41951- ANÁLISE DE SISTEMAS

# UML Use case diagrams

Ilídio Oliveira

v2024-02-20

universidade de aveiro
departamento de eletrónica,
telecomunicações e informática
deti

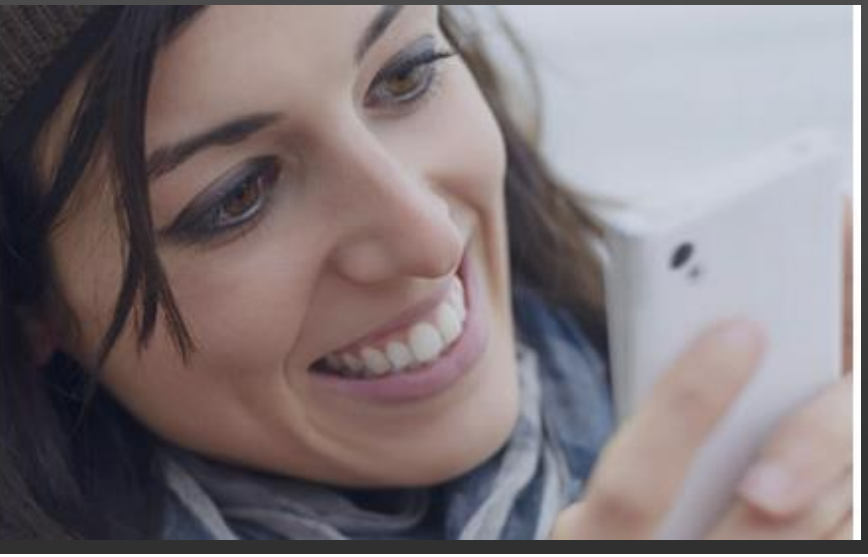
# Learning objectives for this lecture

- Explain the process used to discover use cases
- Read and create Use Case diagrams
- Explain the components of a use case description and the associated templates
- Create functional models of business processes using use-case diagrams and use case descriptions.

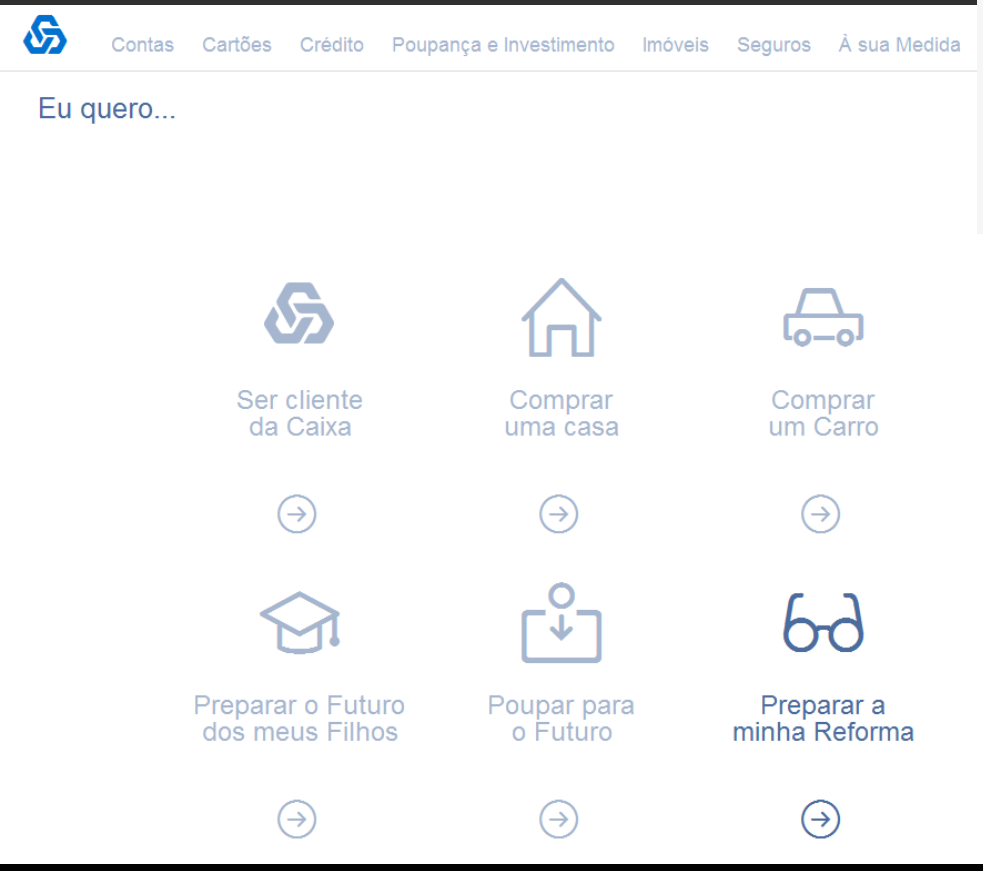
# Which main requirements elicitation approaches exist?

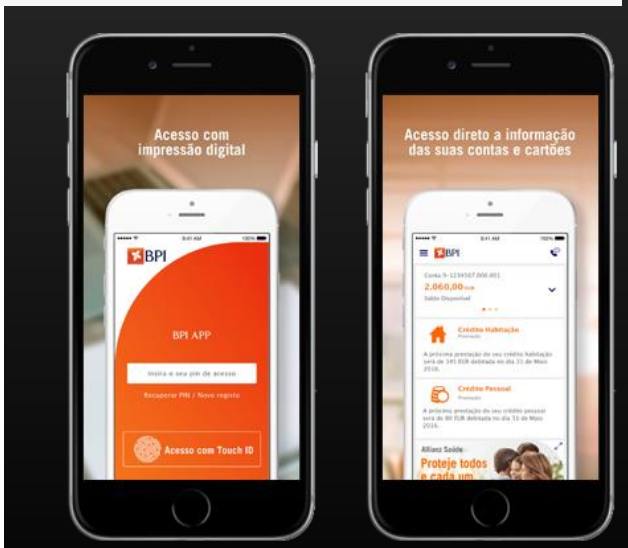
What is the goal the user wants to achieve?  vs.

What capability should the system process?



- Consulta de Saldos e Movimentos de Contas e Cartões de Crédito;
- Consulta de Posição Integrada;
- Transferências para beneficiários, contas BPI ou contas de outros Bancos (zona SEPA);
- Pagamentos de Serviços, Estado e Telemóveis;
- Criação e gestão de beneficiários de transferências e de pagamentos predefinidos;
- Constituição, reforço e mobilização de contas poupança objetivo;
- Cartões: pedido de alteração de Limites de Crédito, alteração de opção de pagamento e pagamento de Saldo ou Reforço;
- Consulta de catálogo e aquisição de Produtos Prestígio;
- Acesso a contactos, localização e serviços de Balcões, Centros de Investimento e Centros de Empresas;
- Login com código de 4 dígitos ou com impressão digital.

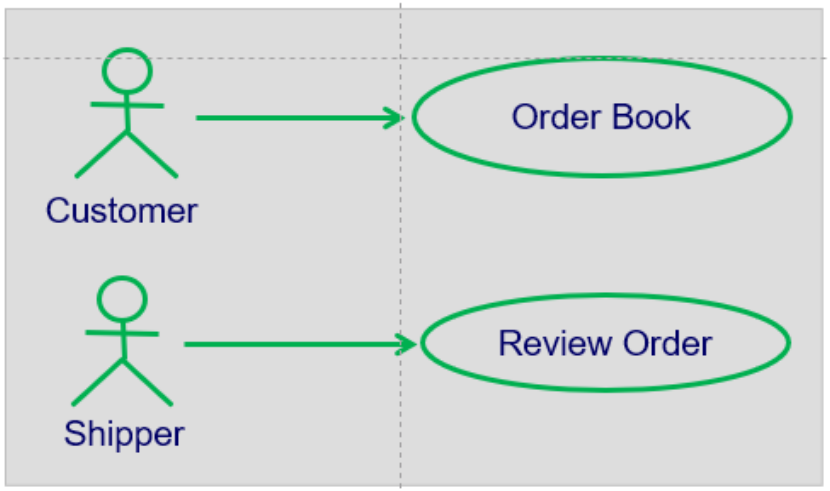


# OpenUP recommended practices

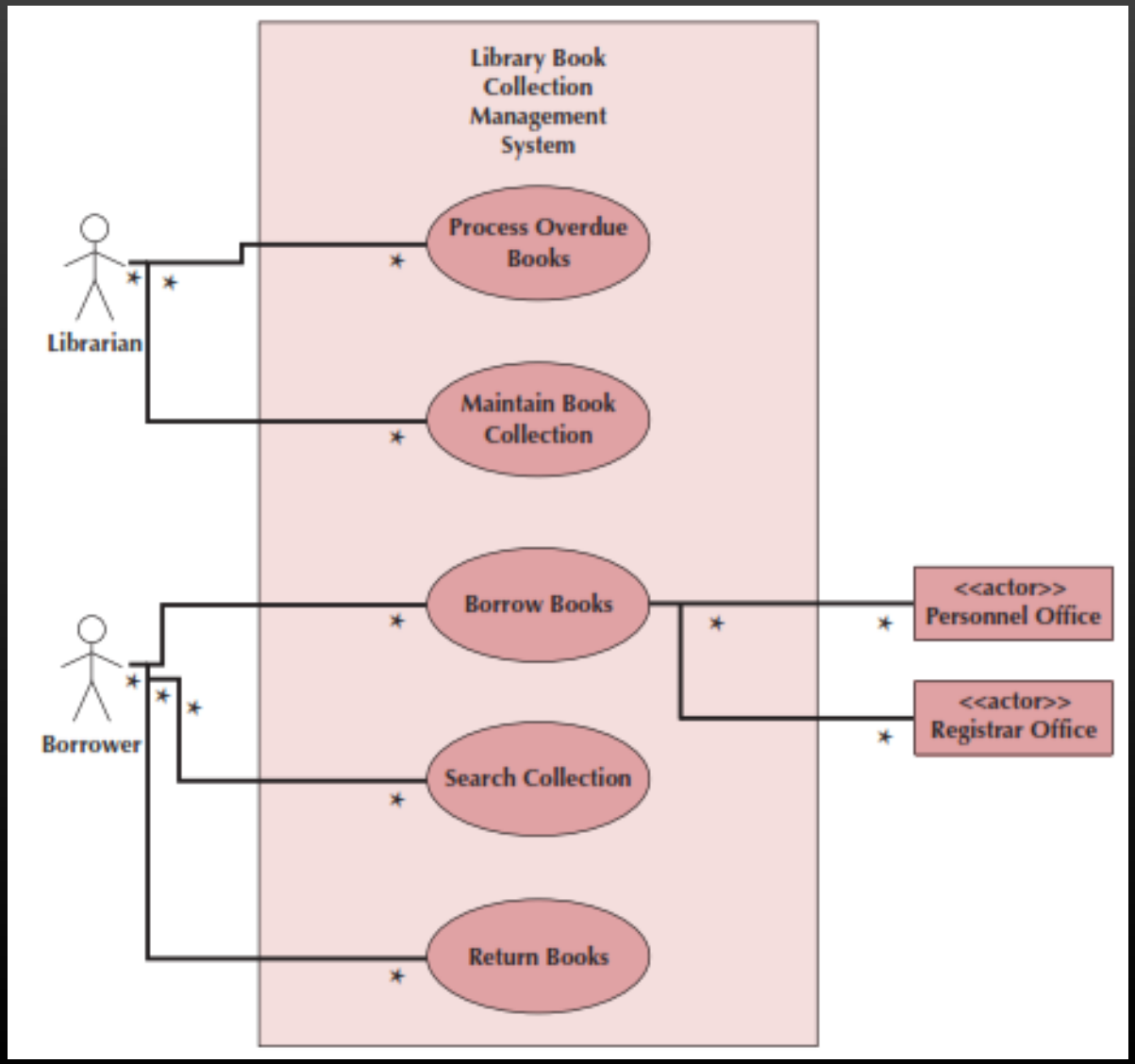
## *Use Case Driven Development*

- This practice describes how to capture requirements with a combination of use cases and system-wide requirements, and then drive development and testing from those use cases.



# Example Use-Case

Library Book Collection Management - System Use Case Diagram




Credit: Dennis et al, "*Systems Analysis and Design: An Object Oriented Approach with UML*", 5th ed.

## Who (the actor) does what (interaction) on the system, whith a goal in mind (motivation)

UC is a flow of actions that produces a result with values to a particular actor (including the variations in the flow, related to the same goal)

The required yet sufficient amount of activity (interaction) that produces a result of interest for an actor

UC provides a context to a related set of requirements. (Favors a more coherent division of the system)

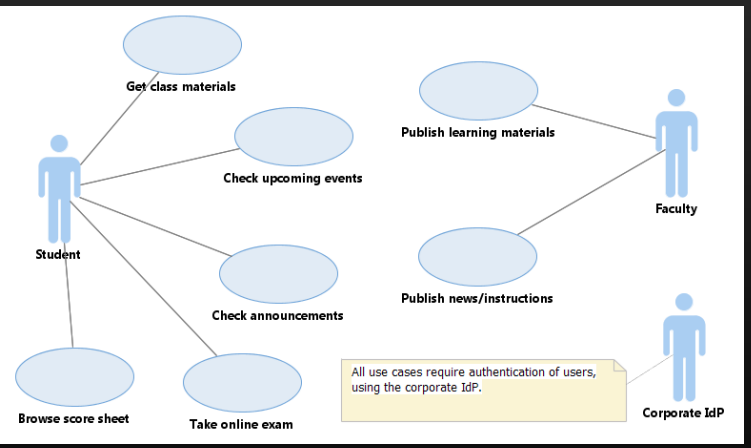
# Use-cases discovery (in-class...)

# The use case model includes an overview (diagrams) + description of scenarios

# Use cases documentation (use cases modeling)

## Visual overview

Use-cases Diagram (UML)




## Description of scenarios

Structured narratives (text)

Optional: activity models.

| | | | |
|---|---|---|---|
| ID and Name: | UC-4 Request a Chemical | | |
| Created By: | Lori | Date Created: | 8/22/13 |
| Primary Actor: | Requester | Secondary Actors: | Buyer, Chemical Stockroom, Training Database |
| Description: | The Requester specifies the desired chemical to request by entering its name or chemical ID number or by importing its structure from a chemical drawing tool. The system either offers the Requester a container of the chemical from the chemical stockroom or lets the Requester order one from a vendor. | | |
| Trigger: | Requester indicates that he wants to request a chemical. | | |
| Preconditions: | PRE-1. User's identity has been authenticated.<br>PRE-2. User is authorized to request chemicals.<br>PRE-3. Chemical inventory database is online. | | |
| Postconditions: | POST-1. Request is stored in the CTS.<br>POST-2. Request was sent to the Chemical Stockroom or to a Buyer. | | |
| Normal Flow: | **4.0 Request a Chemical from the Chemical Stockroom**<br>1. Requester specifies the desired chemical.<br>2. System lists containers of the desired chemical that are in the chemical stockroom, if any.<br>3. System gives Requester the option to View Container History for any container.<br>4. Requester selects a specific container or asks to place a vendor order (see 4.1).<br>5. Requester enters other information to complete the request.<br>6. System stores the request and notifies the Chemical Stockroom. | | |
| Alternative Flows: | **4.1 Request a Chemical from a Vendor**<br>1. Requester searches vendor catalogs for the chemical (see 4.1.E1).<br>2. System displays a list of vendors for the chemical with available container sizes, grades, and prices.<br>3. Requester selects a vendor, container size, grade, and number of containers.<br>4. Requester enters other information to complete the request.<br>5. System stores the request and notifies the Buyer. | | |
| Exceptions: | **4.1.E1 Chemical Is Not Commercially Available**<br>1. System displays message: No vendors for that chemical.<br>2. System asks Requester if he wants to request another chemical (3a) or to exit (4a).<br>3a. Requester asks to request another chemical.<br>3b. System starts normal flow over.<br>4a. Requester asks to exit.<br>4b. System terminates use case. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 times per week by each chemist, 200 times per week by chemical | | |

# Types of Use Cases

| Purpose | | Amount of information: Overview/High-level | Amount of information: Detail |
|---|---|---|---|
| | Essential | High-level **overview** of issues **essential** to understanding required functionality | **Detailed** description of issues **essential** to understanding required functionality |
| | Real | High-level **overview** of a specific set of steps performed on the **real** system once implemented | **Detailed** description of a specific set of steps performed on the **real** system once implemented |

Credit: Dennis et al, "*Systems Analysis and Design: An Object Oriented Approach with UML*", 5th ed.

# Use case model

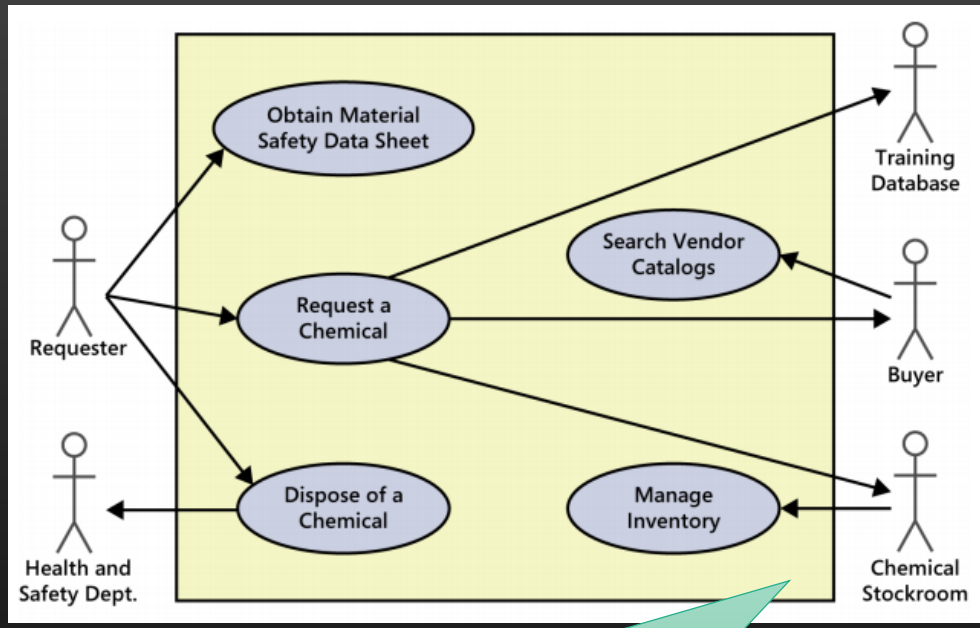
The **users (or systems)** who interact with the system with some objective in mind are modeled as **actors**.

The ways/episodes in which the system will be used to achieve these goals are modeled as **use cases**.

A **use case diagram** is a model of the useful ways to use a system. This allows you to quickly grasp the scope of the system – what is included and what is not – and give the team a global view of what the system will do.

The intentional high-level perspective offers a birds-eye view, without losing ourselves in the details of the internal parts of the system.

→ A context/instrument to discuss and discover the system requirements!




Note: we will not use "arrows" in use-case/actor associations

# Elements of the use case model

## Actor

Any entity (the role of someone, another system,...) external to the system under specification, which interacts with it

## Scenario

A particular situation/history of use of the system, i.e., a possible path in the execution of a case of use

E.g.: payment of the purchase with cash or card; failure in payment due to lack of card authorization;...

## Use case

Set of scenarios related to the same goal

An episode of using + variants.

## Associations

Relationships of interest between actors/CaU, CaU/CaU, actors/actors

Most relevant:

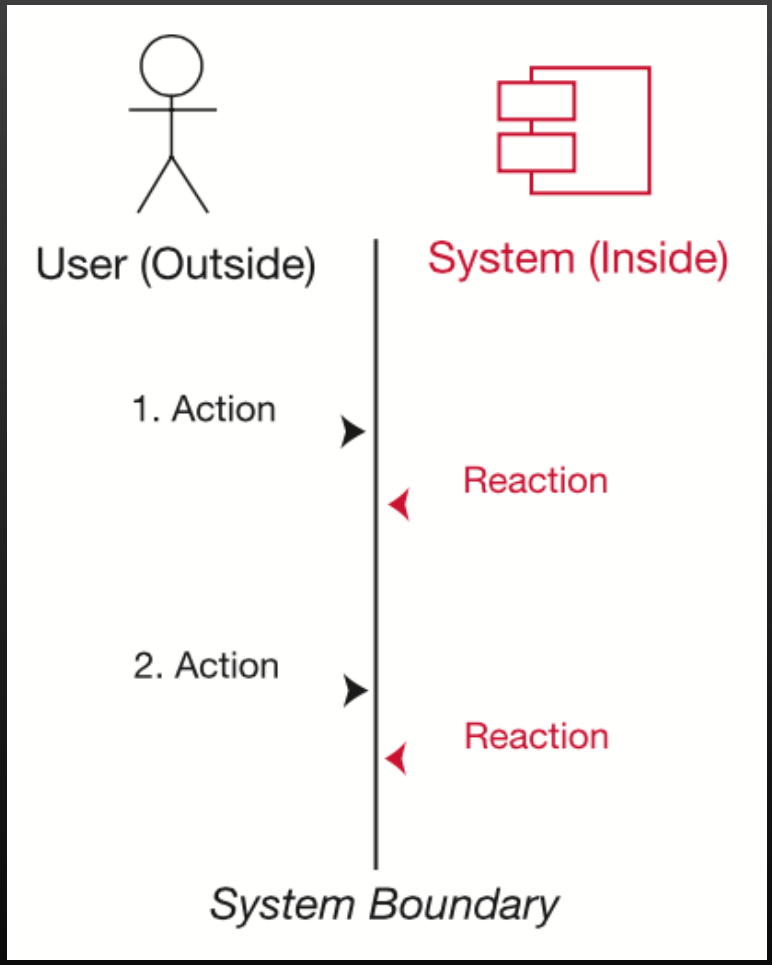
Actor *participates in* a use case.

# Use cases hold/encapsulate interaction scenarios

# The use case captures a dialog between the actor(s) and the system

**CaU: Pay at checkout**

1. Customer arrives at POS checkout with goods to purchase.
2. Cashier starts a new sale.
3. Cashier enters item identifier.
4. System records sale line item and presents item description, price, and running total. Price is calculated from a set of price rules.
   Cashier repeats steps 3-4 until indicates done.
5. System presents total with taxes calculated.
6. Cashier tells Customer the total and asks for payment.
7. Customer pays and System handles payment.
8. System logs completed sale and sends sale and payment information to the external Accounting system (for accounting and commissions) and Inventory system (to update inventory).
9. System presents receipt.
10. Customer leaves with receipt and goods (if any)



# The Use Case has several flows

### The typical flow

The “normal script” for the actor/system collaboration

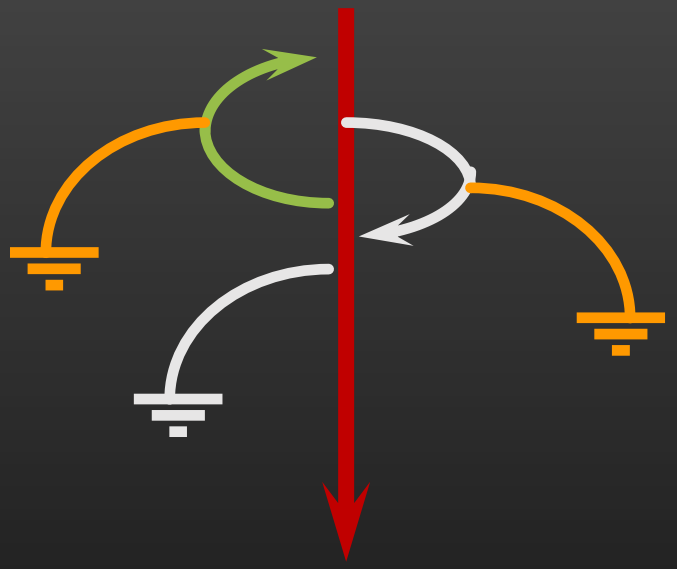
### Several alternative flows

Variations due to options from the users

Uncommon/special cases

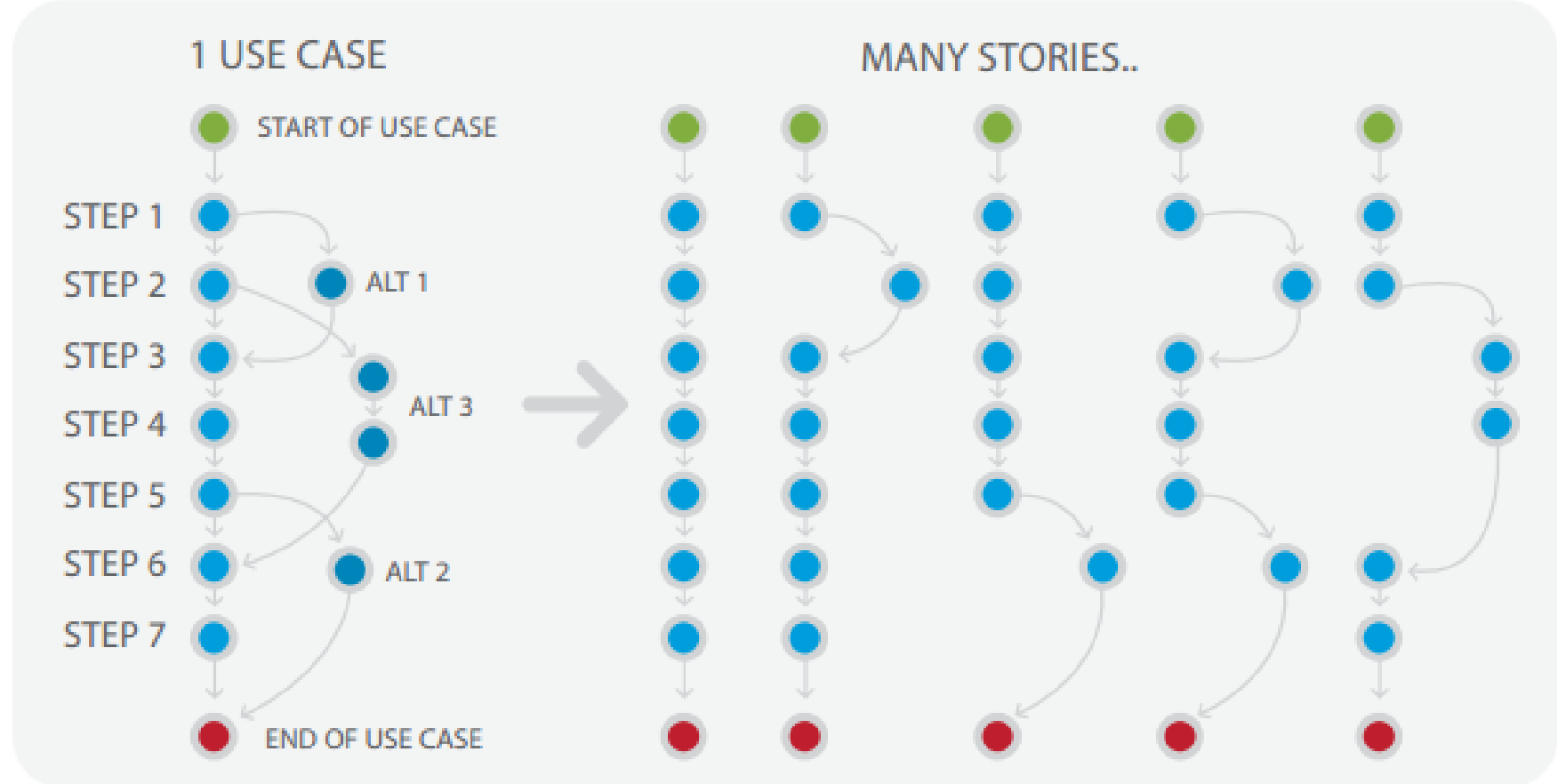
Exception conditions and errors (and what should be done)

e.g.: how to enter the product description (in the sale scenario) when the bar code can not be read?



# One use case has different flows

Still, all flows are related to the same goal in-mind

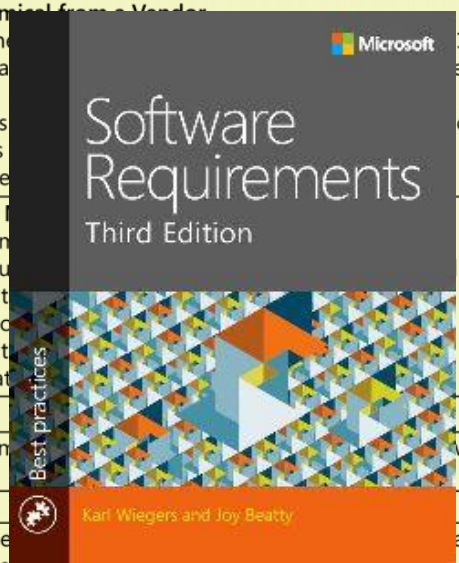


# **Essential elements** of a use case specification

- ✓ A unique identifier and a succinct name that states the user goal
- ✓ A brief textual description that describes the purpose of the use case
- ✓ A trigger condition that initiates execution of the use case
- ✓ Zero or more preconditions that must be satisfied before the use case can begin
- ✓ One or more postconditions that describe the state of the system after the use case is successfully completed
- ✓ A numbered list of steps that shows the sequence of interactions between the actor and the system — a dialog— that leads from the preconditions to the postconditions

→ Credit: Wiegers 2013

| | | | |
|---|---|---|---|
| ID and Name: | UC-4 Request a Chemical | | |
| Created By: | Lori | Date Created: | 8/22/13 |
| Primary Actor: | Requester | Secondary Actors: | Buyer, Chemical Stockroom, T |
| Description: | The Requester specifies the desired chemical to request by entering number or by importing its structure from a chemical drawing tool. the Requester a container of the chemical from the chemical stockr order one from a vendor. | | |
| Trigger: | Requester indicates that he wants to request a chemical. | | |
| Preconditions: | PRE-1. User's identity has been authenticated.<br>PRE-2. User is authorized to request chemicals.<br>PRE-3. Chemical inventory database is online. | | |
| Postconditions: | POST-1. Request is stored in the CTS.<br>POST-2. Request was sent to the Chemical Stockroom or to a Buyer. | | |
| Normal Flow: | **4.0 Request a Chemical from the Chemical Stockroom**<br>1. Requester specifies the desired chemical.<br>2. System lists containers of the desired chemical that are in the che<br>3. System gives Requester the option to View Container History for<br>4. Requester selects a specific container or asks to place a vendor or<br>5. Requester enters other information to complete the request.<br>6. System stores the request and notifies the Chemical Stockroom. | | |
| Alternative Flows: | **4.1 Request a Chem**<br>1. Requester search<br>2. System displays a<br>and prices.<br>3. Requester selects<br>4. Requester enters<br>5. System stores the | | |
| Exceptions: | **4.1.E1 Chemical Is**<br>1. System displays n<br>2. System asks Requ<br>3a. Requester asks t<br>3b. System starts nc<br>4a. Requester asks t<br>4b. System terminat | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 tin<br>stockroom staff | | |
| Business Rules: | BR-28, BR-31 | | |
| Other Information: | The system must be | | |



# **"Request a Chemical"** use case specification

| | | | |
|---|---|---|---|
| ID and Name: | UC-4 Request a Chemical | | |
| Created By: | Lori | Date Created: | 8/22/13 |
| Primary Actor: | Requester | Secondary Actors: | Buyer, Chemical Stockroom, Training Database |
| Description: | The Requester specifies the desired chemical to request by entering its name or chemical ID number or by importing its structure from a chemical drawing tool. The system either offers the Requester a container of the chemical from the chemical stockroom or lets the Requester order one from a vendor. | | |
| Trigger: | Requester indicates that he wants to request a chemical. | | |
| Preconditions: | PRE-1. User's identity has been authenticated.<br>PRE-2. User is authorized to request chemicals.<br>PRE-3. Chemical inventory database is online. | | |
| Postconditions: | POST-1. Request is stored in the CTS.<br>POST-2. Request was sent to the Chemical Stockroom or to a Buyer. | | |
| Normal Flow: | **4.0 Request a Chemical from the Chemical Stockroom**<br>1. Requester specifies the desired chemical.<br>2. System lists containers of the desired chemical that are in the chemical stockroom, if any.<br>3. System gives Requester the option to View Container History for any container.<br>4. Requester selects a specific container or asks to place a vendor order (see 4.1).<br>5. Requester enters other information to complete the request.<br>6. System stores the request and notifies the Chemical Stockroom. | | |
| Alternative Flows: | **4.1 Request a Chemical from a Vendor**<br>1. Requester searches vendor catalogs for the chemical (see 4.1.E1).<br>2. System displays a list of vendors for the chemical with available container sizes, grades, and prices.<br>3. Requester selects a vendor, container size, grade, and number of containers.<br>4. Requester enters other information to complete the request.<br>5. System stores the request and notifies the Buyer. | | |
| Exceptions: | **4.1.E1 Chemical Is Not Commercially Available**<br>1. System displays message: No vendors for that chemical.<br>2. System asks Requester if he wants to request another chemical (3a) or to exit (4a). | | |

# The use case details describe an interaction

## HOW TO WRITE A USE CASE: THE THREE MAGIC QUESTIONS

Well, OK, this whole chapter describes how to write a use case. But when writing use cases, you need to keep asking the following three fundamental questions:[1]

1. What happens?

   (This gets your "sunny-day scenario" started.)

2. And then what happens?

   (Keep asking this question until your "sunny-day scenario" is complete.)

3. What else might happen?

   (Keep asking this one until you've identified all the "rainy-day scenarios" you can think of, and described the related behavior.)

***BASIC COURSE:***

*The Customer clicks the Write Review button for the book currently being viewed, and the system shows the Write Review screen. The Customer types in a Book Review, gives it a Book Rating out of five stars, and clicks the Send button. The system ensures that the Book Review isn't too long or short, and that the Book Rating is within one and five stars. The system then displays a confirmation screen, and the review is sent to a Moderator, ready to be added.*

***ALTERNATE COURSES:***

***User not logged in**: The user is first taken to the Login screen and then to the Write Review screen once he is logged in.*

***The user enters a review that is too long (text > 1MB)**: The system rejects the review and responds with a message explaining why the review was rejected.*

***The review is too short (< 10 characters)**: The system rejects the review.*

| Use case: | Brief description: |
|---|---|
| Create new assigment | The Teaching Staff creates a new Activity of type Assignment, directly inserting it in the page layout. The assignment must define a title and a time period for submissions and can be configured to work with individual or group submissions. The assignment is listed in the student view and on the specified date (or immediately, if none is given) accepts submissions from registered students. |

| | |
|---|---|
| **Use case:** | Add new assigment |
| **Brief description:** | The Faculty creates assignments for students, directly inserting it in the course page. The assignment defines a time period for submissions and can be configured to work with individual or group submissions. The assignment is listed in the student view and on the specified date (or immediately, if none is given) accepts submissions from students. |
| **Basic flow:** | 1. Log-in using corporate IdP.<br>2. Select desired course.<br>3. Turn editing mode on.<br>4. Add Assignment activity in the page layout.<br>5. Configure Assignment activity.<br>6. Commit changes. |
| **Alternative flows:** | Step 1: IdP unavailable.<br>Step 4/5: Instead of a new, empty assignment, the user may reuse an existing one. |
| **Open issues:** | Step 3/4. The course is closed. Are changes allowed to past courses?<br>Step 5. The browser does not accept the rich text editor. Default to plain text? |



→ More info...

# Use Case Writing Guidelines

1. Write in the form of subject-verb-direct object
2. Make sure it is clear who the initiator of the step is
3. Write from independent observer's perspective
4. Write at about the same level of abstraction
5. Ensure the use case has a sensible set of steps
6. Apply the KISS principle liberally.
7. Write repeating instructions after the set of steps to be repeated

Credit: Dennis et al, "*Systems Analysis and Design: An Object Oriented Approach with UML*", 5th ed.

# How to discover the Use Cases?

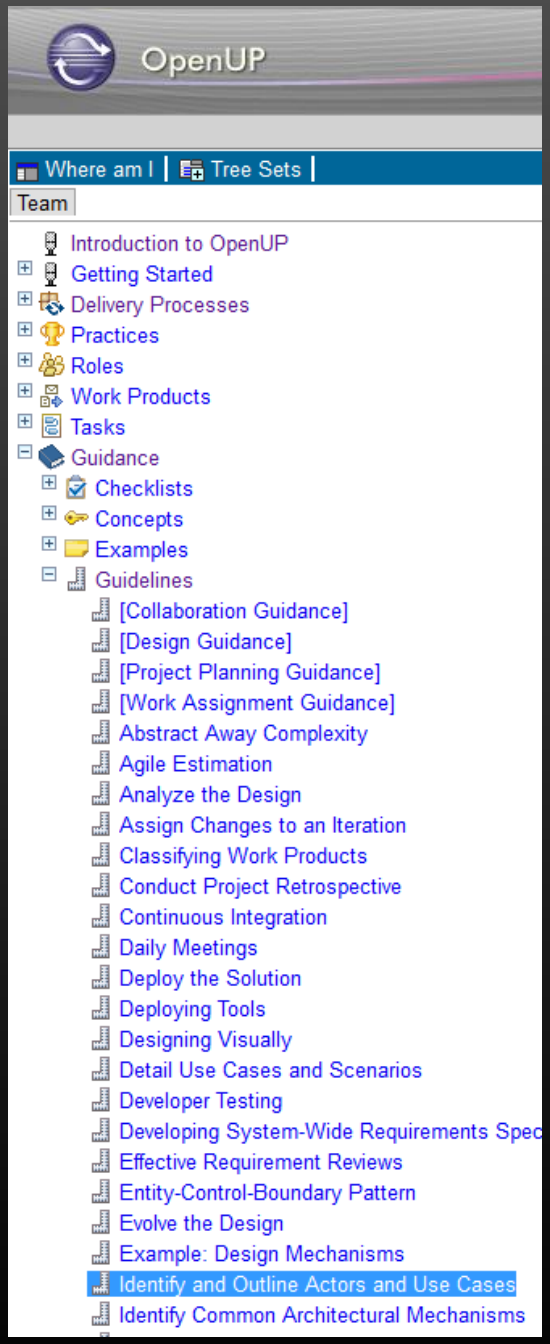
Identify the system boundary

Identify the actors who somehow interact with the system

For each actor, identify the objectives/motivations to use the system

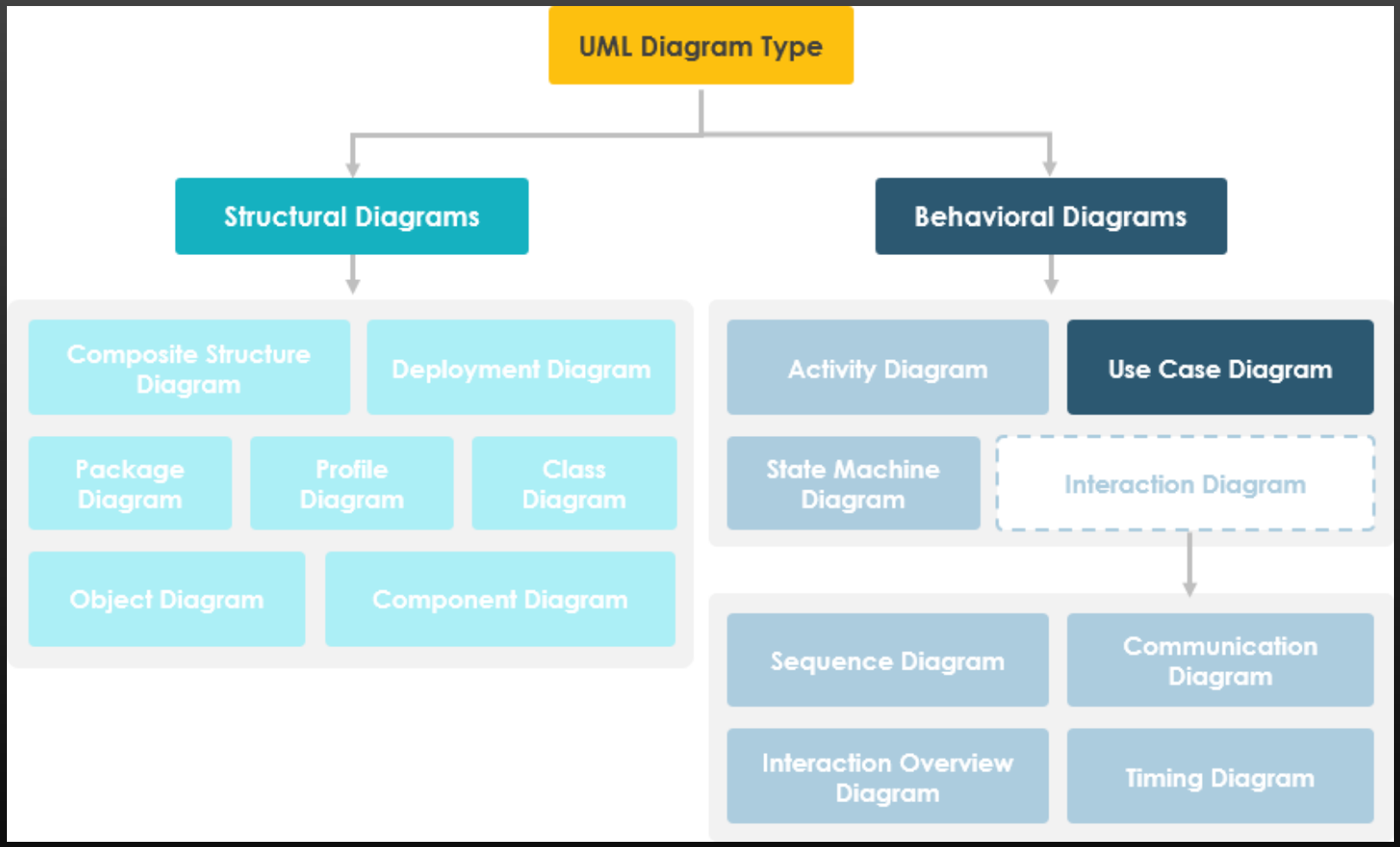
Define CaU that satisfies the objectives of the actors

Give names that reflect the motivation of the actor




**Guideline: Identify and Outline Actors and Use Cases**

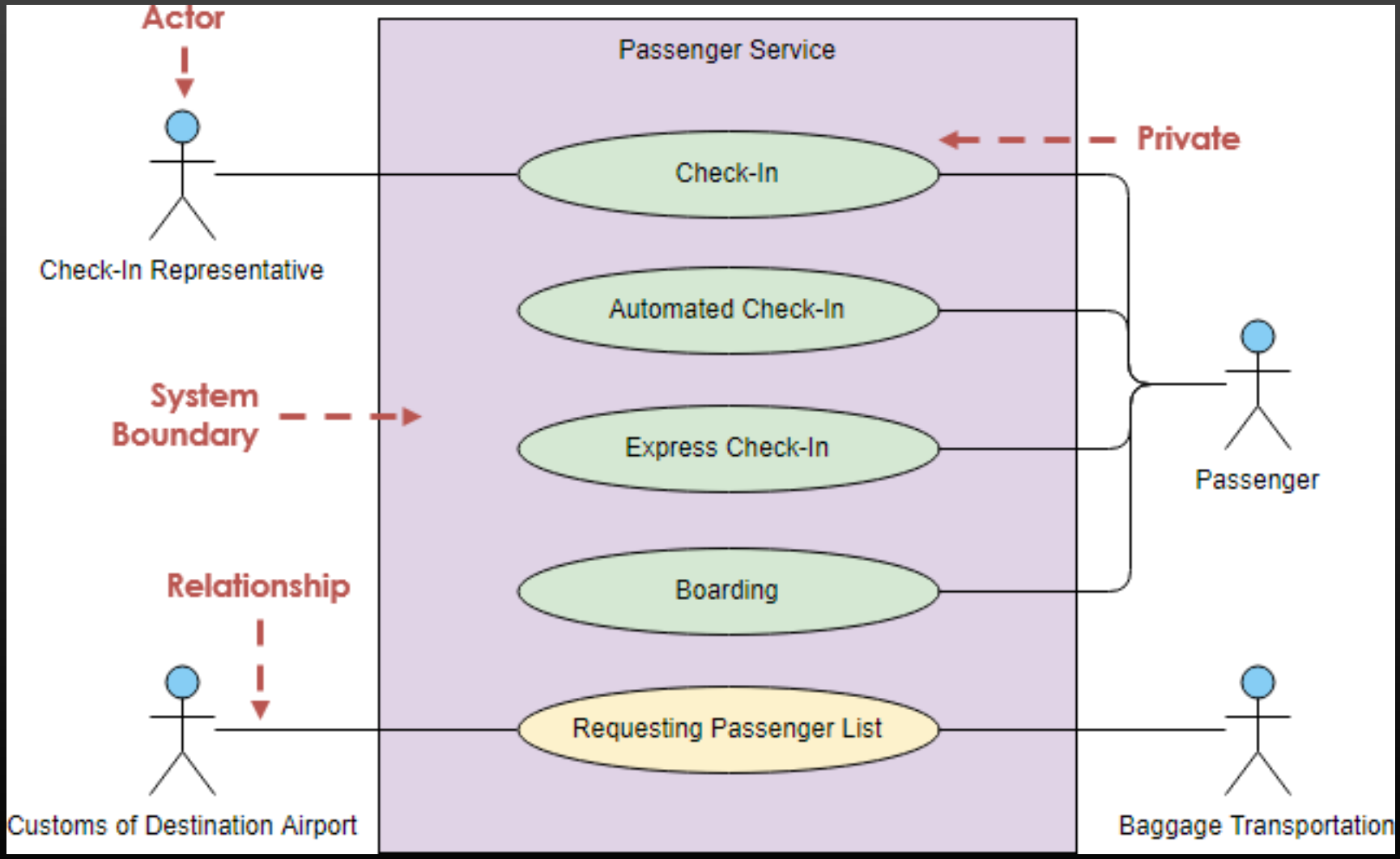
# UML support




https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/

# Elements of the UC diagram

See also: tutorial on use-case diagrams

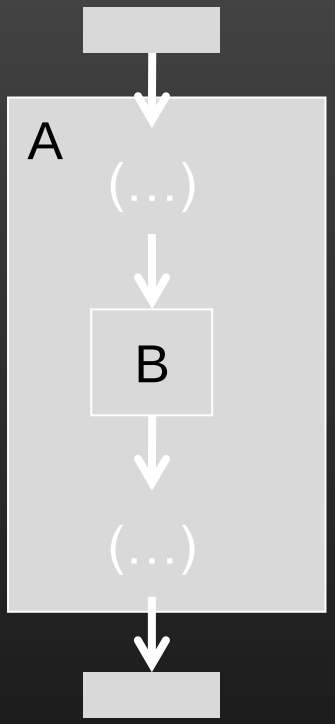


# Reusing behavior with **include**

**A-story includes the B-story**

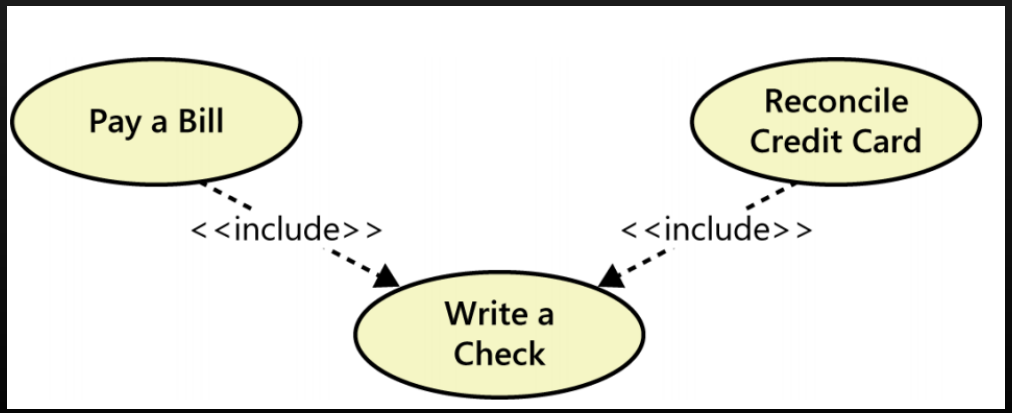
The behavior in A always includes the behavior modeled in B

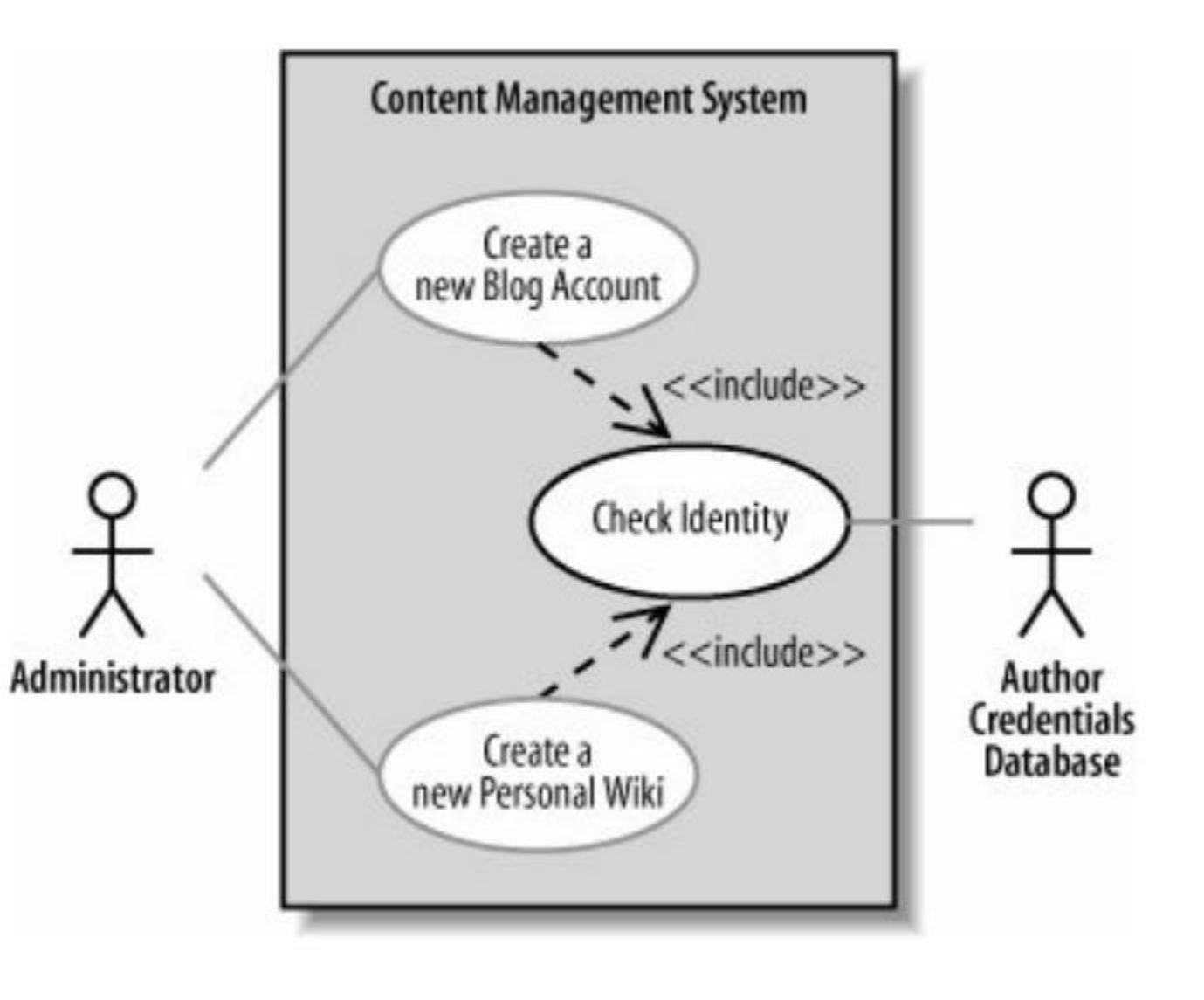
Can help with "factoring out" common behavior




**Include** is the stereotype of the dependency relationship

- Not a verb here



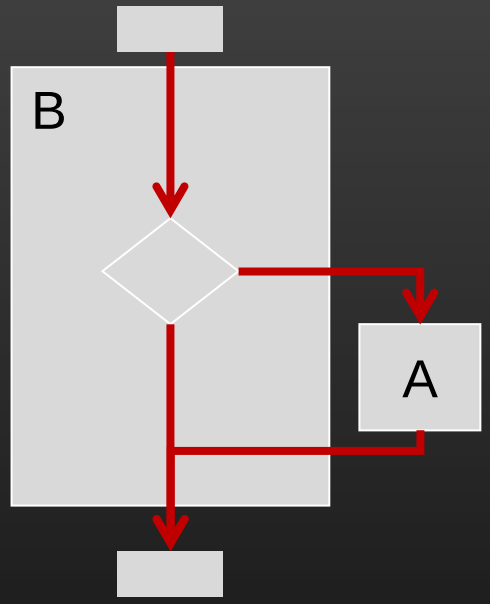

Content Management System
Create a new Blog Account
<<include>>
Check Identity
<<include>>
Create a new Personal Wiki
Administrator
Author Credentials Database

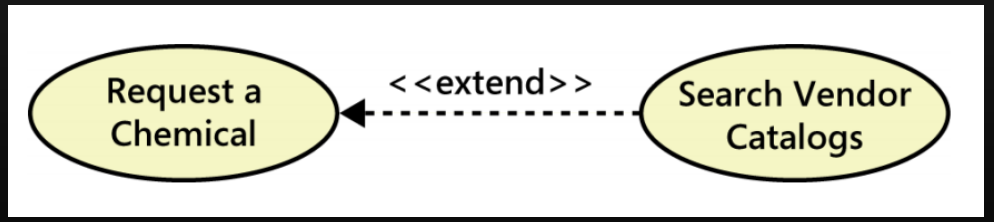
# Optional behavior activation with **extend**

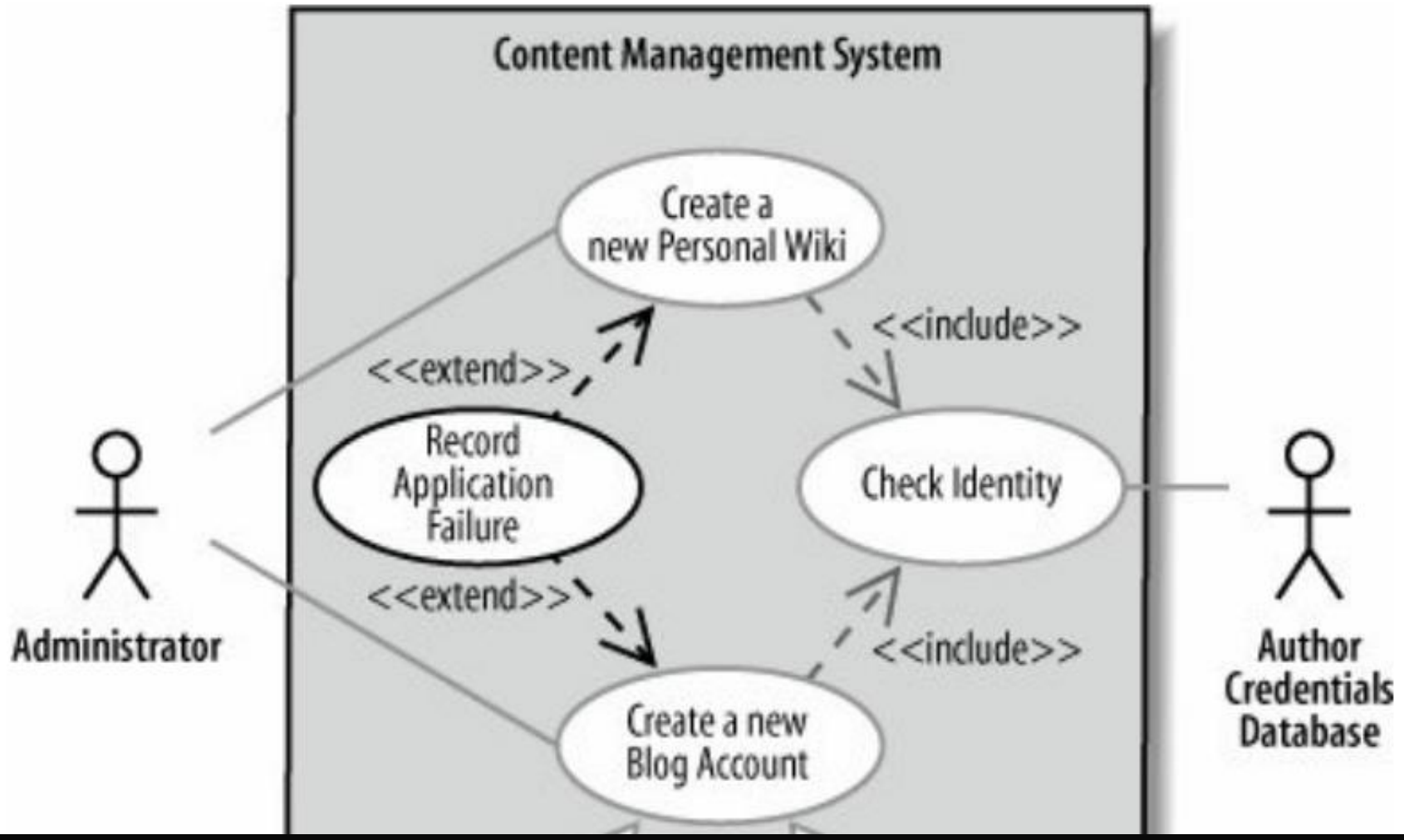
**A-story may extend B-story**

The behavior of B can incorporate the behavior of A, depending on the verification of an "extension condition" (*extension point*)

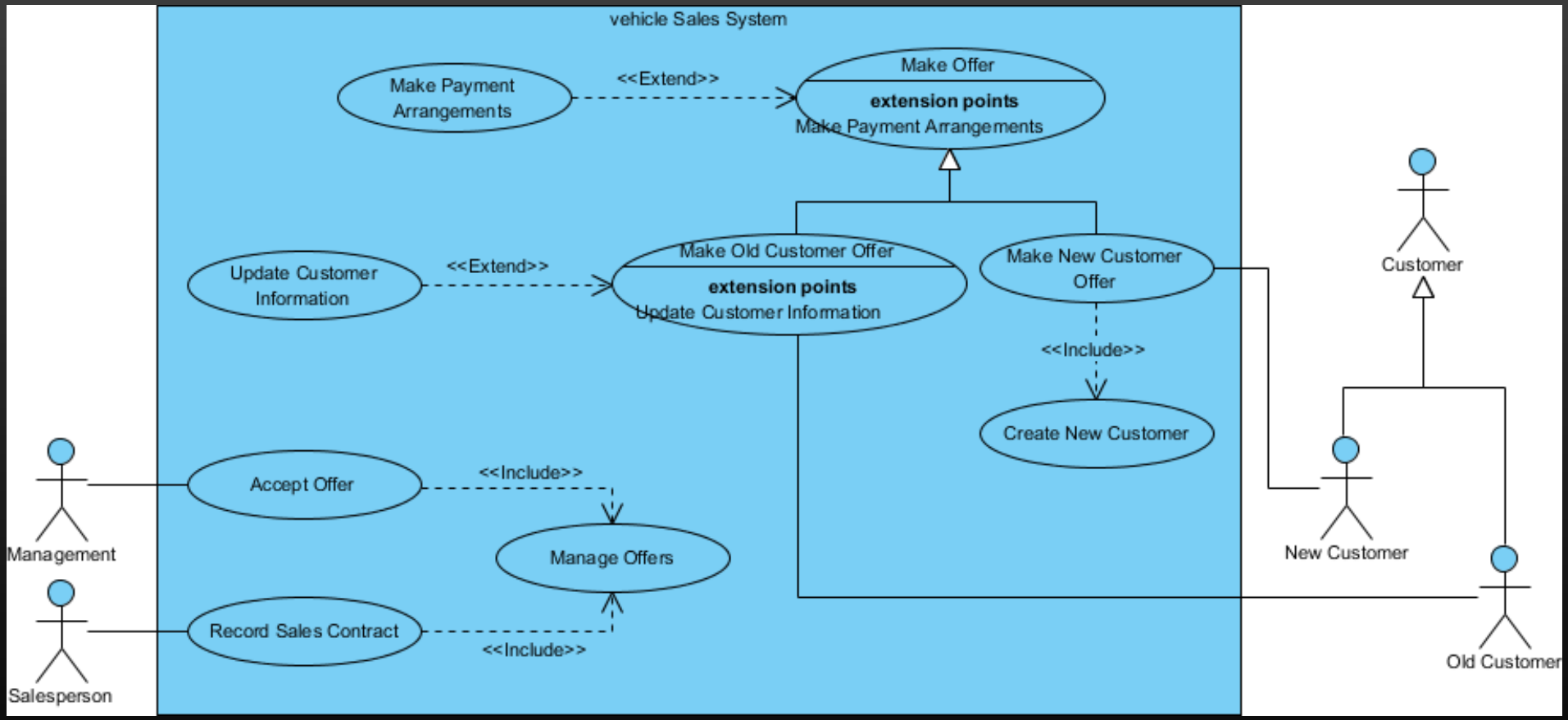
Unlike the include relationship, extend models optional/conditional behavior



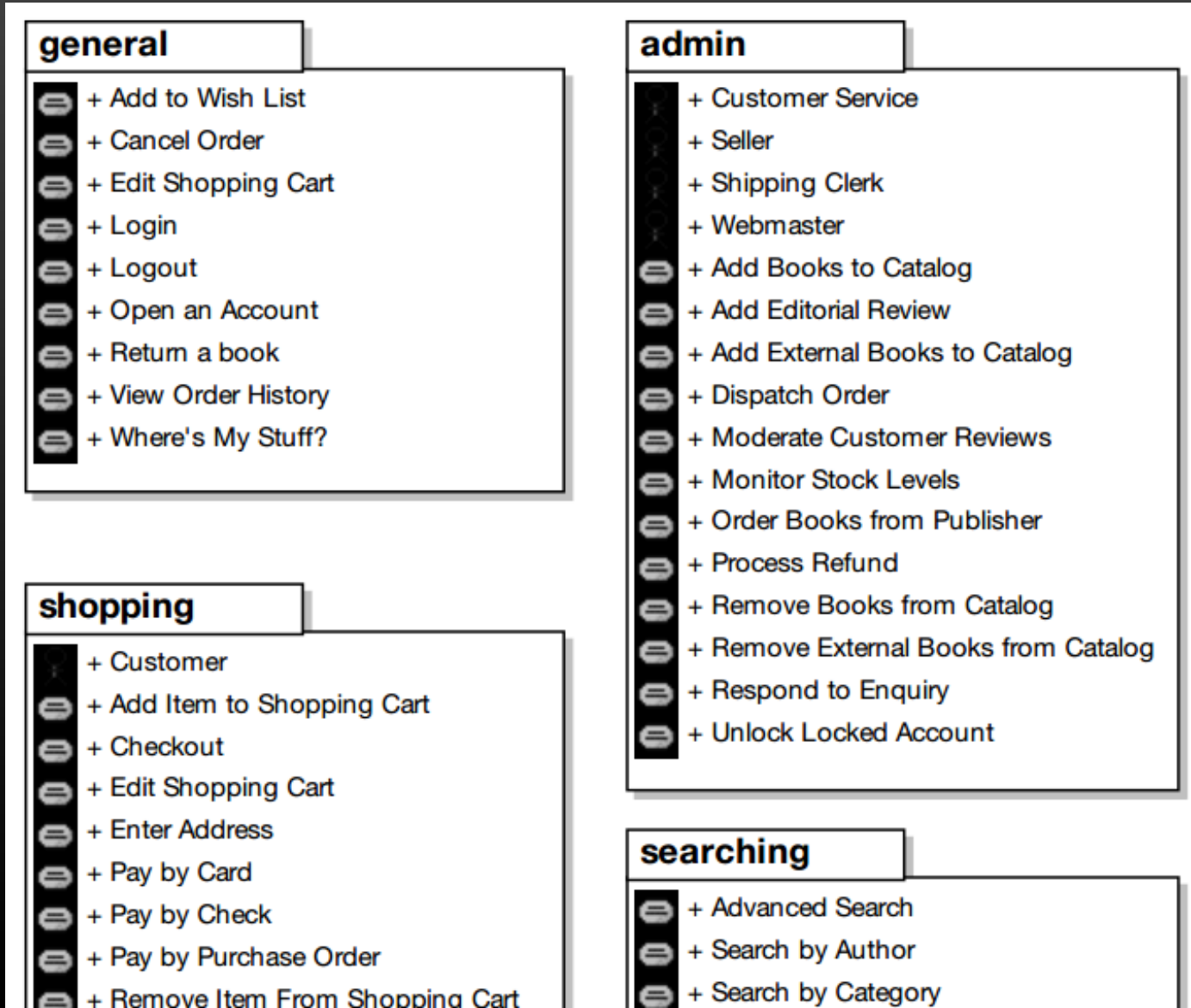




Content Management System
Create a new Personal Wiki
<<include>>
<<extend>>
Record Application Failure
Check Identity
<<extend>>
<<include>>
Create a new Blog Account
Administrator
Author Credentials Database

# Sample (full syntax)



# The use cases may be organized in coherent functional groups → using packages



# Readings & references

| Core readings | Suggested readings |
| --- | --- |
| • [Dennis15] – Chap. 4<br>• How to write effective use cases, VisualParadigm documentation<br>• Jacobson, I., Spence, I., & Kerr, B. (2016). Use-case 2.0. Communications of the ACM, 59(5), 61–69. | |