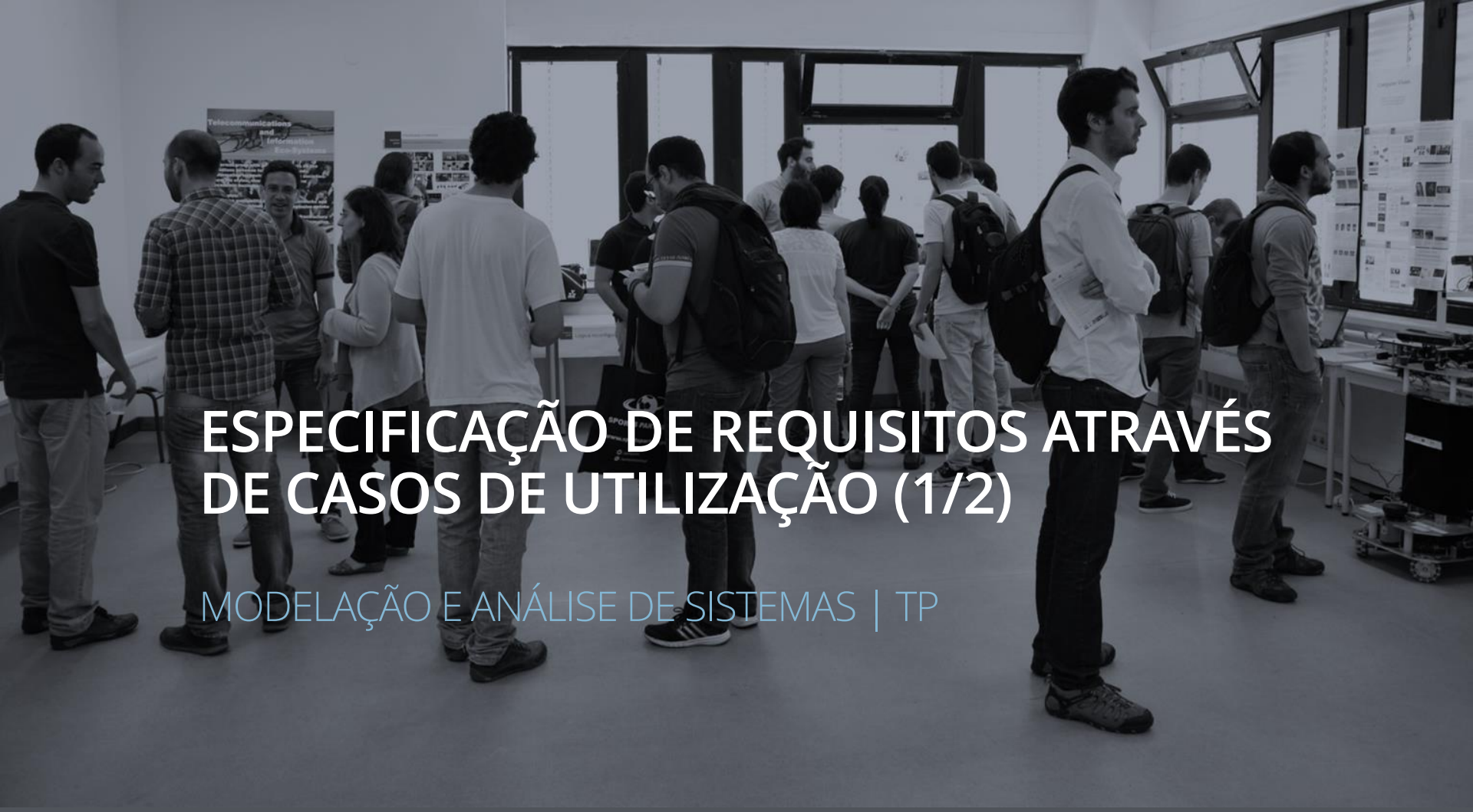


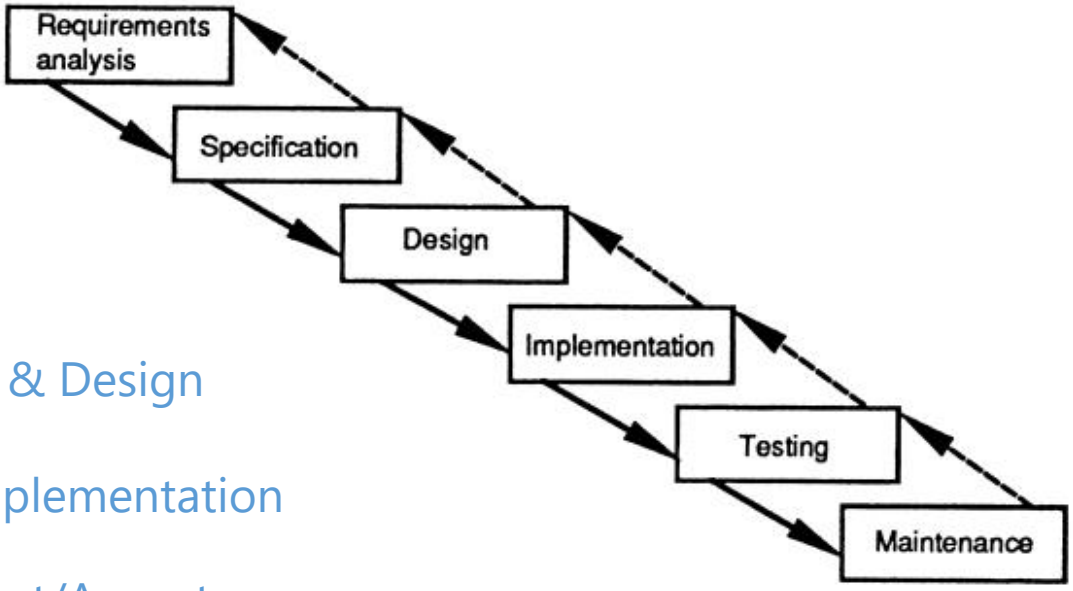
ESPECIFICAÇÃO DE REQUISITOS ATRAVÉS DE CASOS DE UTILIZAÇÃO (1/2)

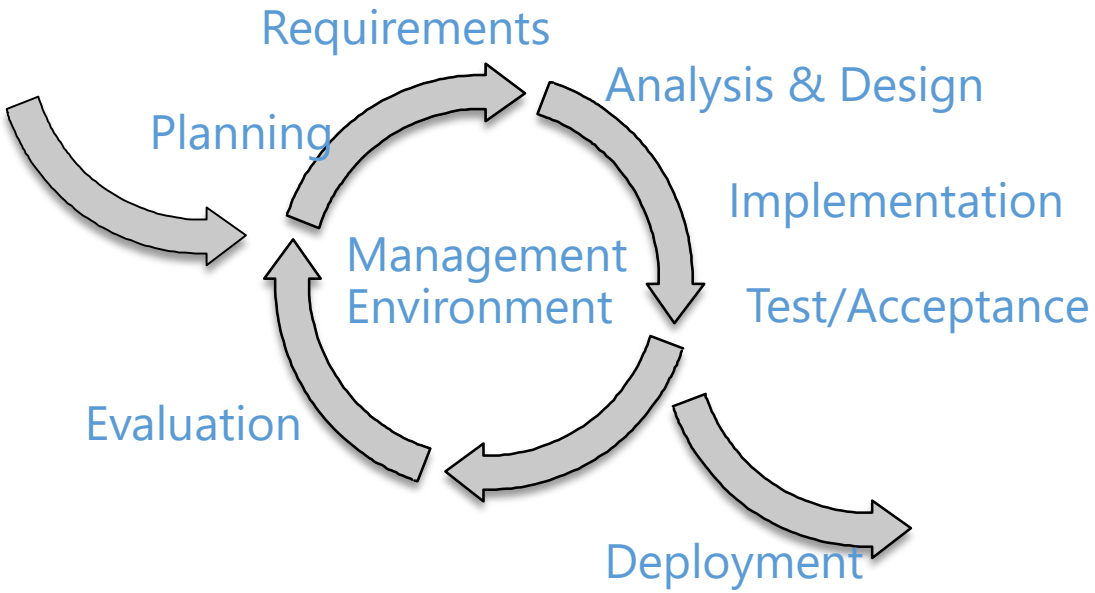
MODELAÇÃO E ANÁLISE DE SISTEMAS | TP

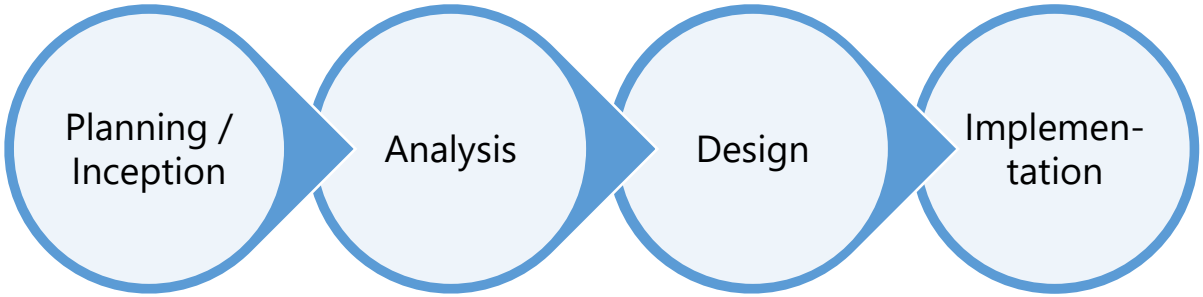
ILÍDIO OLIVEIRA ico@ua.pt
v2018-02-22

universidade de aveiro
departamento de eletrónica, telecomunicações e informática
deti

Modelos de processo







Requisito

Um requisito é uma afirmação sobre algo que o (novo) sistema deve fazer ou uma característica que deve ter.

Processo de **Definição de requisitos**

Esforço sistemático para tornar a explicação geral, de alto nível, das necessidades do negócio/organização numa lista de capacidades que possa ser usada como input para:

- → o resto da Análise (criar modelos estruturais e de comportamento)
- → o desenho do sistema.

Requirements are a specification of what should be implemented. They are descriptions of how the system should behave, or of a system property or attribute.
They may be a constraint on the development process of the system.

Ian Sommerville and Pete Sawyer (1997)

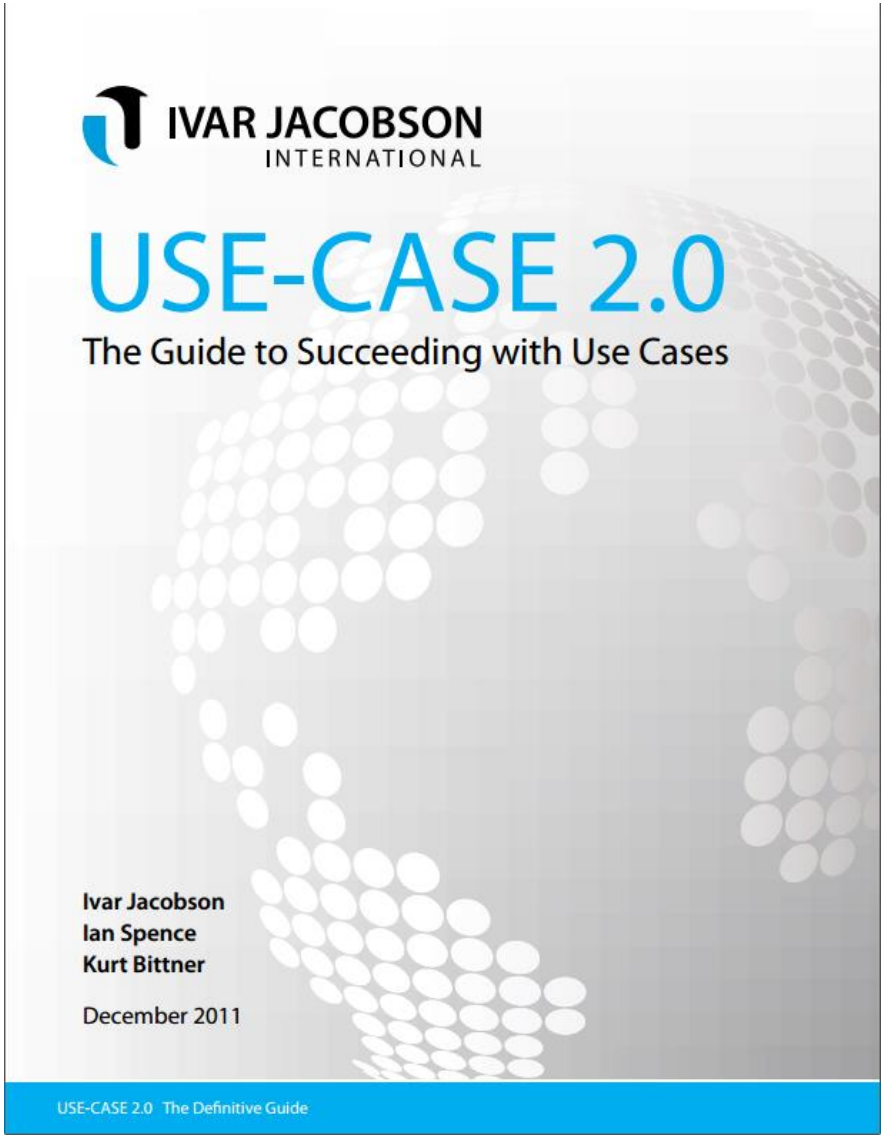
Várias estratégias para levantamento e documentação de requisitos

Instrumento proposto por Jacobson *et al*:

→ **Casos de utilização**

Conceito apresentado originalmente em:

Jacobson, I., 1993. *Object-oriented software engineering: a use case driven approach*. Pearson Education.



https://www.ivarjacobson.com/publications/white-papers/use-case-ebook

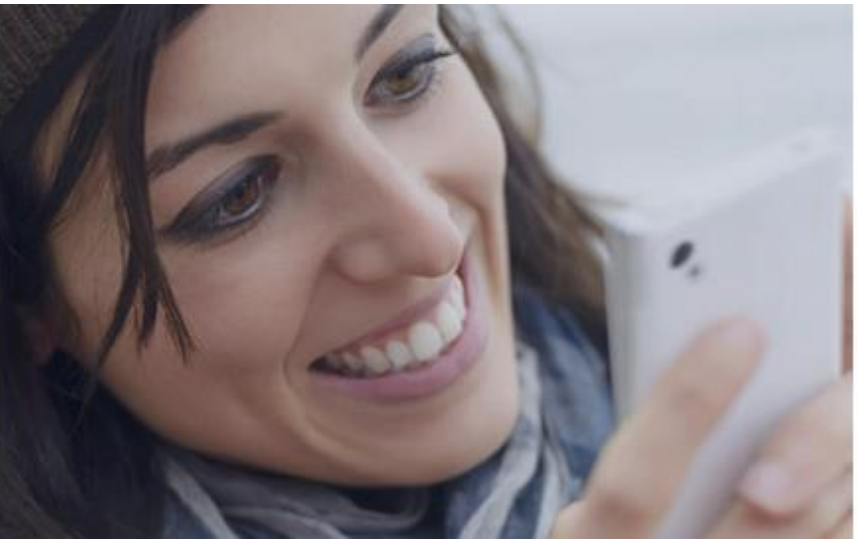
Caso de utilização (CaU)

Engloba uma sequência de ações que um sistema executa e que produzem um resultado com valor para algum ator em particular.

Implica:

→ **Foco no utilizador** do sistema e nos episódios de uso

→ Foco na compreensão daquilo que os atores consideram um **resultado de valor** (motivações para usar um sistema)



- 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.

Contas Cartões Crédito Poupança e Investimento Imóveis Seguros À sua Medida

Eu quero...

Ser cliente da Caixa

Comprar uma casa

Comprar um Carro

Viajar

Preparar o Futuro dos meus Filhos

Poupar para o Futuro

Preparar a minha Reforma

Proteger a minha Família

→ .


Spotify®

O caso de utilização capta *quem* (ator) faz *o quê* (interação) com o sistema, *com que fim* (objetivo)

Sequência de ações que um sistema executa e que produz um resultado com valor para algum ator em particular (incluindo as variantes relacionadas)

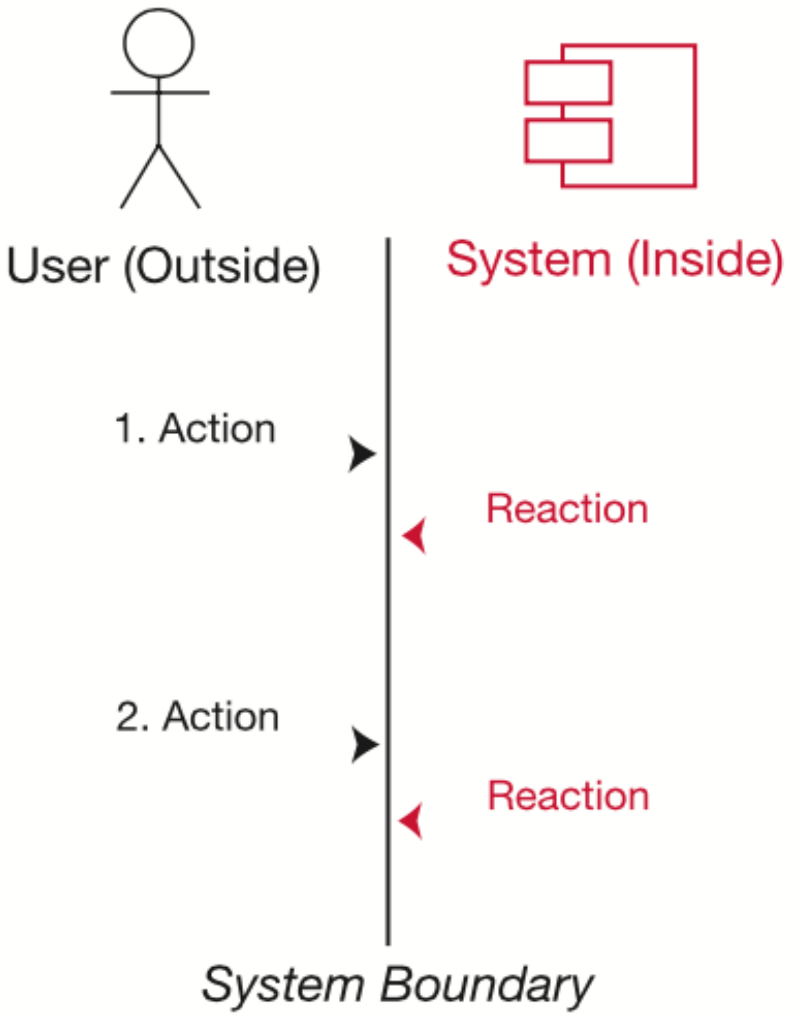
A menor porção de atividade (interação) que produz um resultado relevante para o utilizador

O contexto para um conjunto de requisitos relacionados.

O CaU descreve **um diálogo entre o ator e o sistema**

Narrativas para contar como é que "alguém" usa um sistema

1. **"**um Cliente chega a uma caixa com artigos para comprar.
2. O Operador passa cada artigo no leitor de código de barras para registo.
3. O sistema apresenta o total provisório e a lista de artigos incluídos.
4. O Operador termina a venda e indica o tipo de pagamento.
5. O cliente introduz a informação de pagamento.
6. O sistema valida o pagamento, atualiza o stock e imprime o recibo.
7. O cliente leva o recibo e os artigos."



O CaU tem um fluxo

Um fluxo-tipo (o cenário “normal”)

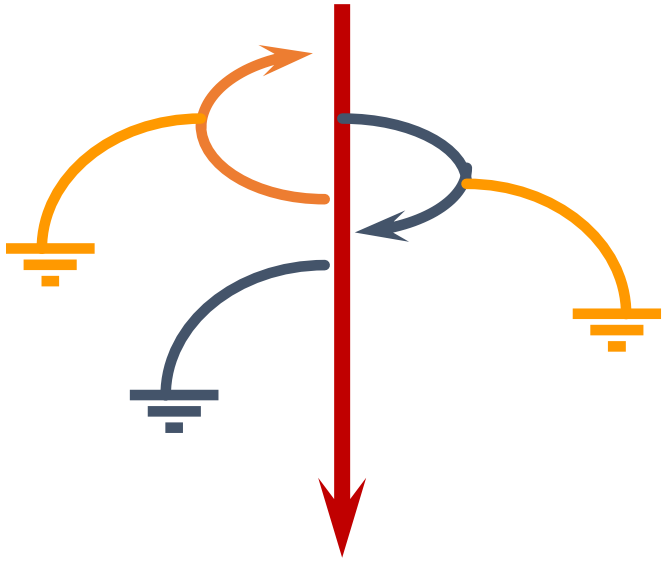
Guião para o diálogo ator/sistema

Vários fluxos alternativos

Variações “normais”

Casos particulares/incomuns

Situações de erro e o seu tratamento



Um cenário é uma das formas de percorrer um CaU → uma instância de um CaU

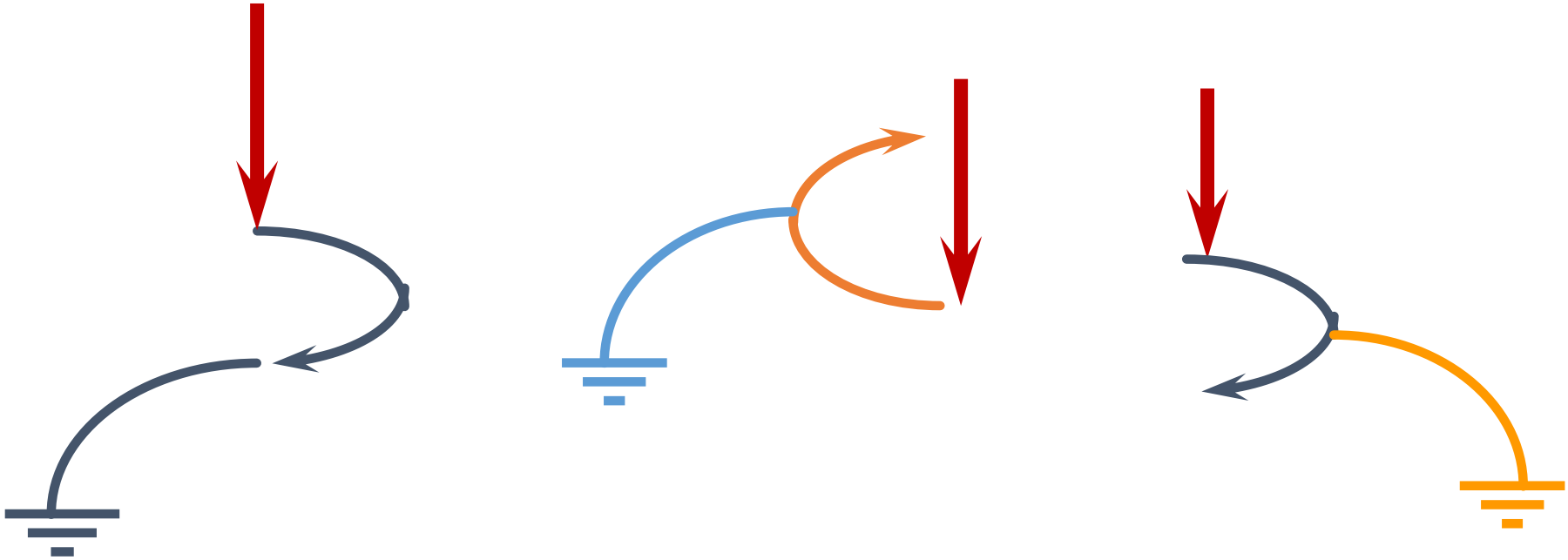
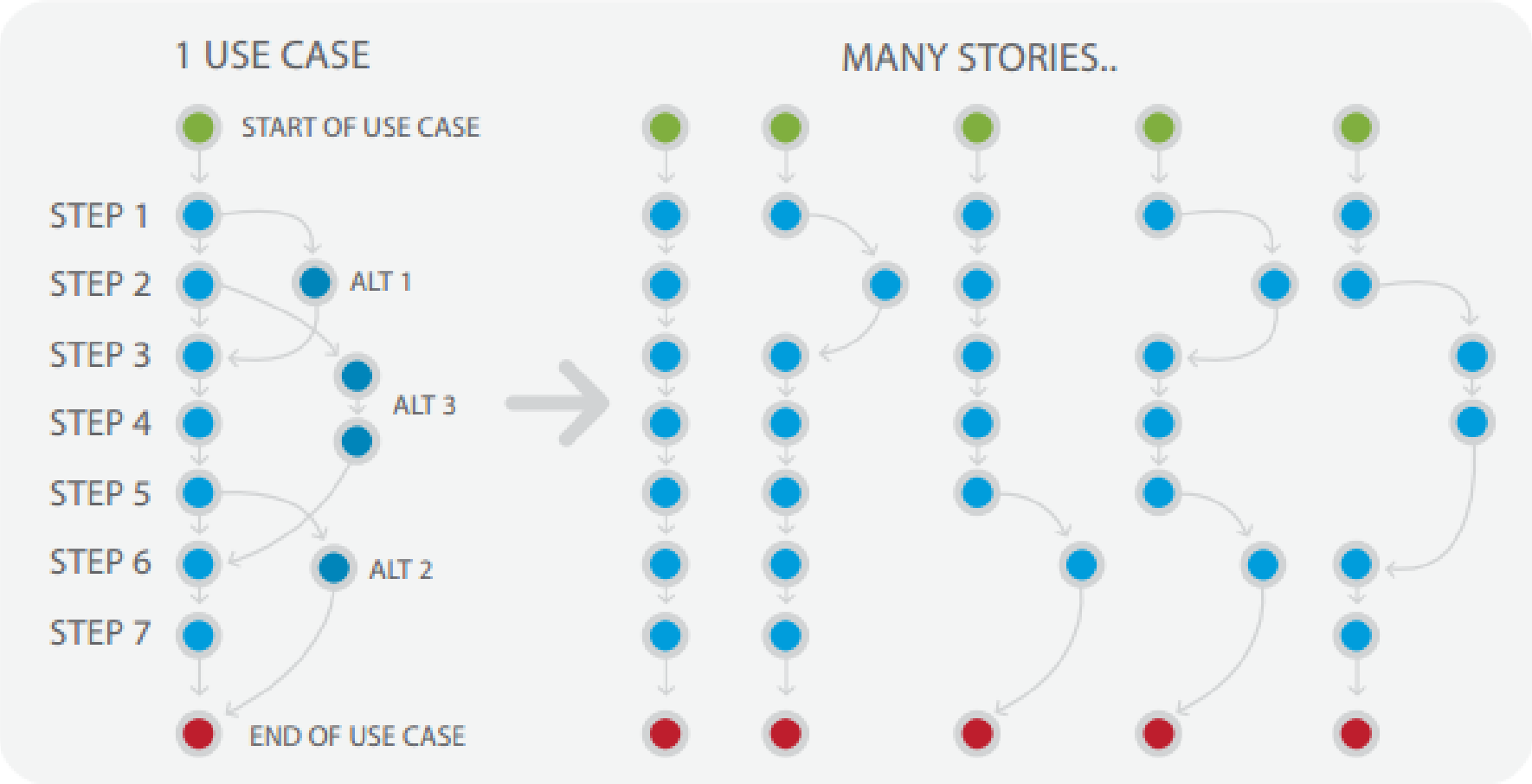


Figure One use case has different flows



Modelo de casos de utilização

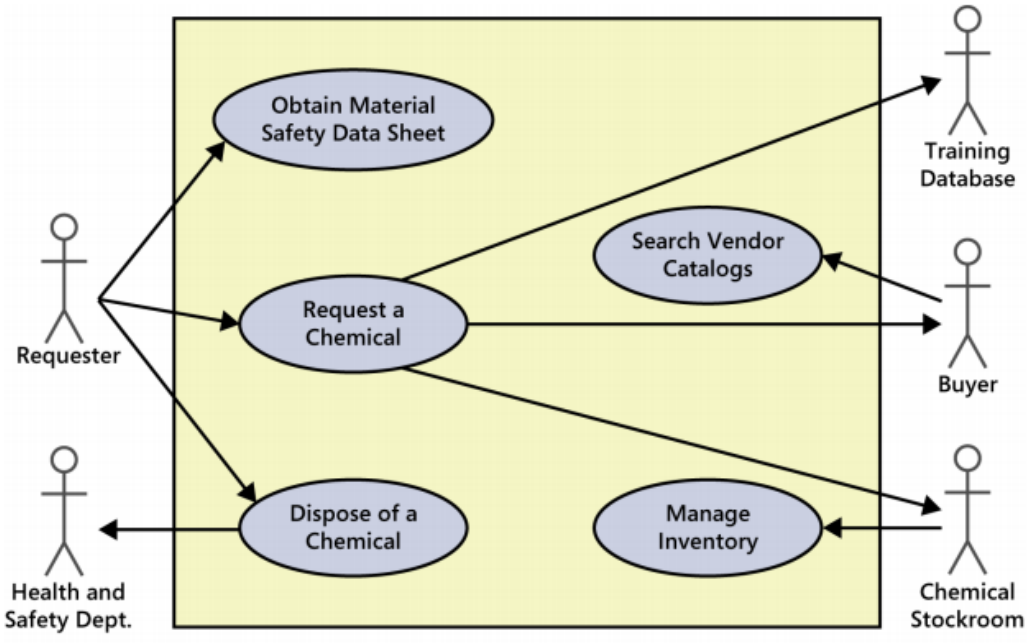
Os intervenientes que interagem com o sistema e contribuem para a realização dos objetivos são modelados como **atores .**

As formas como o sistema será usado para atingir esses objetivos são modelados como **casos de utilização**.

O **modelo de casos de utilização** fornece o contexto para a descoberta, partilhada e compreensão dos requisitos do sistema.

Um modelo de caso de utilização é um modelo de todas as maneiras úteis de usar um sistema. Isso permite apreender rapidamente o **âmbito do sistema** – o que é incluído e o que não é – e dar a equipa uma visão global de que o sistema vai fazer.

A visão gera é conseguida sem nos perdermos nos detalhes dos requisitos ou a parte interna do sistema.



Elementos do modelo de CaU

Ator

Qualquer entidade (o papel de alguém, outro sistema,...) externa ao sistema sob especificação, que interage com este

Cenário

Uma situação/história particular de utilização do sistema, i.e., um caminho possível na execução de um caso de utilização

E.g.: pagamento da compra com dinheiro, falha no pagamento por falta de autorização do cartão

Caso de utilização

Conjunto de cenários relacionados com o mesmo objetivo

Um episódio de utilização + variantes.

Associações

Relacionamento Actores/CaU, CaU/CaU, Actores/Actores.

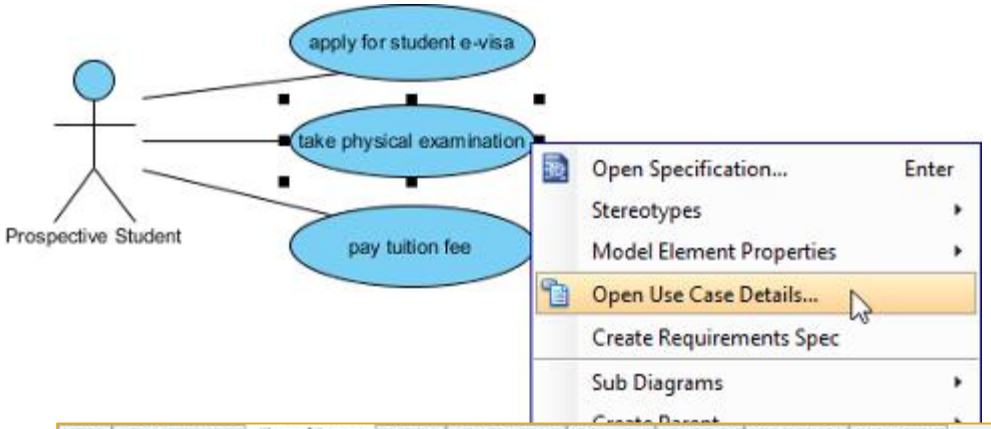
Documentação dos **cenários com narrativas**

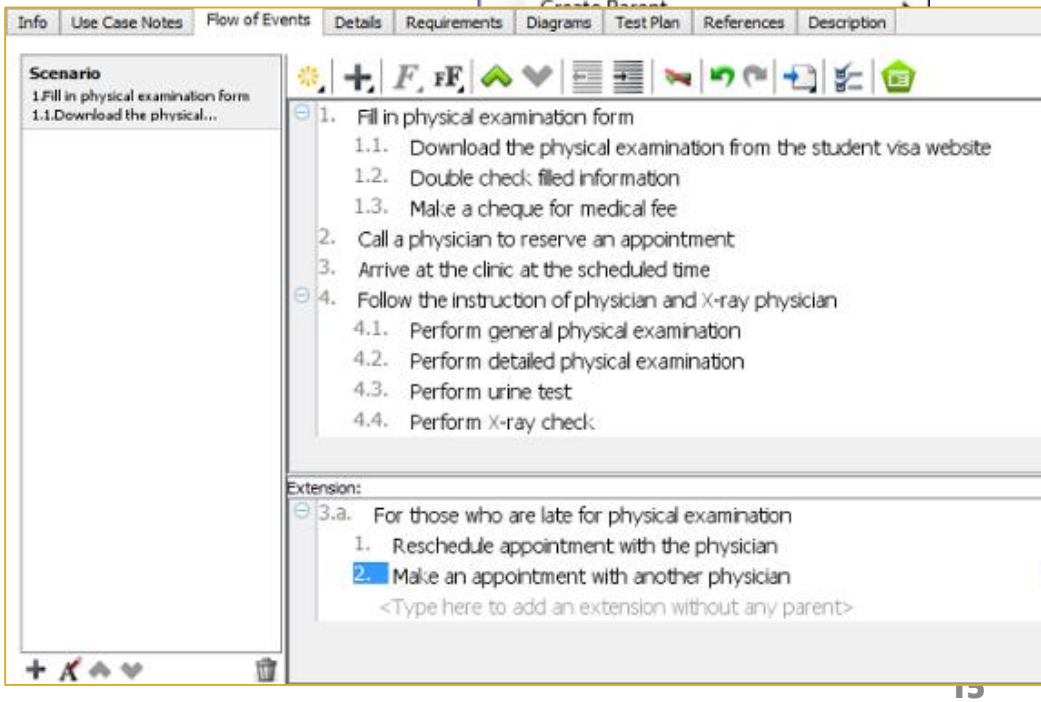
O propósito de uma narrativa de CaU é contar a história **como o systema e os seus atores trabalham em conjunto** para atingir um determinado objetivo.

As narrativas:

- → Delineiam as **histórias** usadas para explorar os requisitos e identificar cenários.
- → Descrevem uma **sequência de ações**, incluindo as **variantes**, que os atores e um Sistema podem executar para alcançar um objetivo
- → São apresentados com um conjunto de **fluxos** que descrevem como um ator usa um Sistema para atingir um objetivo e o que é que o sistema faz (responsabilidades) para ajudar nisso.
- → Capta/organiza as informações sobre os requisitos necessárias para apoiar as atividades de desenvolvimento.

As narrativas podem ser apresentadas de muitas maneiras, incluindo Wiki, pequenas "fichas", relatórios no MS Word, ou inseridas em ferramentas de gestão de projeto ou de modelação (e.g.: VisualParadigm).





Um caso de utilização conta a história de utilização

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.)

| Oliveira (2017)

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.*

A narrativa segue uma estratégia

Descrever como é que o CaU começa e acaba.

Descrever o fluxo de eventos que provocam as ações/reações

Forçar o início com “Começa quando o <ACTOR> <EVENTO>”

Descrever apenas o cenário “dentro” deste CaU

Não incluir fluxo de outros, nem comportamento externo ao sistema

Clarificar a troca de informação entre Atores e Sistema

Evitar designação vagas, ambíguas

E.g.: “E outros assim…”, “algo demorado…”

Verificação de qualidade:

este texto pode ser usado para os testes (de aceitação)? Posto de outra forma: é claro, completo, específico?

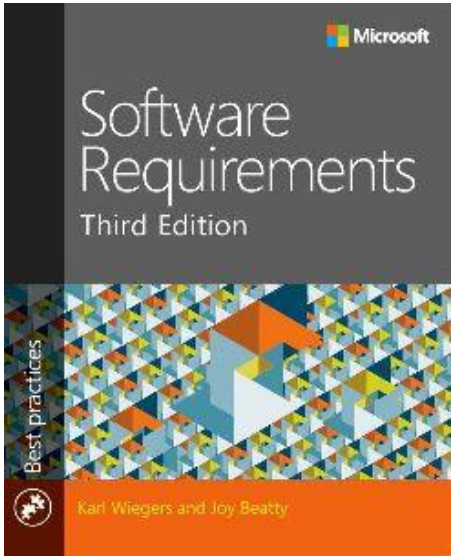
Use case:	Brief description:
Create new assigment	The Teaching Staff creates a new Activity of type Assignment, directly inserting it in the page layout. The assignment must define a title and a time period for submissions and can be configured to work with individual or group submissions. The assignment 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. 2. Select desired course. 3. Turn editing mode on. 4. Add Assignment activity in the page layout. 5. Configure Assignment activity. 6. Commit changes.
Alternative flows:	Step 1: IdP unavailable. 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? Step 5. The browser does not accept the rich text editor. Default to plain text?

→ mais detalhes

Essential elements of a use case

- → 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



→ Wiegers 2013

Figure "Request a Chemical" use case

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. PRE-2. User is authorized to request chemicals. PRE-3. Chemical inventory database is online.		
Postconditions:	POST-1. Request is stored in the CTS. POST-2. Request was sent to the Chemical Stockroom or to a Buyer.		
Normal Flow:	**4.0 Request a Chemical from the Chemical Stockroom** 1. Requester specifies the desired chemical. 2. System lists containers of the desired chemical that are in the chemical stockroom, if any. 3. System gives Requester the option to View Container History for any container. 4. Requester selects a specific container or asks to place a vendor order (see 4.1). 5. Requester enters other information to complete the request. 6. System stores the request and notifies the Chemical Stockroom.		
Alternative Flows:	**4.1 Request a Chemical from a Vendor** 1. Requester searches vendor catalogs for the chemical (see 4.1.E1). 2. System displays a list of vendors for the chemical with available container sizes, grades, and prices. 3. Requester selects a vendor, container size, grade, and number of containers. 4. Requester enters other information to complete the request. 5. System stores the request and notifies the Buyer.		
Exceptions:	**4.1.E1 Chemical Is Not Commercially Available** 1. System displays message: No vendors for that chemical.		

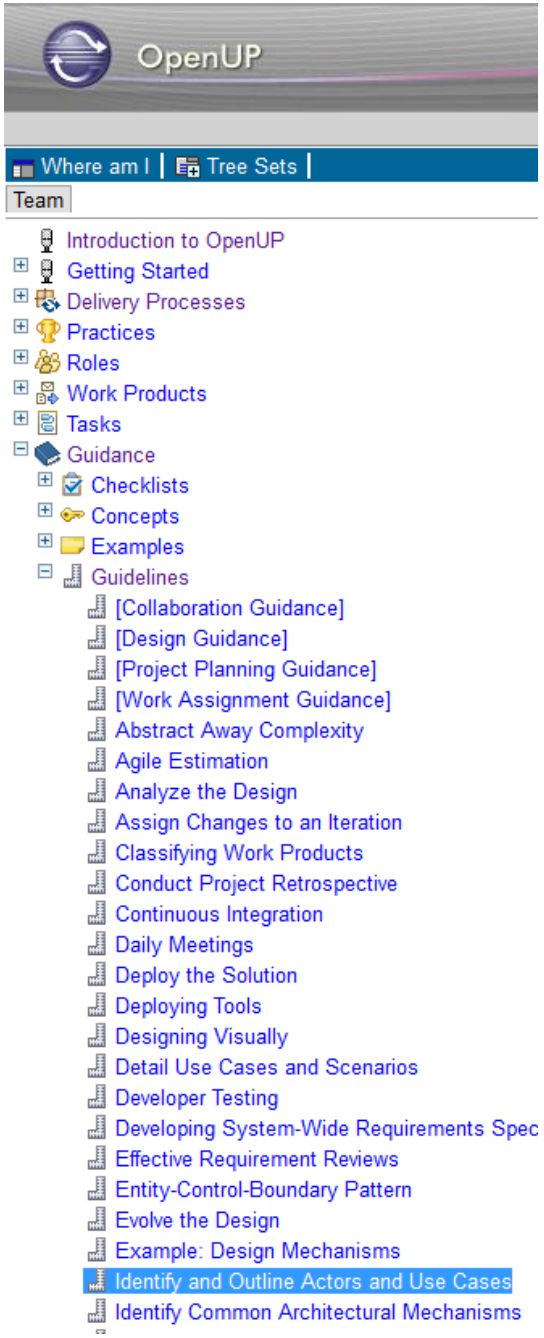
→ Wiegers 2013


espaço moodle
da universidade
de aveiro

Como encontrar os CaU?

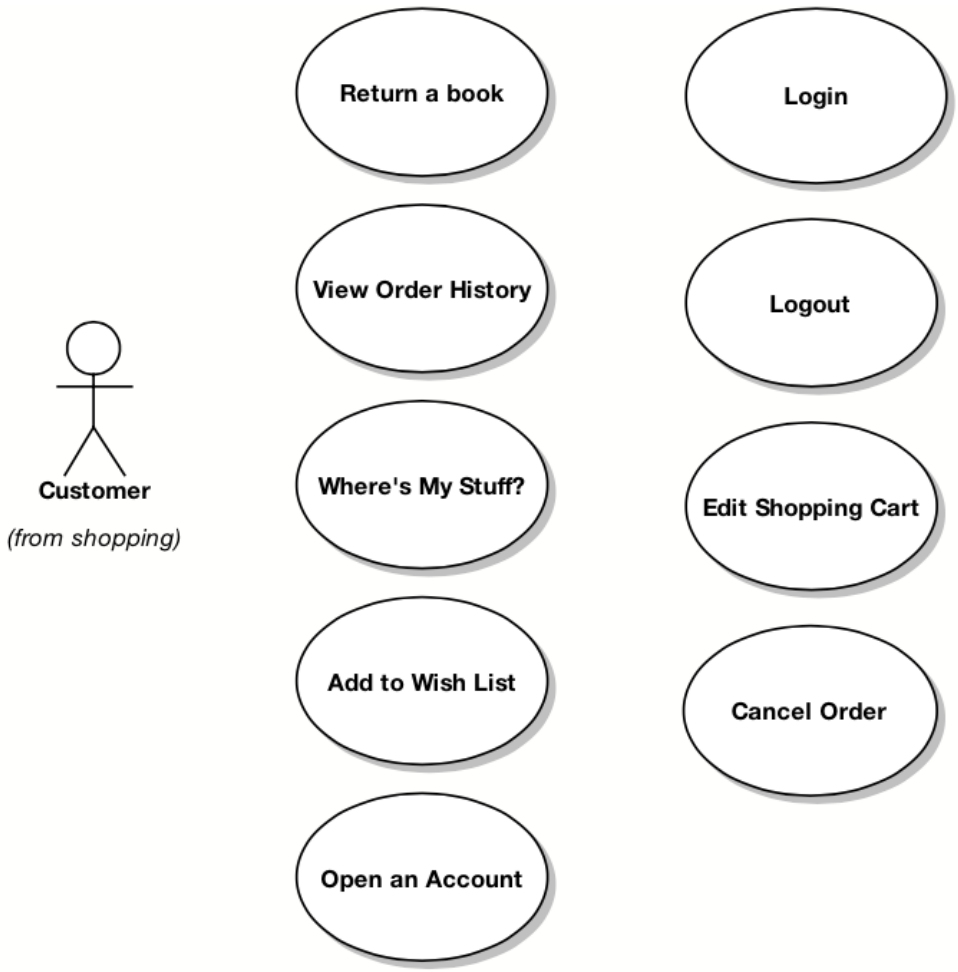
1. Identificar a fronteira do sistema
2. Identificar os atores que, de alguma forma, interagem com o sistema
3. Para cada ator, identificar os objetivos/motivações para usar o sistema
4. Definir os CaU que satisfazem os objetivos dos atores

Dar nomes que refletem a motivação do ator

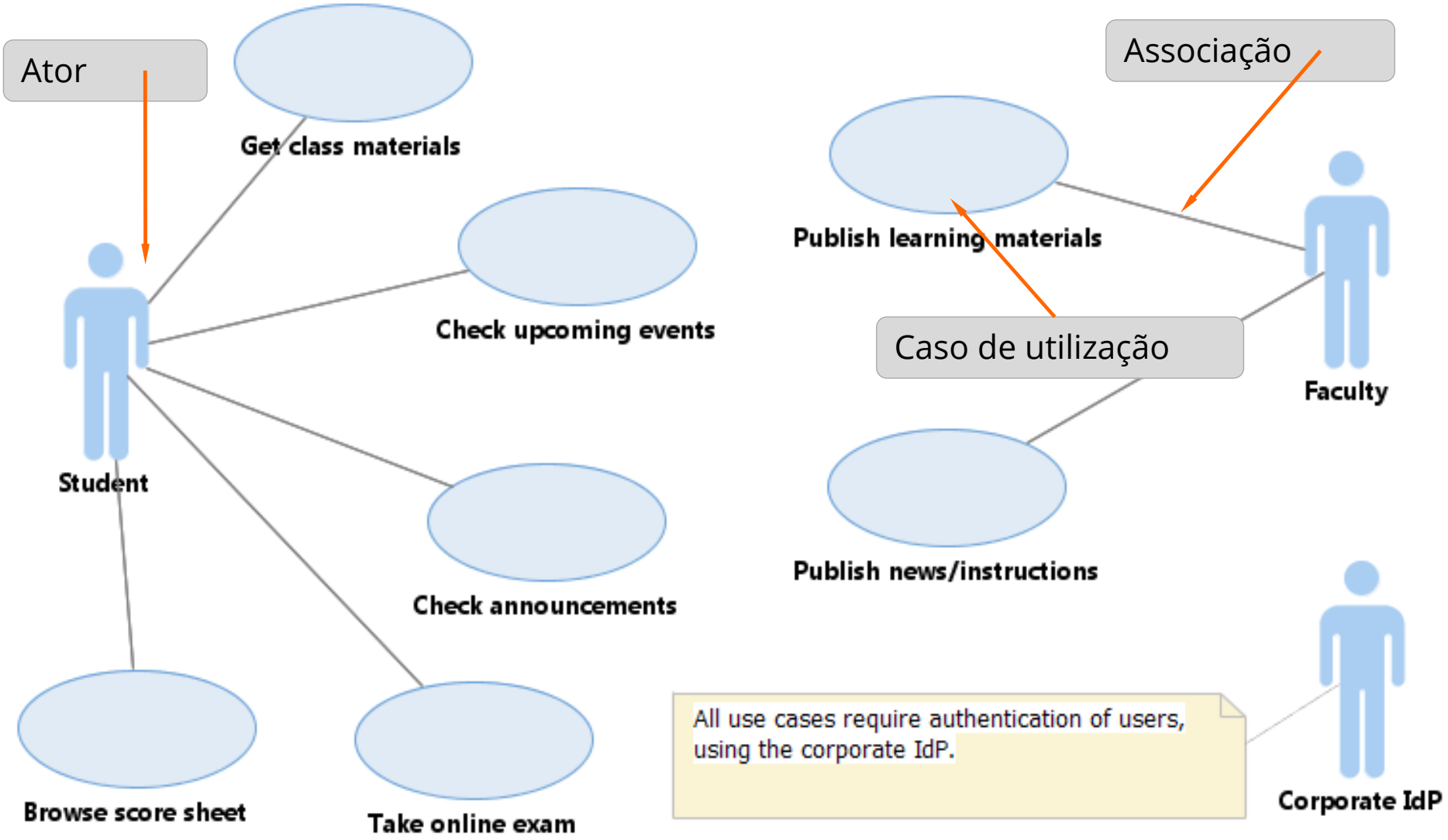


Guideline: Identify and Outline Actors and Use Cases

Visualização do modelo de casos de utilização com a UML



Elementos do Diagrama de Casos de Utilização

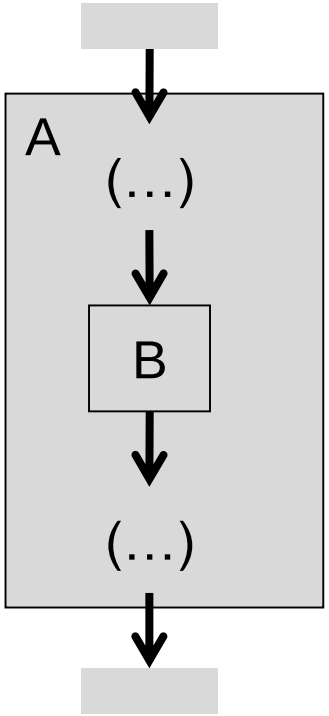


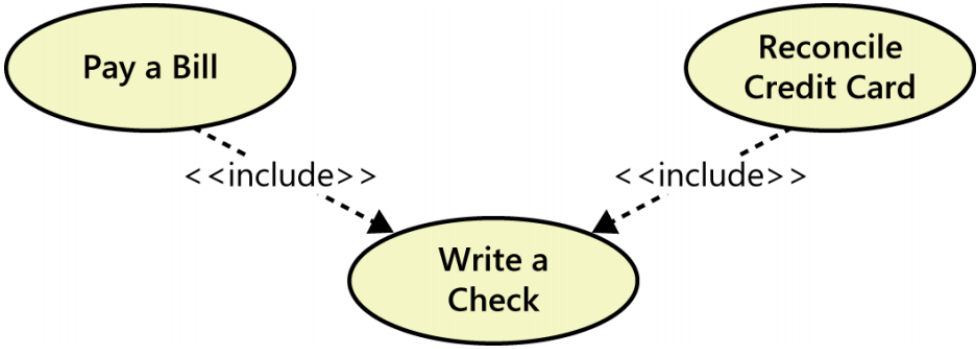
Reutilização de comportamento com **include**

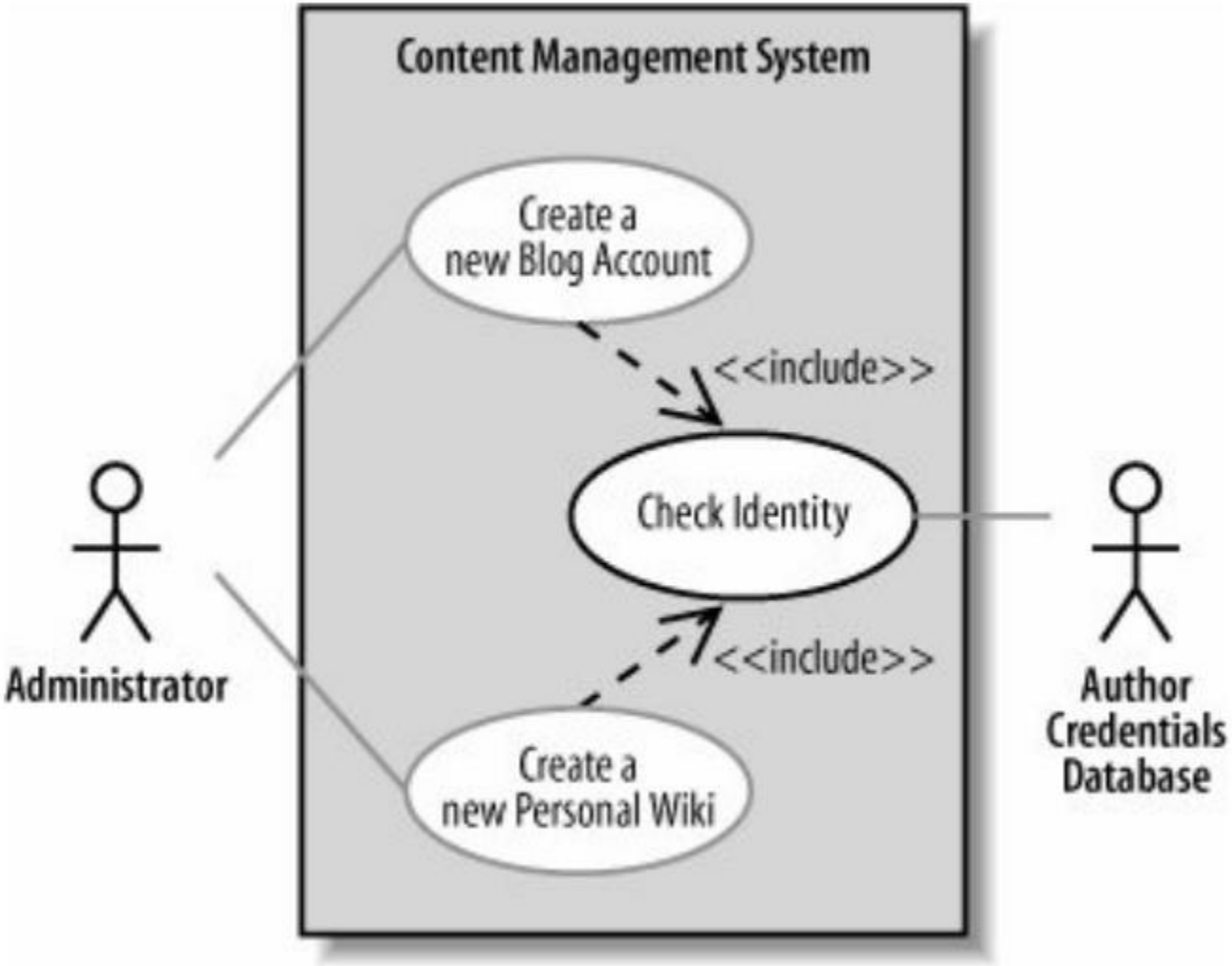
A inclui B

A comportamento (cenário) de A incorpora o comportamento em B

Facilita a “factorização” e reutilização





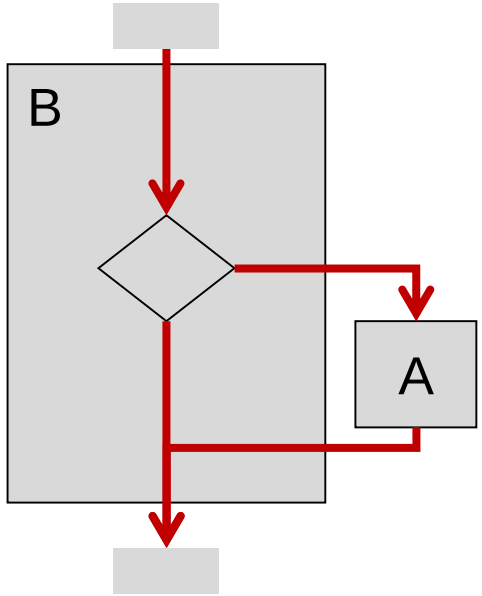

Content Management System
Create a new Blog Account
<<include>>
Check Identity
<<include>>
Create a new Personal Wiki
Administrator
Author Credentials Database

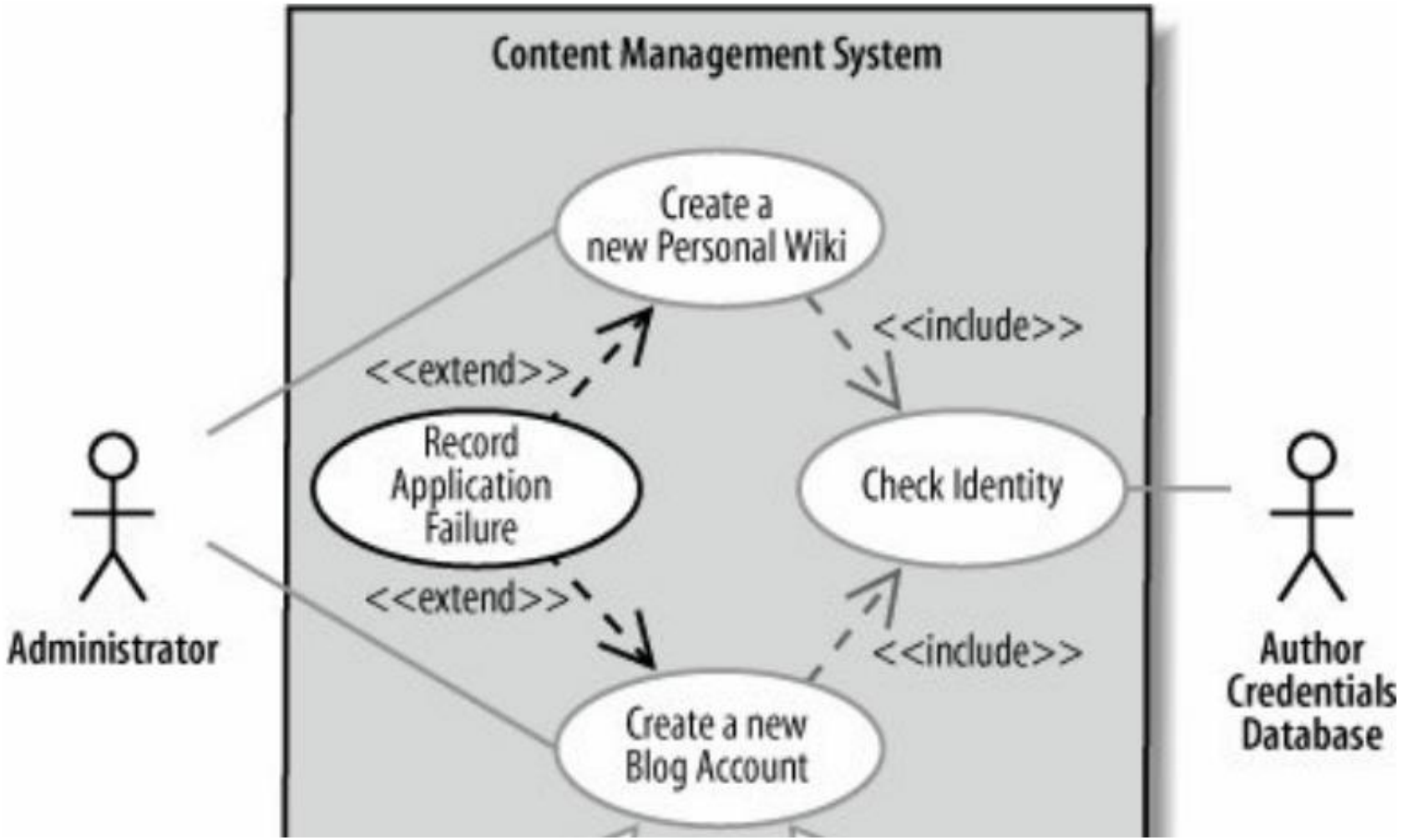
Ativação de comportamento opcional com **extend**

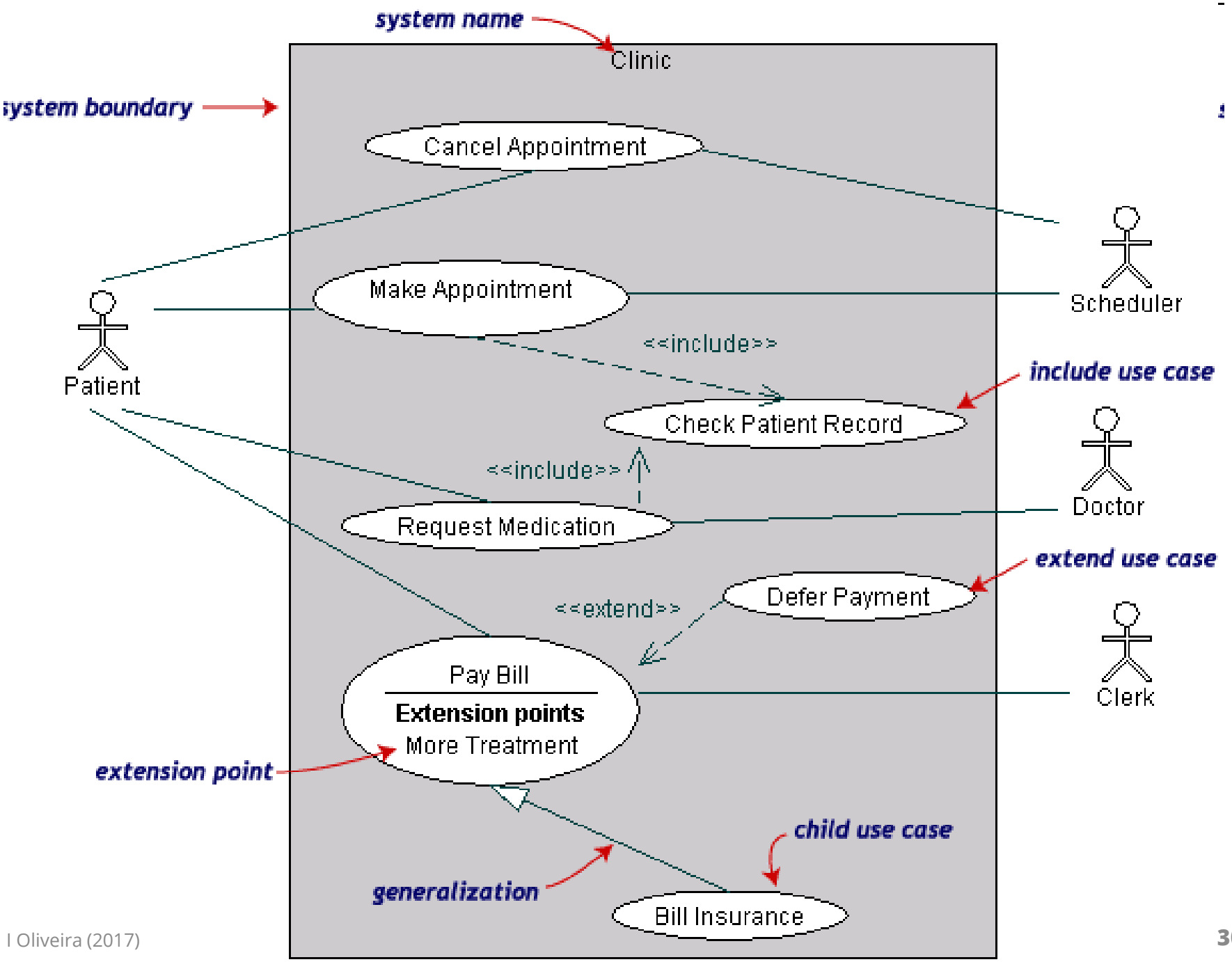
A estende **B**

O comportamento de B pode incorporar o comportamento em A, dependendo da verificação de uma "condição de extensão" (*extension point*)

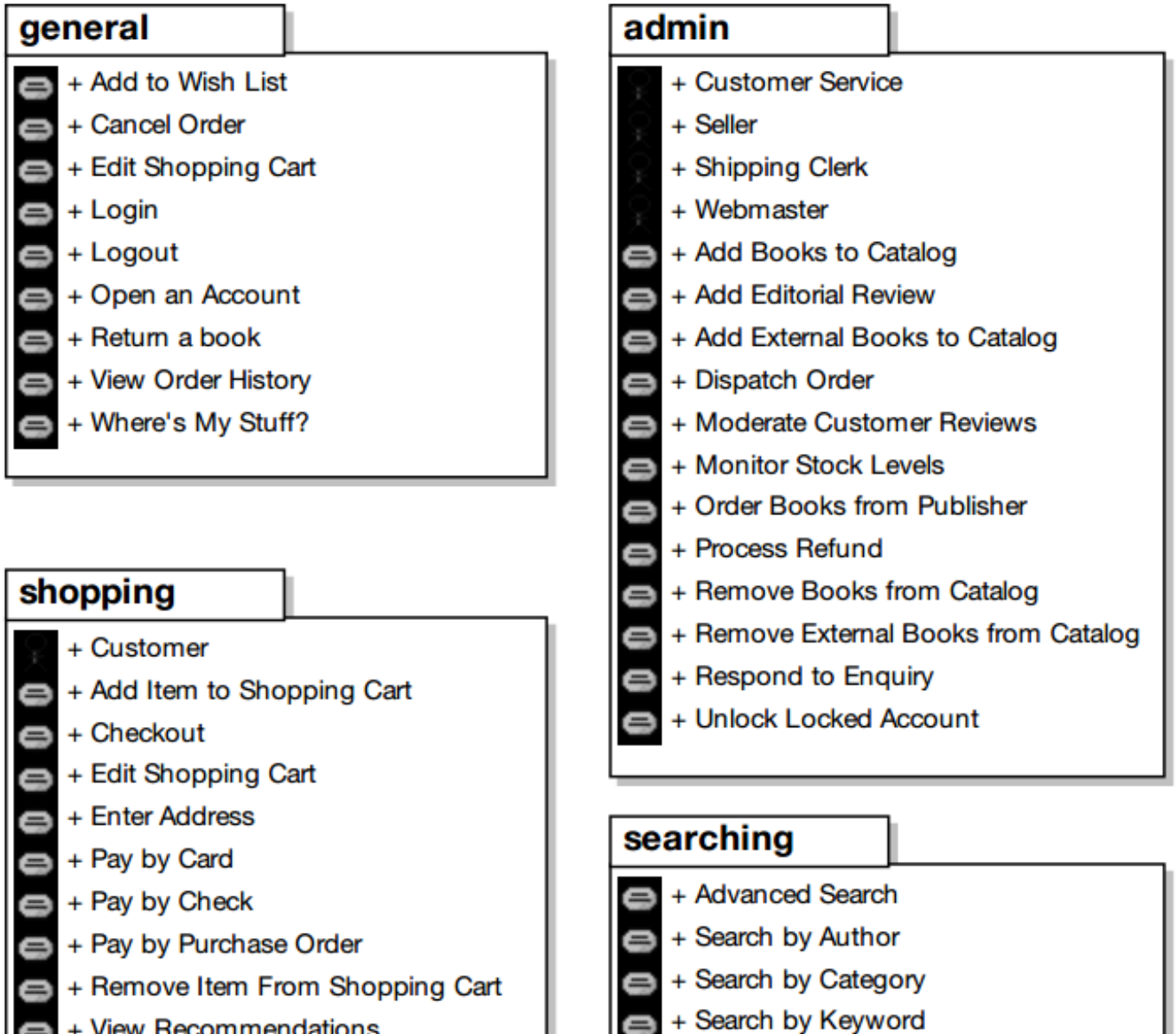
Modela a incorporação opcional/condicional




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


system name
Clinic
system boundary
Cancel Appointment
Make Appointment
Check Patient Record
Request Medication
Defer Payment
Pay Bill
Extension points
More Treatment
Bill Insurance
Patient
Scheduler
Doctor
Clerk
<<include>>
<<include>>
<<extend>>
include use case
extend use case
extension point
child use case
generalization

Os casos de utilização podem ser **agrupados em pacotes**



Recapitulando: **para que serve o modelo de casos de utilização?**

Uma vista de um sistema que destaca o **comportamento observável, tal como é percecionado pelos utilizadores**

O modelo de casos de utilização **divide a funcionalidade do sistem em episódios** relevantes para os utilizadores/atores

Capta os probelmas/motivações que levam à utilização do sistema

O modelo capta os requisitos do sistema (“o quê”), não a implementação da solução (“o como”)

A UML fornece uma visualização, mas o mais importante é a narrativa!

IVAR JACOBSON INTERNATIONAL

TRAINING | INSIGHTS & PUBLICATIONS ▸ | CUSTOMERS | ABOUT US ▸ | CONTACT U

Search on Ivar Jacobson Internat

HOME SERVICES ▸ SCALED AGILE FRAMEWORK ▸ ESSENCE ▸ ESSENCE TOOLS & PRACTICES ▸ USE CASES ▸

6 PRINCIPLES FOR SUCCESS WITH USE CASE ADOPTION

HOW TO APPROACH APPLYING USE CASE DATA TO YOUR BUSINESS.

Since their inception some 30 years ago, use cases have been used to identify, organize, synthesize and clarify system requirements for organizations across the globe. In most recent years, they have been used in techniques such as user stories. Use-Case 2.0 is the new generation of use-case driven development – light, agile and lean – inspired by user stories, Scrum and Kanban.

Although they are much more agile and lean, they still embody the same popular values from the past while expanding to architecture, design, test, user experience, and also instrumental in business modeling and software reuse. But, for the adoption of use cases to be seamless, there should be a balance of principles applied.

https://www.ivarjacobson.com/publications/blog/6-principles-success-use-case-adoption

Principle: **Keep it simple by telling stories**

The use cases capture the goals of the system.

To understand a use case we tell stories. The stories cover how to successfully achieve the goal, and how to handle any problems that may occur on the way.

Use cases provide a way to identify and capture all the different but related stories in a simple but comprehensive way.

This enables the **system's requirements to be easily captured, shared and understood**.

Principle: **Goal-oriented**

As a use case is **focused on the achievement of a particular goal**, it provides a focus for the storytelling.

Rather than trying to describe the system in one go we can approach it use case by use case.

The results of the storytelling are captured and presented as part of the use-case narrative that accompanies each use case.

Principle: **Understand the big picture**

Form the big picture doesn't mean capturing all the requirements up front.

You just need to create something that sums up the desired system and lets you **understand scope and progress** at a system level.

A use-case diagram is a simple way of presenting an overview of a system's requirements see all the ways the system can be used, who is involved in an interaction.

Principle: **Focus on value**

Value is only generated if the system is actually used; so it is much better to focus on how the system will be used than on long lists of the functions or features.

To make the value easy to quantify, identify and deliver you need to structure the use-case narrative.

To keep things simple start with the simplest possible way to achieve the goal. Then capture any alternative ways of achieving the goal and how to handle any problems that might occur whilst trying to achieve the goal.

BASIC FLOW	ALTERNATIVE FLOWS
1. Insert Card	A1 Invalid Card
2. Validate Card	A2 Non-Standard Amount
3. Select Cash Withdrawal	A3 Receipt Required
4. Select Account	A4 Insufficient Funds in ATM
5. Confirm Availability of Funds	A5 Insufficient Funds in Acct
	A6 Would Cause Overdraft
6. Return Card	A7 Card Stuck
7. Dispense Cash	A8 Cash Left Behind
	etc..

FIGURE 2: THE STRUCTURE OF A USE-CASE NARRATIVE

Use case adoption

First you need to **find some actors and use cases** to help you to:

- → Agree on the goals of the system
- → Scope releases of the system.
- → Agree on the value the system provides.
- → Identify ways of using and testing the system.

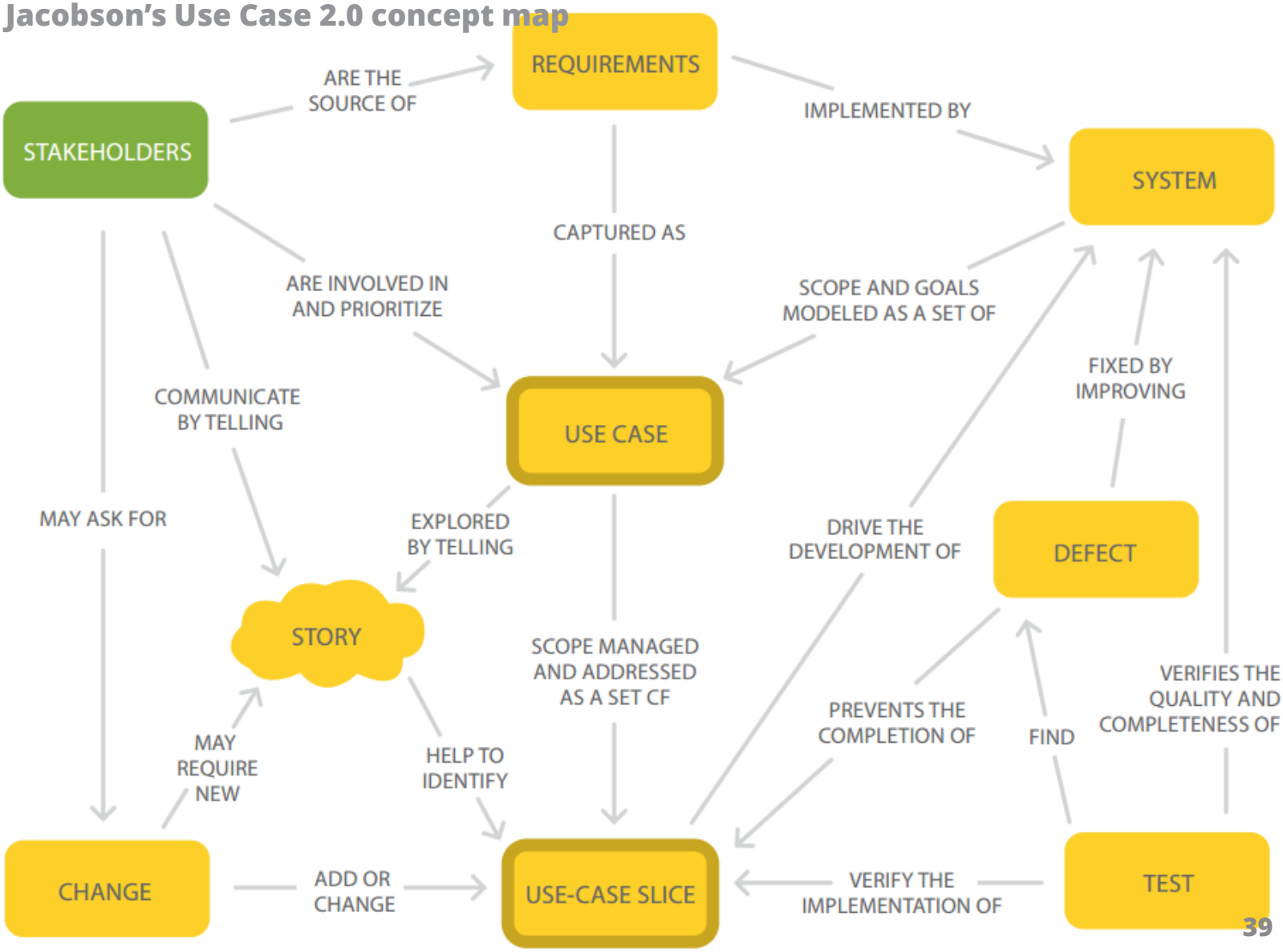
The best way to do this is to hold a **use-case modeling workshop** with your stakeholders.

There is no need to find all the system's use cases, just focus on those that are going to provide the stakeholders with the value they are looking for**. Other actors and use cases will be found** as you inspect and adapt the use cases.

As the use cases are discovered they should be **ordered/prioritized** to support the team's release plans.

One of the great things about use cases is that they enable high-level scope management without the need to discover or author all the stories.

Figure **Jacobson's Use Case 2.0 concept map**



I Oliveira (2017)

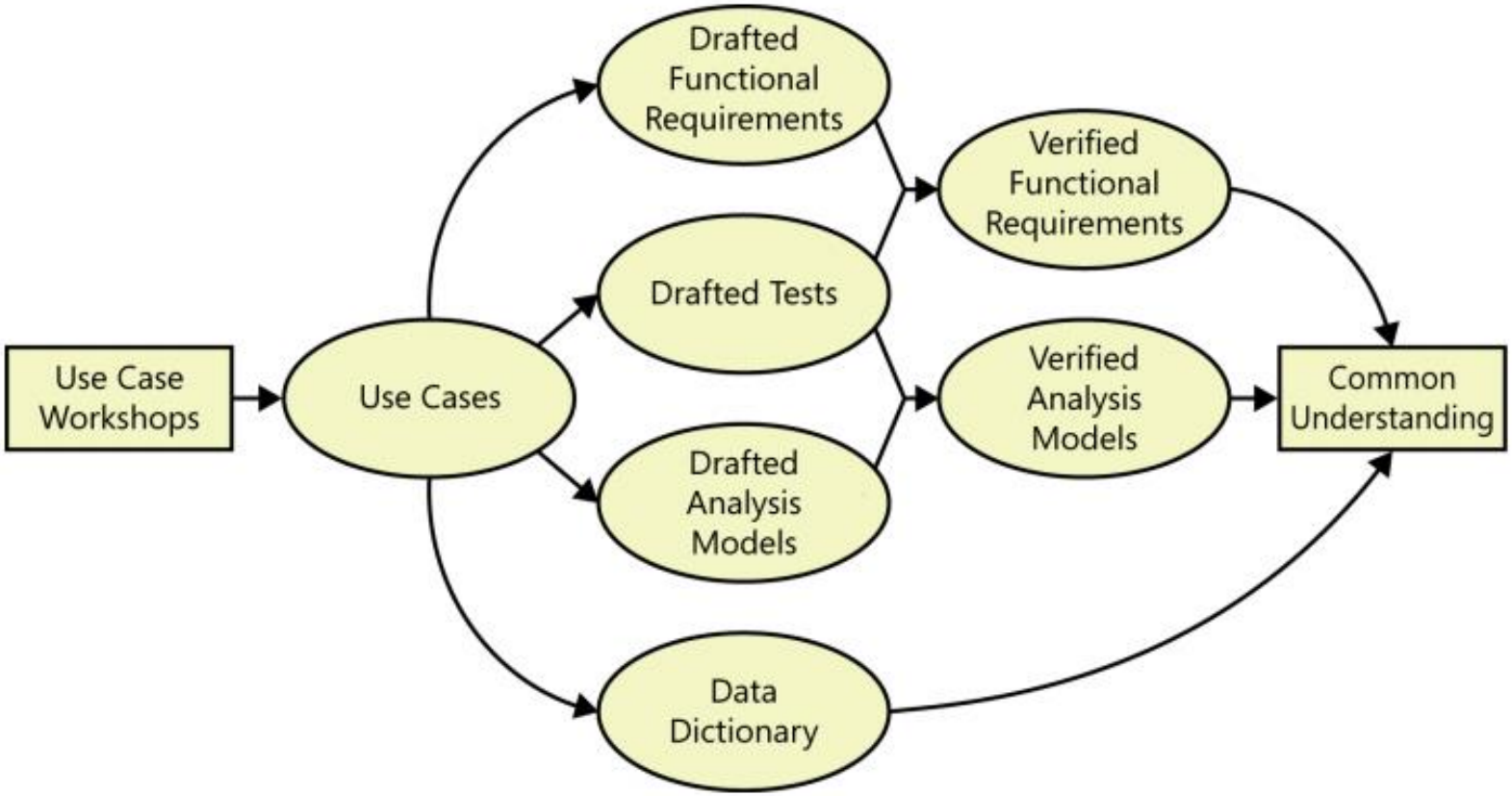


FIGURE 8-8 Use case elicitation work products.

→ Wiegers 2013

Use cases can and should be used to **drive the development**

The use case approach is not just a very practical technique to capture requirements from a usage perspective or to design practical user experiences, but it impacts the whole development lifecycle.

The key use cases or to be more precise the key use case slices (a slice being a carefully selected part of a use case) assist systematically in finding the application architecture.

They drive the identification of components or other software elements in software design.

They are the elements that have to go through test – and truly support test-driven design.

Use Case 2.0:
The Hub of Software Development

Ivar Jacobson, Ian Spence, Brian Kerr

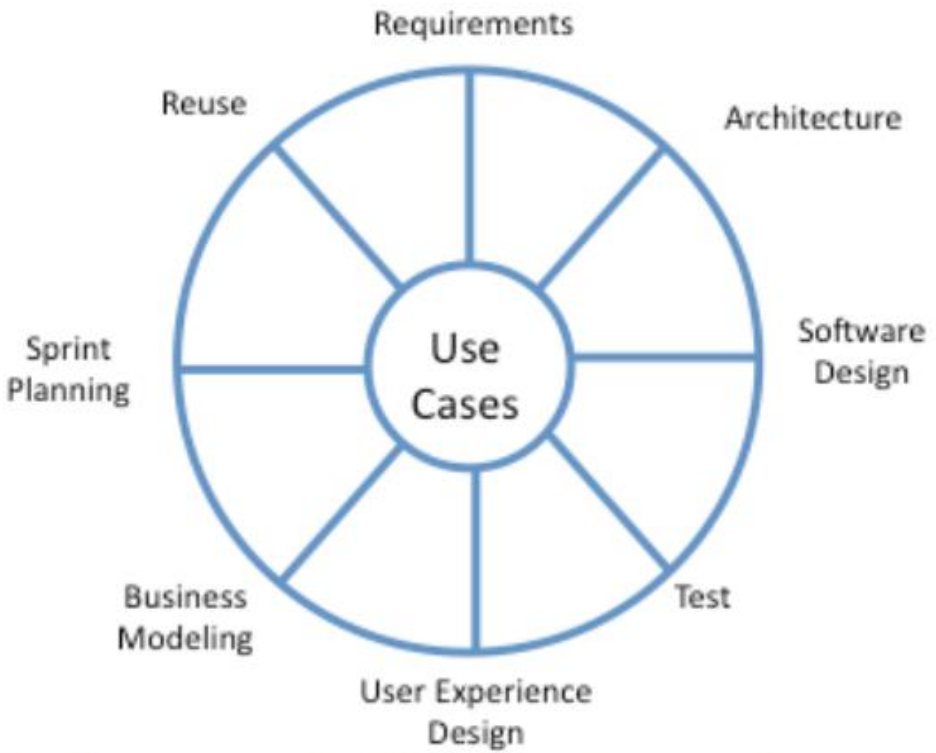
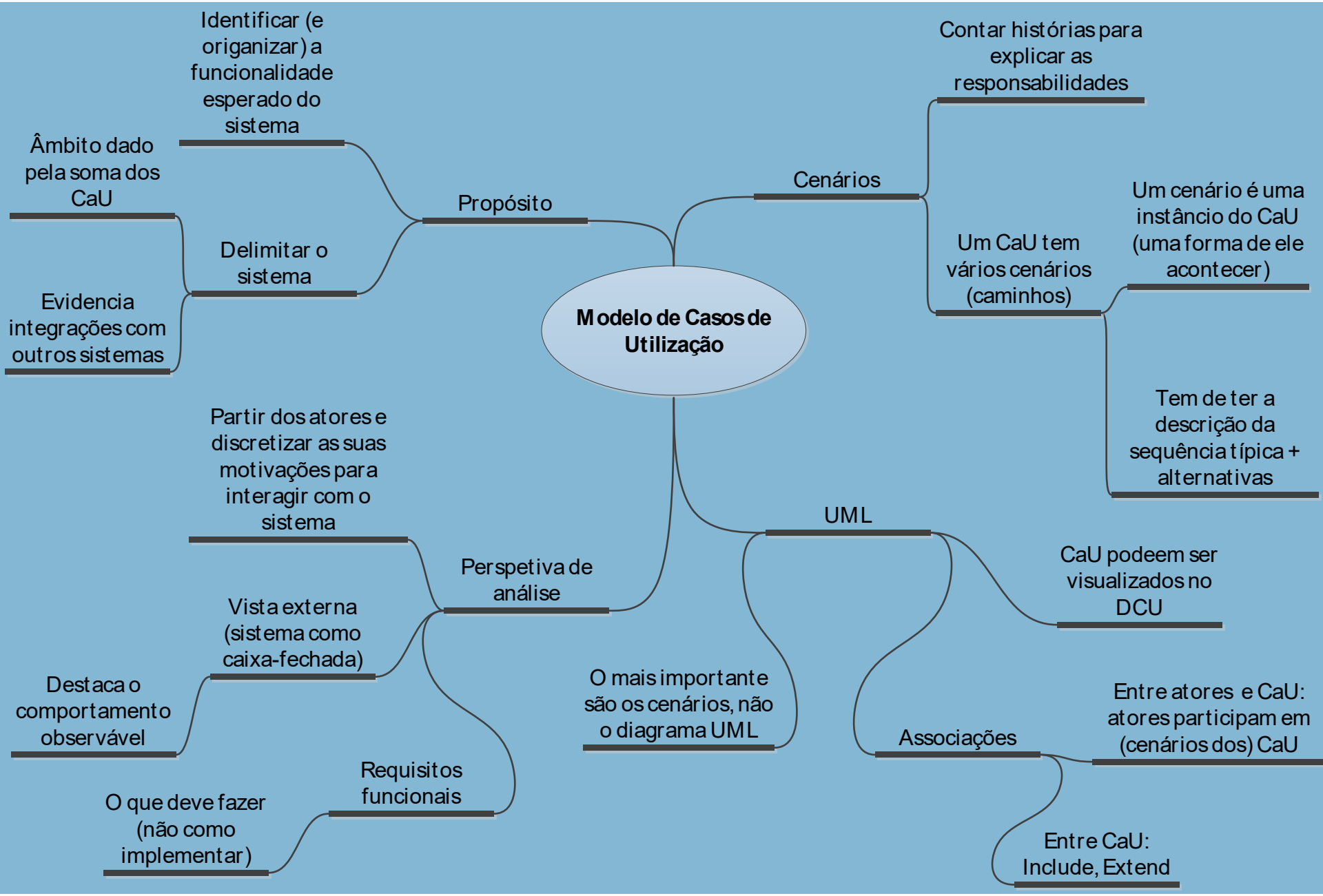


Figure 1: Use cases are the hub of the software development lifecycle.

https://www.ivarjacobson.com/publications/white-papers/use-case-hub-software-development



Mapa de conceitos