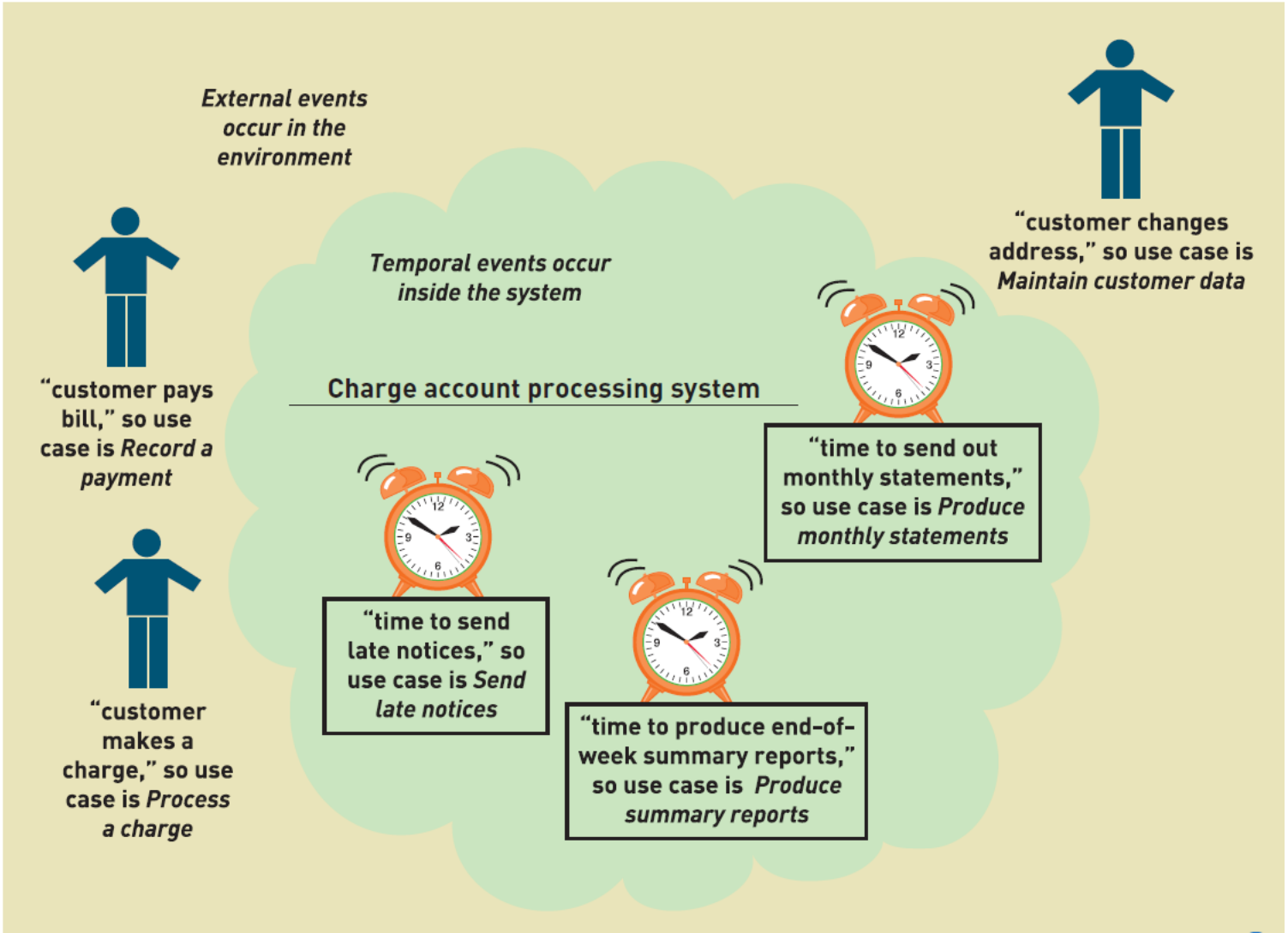
Use case

- A use case is an activity the system performs in response to a request by a user. Describes the functions of the system.
- Two techniques are recommended for identifying use cases:
  1. the user goal technique: to ask users to describe their goals for using the new or updated system.
  2. the event decomposition technique: begins by identifying all the business events the information system responds to, with each event leading to a use case.

Sample use cases for RMO

User	User goal and resulting use case
Potential customer	Search for item Fill shopping cart View product rating and comments
Marketing manager	Add/update product information Add/update promotion Produce sales history report
Shipping personnel	Ship order Track shipment Create item return

Events in a charge account processing system that lead to use cases



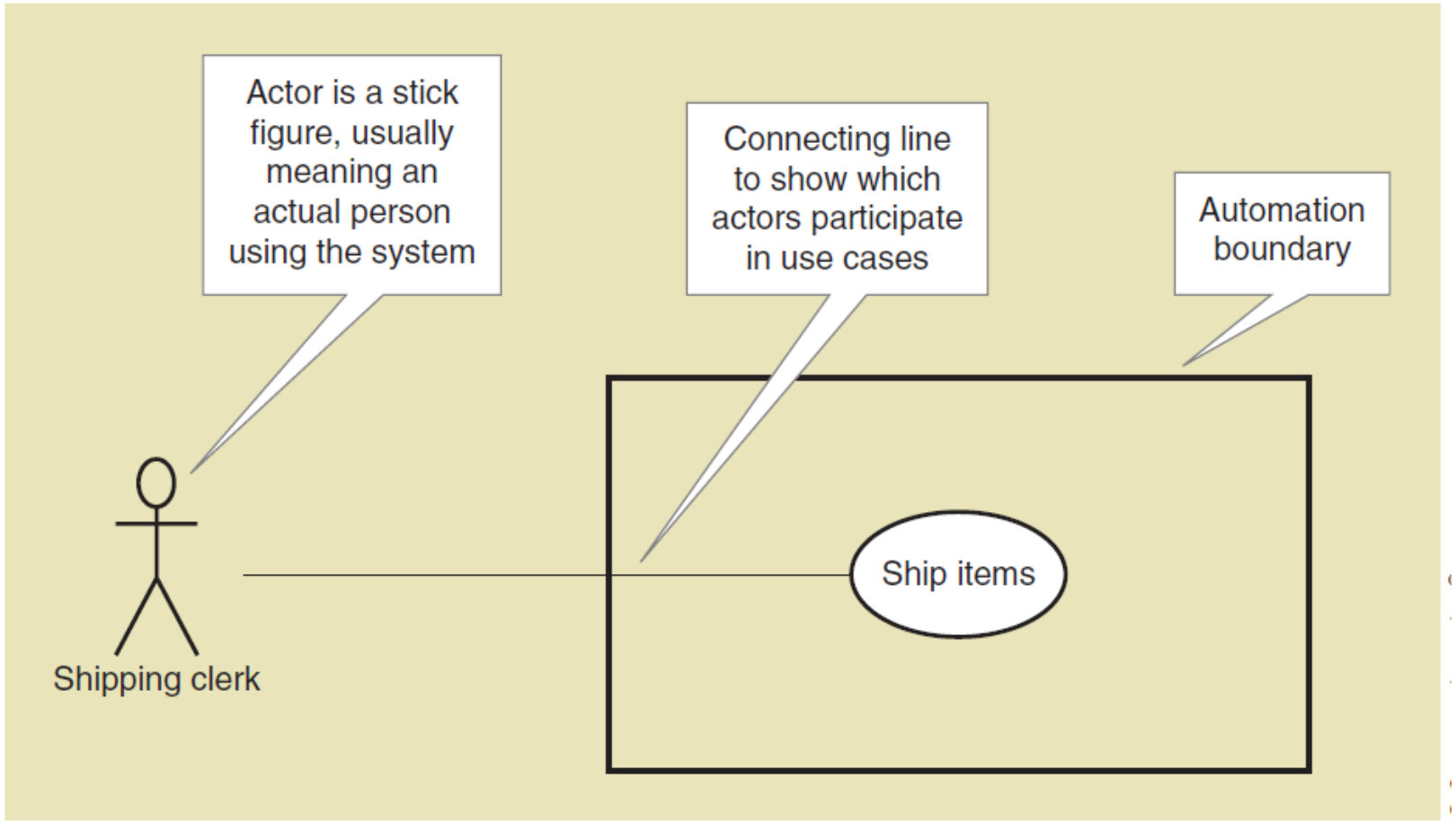
Sample Use Cases in the Ridgeline Mountain Outfitters Case

CSMS Sales Subsystem	
Use cases	**Users/actors**
Search for item	Customer, customer service representative, store sales representative
View product comments and ratings	Customer, customer service representative, store sales representative
View accessory combinations	Customer, customer service representative, store sales representative
Fill shopping cart	Customer
Empty shopping cart	Customer
Check out shopping cart	Customer
Fill reserve cart	Customer
Empty reserve cart	Customer
Convert reserve cart	Customer
Create phone sale	Customer service representative
Create store sale	Store sales representative

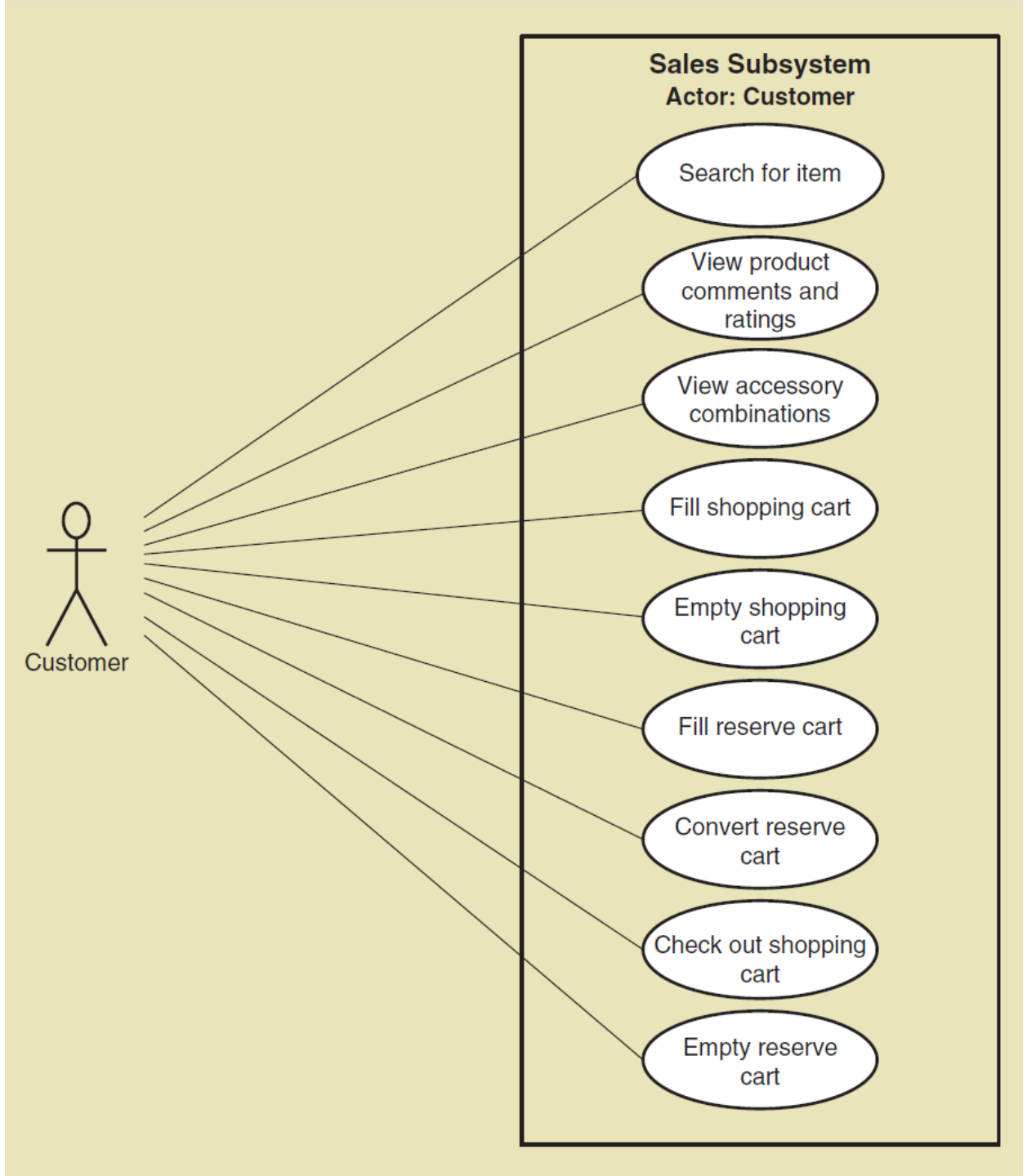
Use case diagram

- Sometimes, it is useful to create diagrams that visually depict use cases and how they are organized. The **use case diagram** is the UML model used to illustrate use cases and their relationship to users.
- **Actor** is a person who uses the system in UML, whom we have referred to up to this point as the user. Sometimes, the actor for a use case is not a person; instead, it can be another system or device that receives services from the system.
- The connecting line between the actor and the use case indicates that the actor is involved with that use case.
- **Automation boundary** defines the border between the computerized portion of the application and the people operating the application.

A simple use case diagram



All use cases involving the customer actor for the Sales subsystem



Use cases involving the customer service representative and store sales representative for the Sales subsystem