ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO

P.PORTO

CTeSP DWDM

Análise e Arquitetura de Sistemas

UML: Diagramas de Casos de utilização

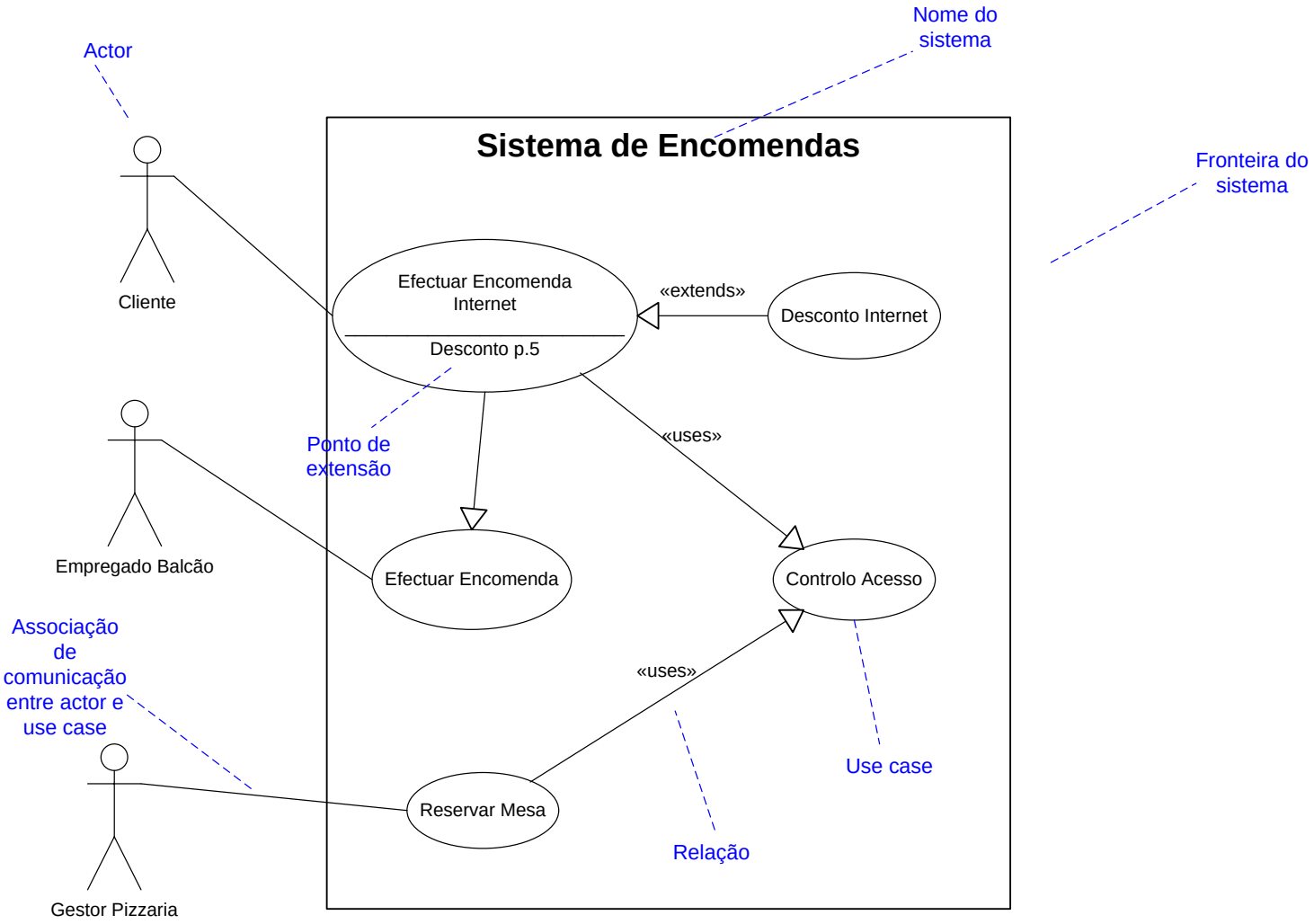
NORTE2020
PORTUGAL 2020
UNIÃO EUROPEIA
Fundo Social Europeu

Diagramas Use Cases

- Permitem:
  - mostrar **para que serve** o sistema (a utilidade do sistema), ignorando a forma como está organizado internamente
  - especificar o **contexto** do sistema
    - com quem interage (atores) e com que finalidade (casos de utilização)
  - capturar os **requisitos funcionais** do sistema
    - casos de utilização são funcionalidades do sistema vistas pelos utilizadores
- Podem referir-se a um sistema de software, um sistema de negócio ou organização, um equipamento, uma classe, etc.
- São elaborados por analistas e especialistas de domínio nos primeiros estágios do desenvolvimento

Exemplo: Sistema de Encomendas

- **O seguinte texto descreve um conjunto de requisitos para um novo sistema, que poderia ser obtido, por exemplo, em resultado de uma entrevista:**
- "Pretende-se desenvolver um sistema de informação de gestão para um grupo de pizzarias, PhonePizza, que permita aos clientes efetuar encomendas na loja e através da Internet. Na loja, o cliente dirige-se ao empregado de balcão que introduzirá no sistema a encomenda pretendida.
- Caso a encomenda seja efetuada através da Internet, o cliente terá que se identificar, através do seu nome de utilizador e palavra-chave (controlo de acesso). O cliente pode então registar os artigos que pretende encomendar, podendo usufruir de um desconto no item, caso esteja em promoção. O sistema deve ainda permitir que o gestor da Pizzaria efetue as reservas de mesa, verificando se este tem autorização para o efetuar."


Nome do sistema
Actor
Sistema de Encomendas
Fronteira do sistema
Cliente
Efectuar Encomenda Internet
Desconto p.5
«extends»
Desconto Internet
Ponto de extensão
«uses»
Empregado Balcão
Efectuar Encomenda
Controlo Acesso
Associação de comunicação entre actor e use case
«uses»
Use case
Reservar Mesa
Relação
Gestor Pizzaria

Atores

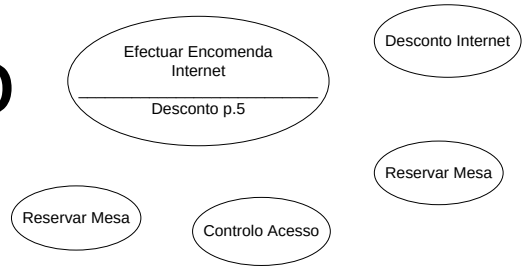
- **Ator = papel** (role)
  - Um ator em relação a um sistema é um papel que alguém ou alguma coisa do ambiente envolvente desempenha quando interage com o sistema
- **Ator = classe**
  - classes são frequentemente usadas para modelar papéis que objetos individuais podem desempenhar
- **Ator = tipo de utilizador** (em sentido lato)
  - pode ser uma pessoa ou outro sistema
  - pode utilizar ou ser utilizado, o que interessa é que interage com o sistema
- **Ator ≠ recurso do sistema**
  - recursos são pessoas, máquinas, etc. que pertencem ao sistema e que são usados para levar a cabo tarefas dentro do sistema
- **Ator ≠ indivíduo**
  - o mesmo indivíduo pode interagir com o sistema em vários papéis (como cliente, como fornecedor, etc.)

Atores

- Os atores devem ser caracterizados através de uma pequena descrição, de forma a assegurar uma correta compreensão do significado por todos os elementos da equipa envolvida na análise.

- Descrição dos atores do exemplo:
  - **Cliente** – uma pessoa que encomenda produtos da PhonePizza pela Internet e nas pizzarias.
  - **Empregado de balcão** – empregado que recebe as encomendas ao balcão da pizzaria.
  - **Gestor Pizzaria** – empregado que está encarregue de efetuar as reservas de mesa numa pizzaria.

Casos de Utilização



- Um caso de utilização é :
  - uma funcionalidade do sistema vista pelos utilizadores
  - um tipo de interação (de alto nível) entre atores e o sistema

- Identificação, para cada ator, dos use cases em que interage com o sistema:

Ator	Use Cases
Cliente	• Efetuar Encomenda Internet • Controlo de Acesso
Empregado Balcão	• Efetuar Encomenda • Controlo de Acesso
Gestor Pizzaria	• Reservar Mesa • Controlo de Acesso

Comunicação entre atores e use cases

- A comunicação entre um ator e os use cases pode ser representada por uma simples linha reta ou uma seta cujas pontas indicam a direção da comunicação.

  - **Linha reta simples** – os atores podem estar colocados em qualquer ponto do diagrama, como o pressuposto que existirá alguma comunicação de emissão ou receção.

  - **Seta unidirecional** – a seta indica o sentido preferencial da comunicação. Normalmente, neste caso é habitual a colocação dos atores emissores à esquerda da fronteira do sistema, e dos atores recetores à direita.

Casos de utilização e cenários

- **Cenário** – é uma determinada sequência de ações que ilustra um comportamento do sistema.
- Numa definição mais abstrata, *deve-se entender um cenário como uma instância de um caso de utilização, sendo normal que um caso de utilização possa ser descrito por dezenas de possíveis cenários.*

- Esta especificação deve incluir:
  - Como e quando o caso de utilização **começa e termina**
  - Quando é que o caso de utilização **interatua** com os atores
  - Que **objetos são trocados**

- **Cenário principal:** cenário onde não surgem problemas, pressupõe-se que estão reunidas todas as condições que garantem que tudo corre bem
- **Cenários alternativos ou secundários:** quando se pensa no que poderá correr mal no cenário
  - ex: situações de exceção

Casos de utilização e cenários

- A descrição pode assumir a forma de **texto livre** ou **estruturado** segundo um conjunto de passos numerados, ficando esta decisão ao critério do analista. A descrição estruturada tem provado ser bastante eficaz.
- Complementarmente, a UML disponibiliza um conjunto de técnicas, designadas **diagramas de interação**, que permitem descrever de forma gráfica os diversos cenários.
- Nos cenários secundários, as alternativas podem ser introduzidas diretamente no texto da descrição ou quando mais complexas, na linha de caminhos alternativos.
- A representação gráfica do cenário principal e dos cenários alternativos pode ser efetuada em conjunto quando a sua complexidade é reduzida.

Casos de utilização e cenários

Texto livre

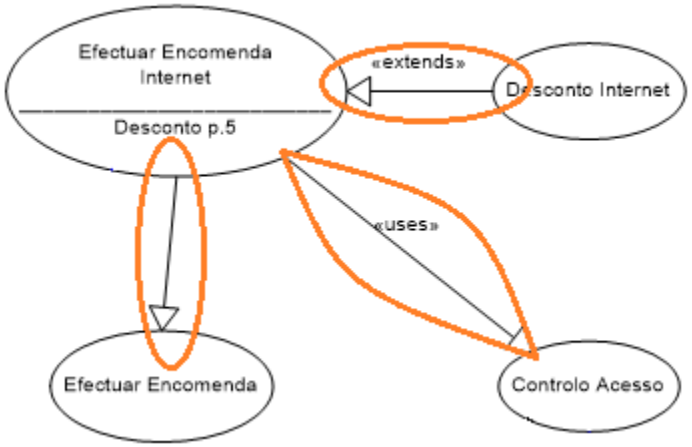
- **"Efetuar Encomenda Internet"**

- O cliente, após ter validado o seu acesso, seleciona a opção Encomendar, sendo mostrado em simultâneo com a sua encomenda o catálogo de produtos. Para adicionar um produto, tem apenas que introduzir o código do mesmo, para que, automaticamente, o seu nome, descrição e preço sejam visualizados no respetivo item da encomenda. Ao mesmo tempo, é calculado o valor total da encomenda.
- Através da opção Confirmar, o cliente confirma a sua encomenda e passa para a função de pagamento, onde após a introdução e confirmação dos dados de cartão de crédito é atribuído um número de identificação à encomenda, que posteriormente será entregue na morada do cliente.

Descrição Estruturada

Efetuar Encomenda Internet (Cenário Principal)	
Pré-condição	O cliente é um utilizador válido no sistema.
Descrição	1. O use case começa quando o cliente selecionada a opção de Encomendar. 2. Em simultâneo com a sua encomenda é mostrado o catálogo de produtos. 3. O cliente adiciona produtos à encomenda através da introdução do código do produto. 4. Automaticamente, o sistema mostra o nome, descrição e preço do produto. 5. De cada vez que é adicionado um produto, o valor total da encomenda é calculado. 6. O cliente confirma a sua encomenda através da opção da opção Confirmar. 7. O sistema pede então os detalhes do cartão de crédito. 8. O sistema confirma os dados do pagamento e atribui um número de identificação à encomenda.
Pós-condição	A encomenda será entregue na morada do cliente.

Relações entre casos de utilização



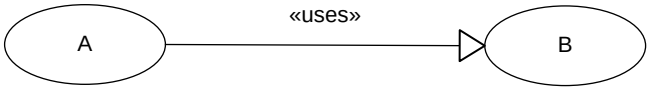
Relações mais frequentes entre use cases:

- <<include>> ou <<uses>>
- <<extend>>
- generalização

Relação <<include>>

- Quando **vários casos de utilização têm uma subsequência de funcionamento comum**, é conveniente separar essa parte comum para um novo caso de utilização que é incluído pelos primeiros

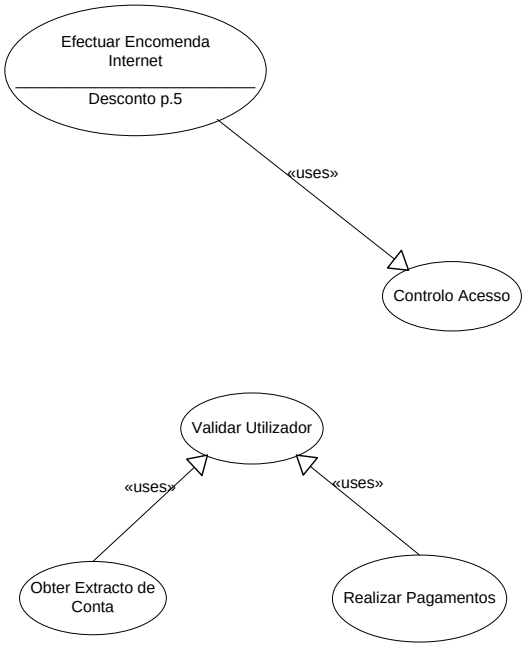
- **Notação**:



(parte comum a outros casos de utilização além de A)

- **Significado**:
  - uma instância do caso de utilização A inclui obrigatoriamente o comportamento especificado por B
  - os atores interagem com A
  - na descrição textual de A: include(B)

Relação <<include>>

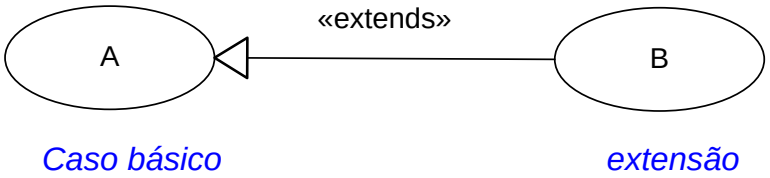


- A relação <<include>> demonstra que a funcionalidade "Controlo Acesso" é utilizada quando uma encomenda é efetuada através da Internet.
- Relação também útil quando existem use cases repetidos, pois evita a sua duplicação no diagrama.
- Os casos de utilização "Obter Extrato de Conta" ou "Realizar Pagamentos" exigem que seja realizada previamente uma validação do utilizador. Para que essa funcionalidade não seja especificada mais que uma vez, os casos anteriores incorporam-na (como sua) ao estabelecerem uma relação de inclusão com o caso "Validar Utilizador"

Relação <<extend>>

- Para simplificar a descrição dos casos de utilização, podem-se organizar os casos de utilização em casos básicos (casos de utilização de acordo com a definição) e extensões aos casos básicos, que traduzem partes ou modalidades acrescentadas condicionalmente (opções)
- Ocorre quando existe um comportamento opcional que deve ser incluído num use case. Este comportamento é definido num segundo use case e invocado pelo use case base, através de um mecanismo de pontos de extensão.

- **Notação**:

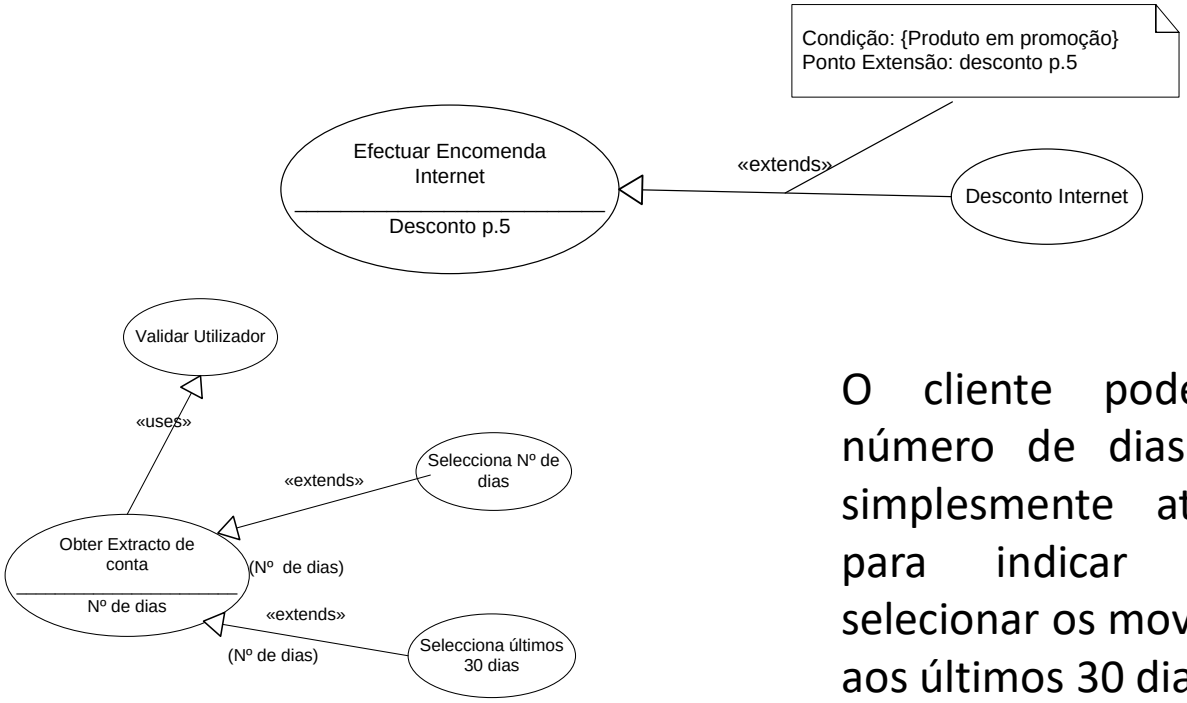


Relação <<extend>>

- **Significado**:
  - uma instância do caso de utilização A pode incluir (sujeito a condições especificadas na extensão) o comportamento especificado por B
  - o caso básico deve fazer sentido sozinho
  - os atores interagem com o caso básico (A)

- Em software, corresponde normalmente a seguir um botão ou um link num formulário que desencadeia uma ação ou dá acesso a outro formulário ou relatório

Relação <<extend>>



O cliente pode selecionar o número de dias pretendido, ou simplesmente ativar um botão para indicar que pretendia selecionar os movimentos relativos aos últimos 30 dias.

Relação de Generalização

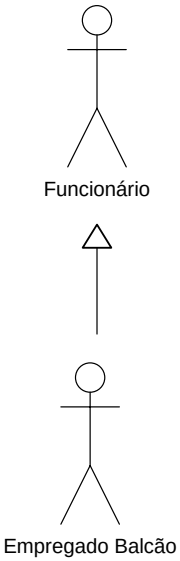
- **Relação de generalização**: entre uma coisa mais genérica e uma coisa mais especializada

- Significa que o caso de utilização "filho" (mais especializado) herda o comportamento, significado e atores do caso de utilização "pai" (mais genérico)
  - O filho pode adicionar ou substituir comportamento do pai
  - O filho pode aparecer em qualquer contexto em que o pai pode aparecer

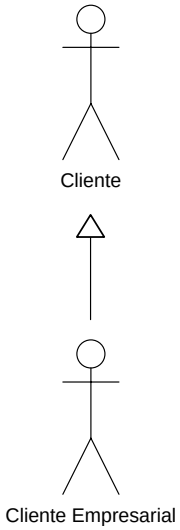
- Permite definir casos à custa de outros já existentes, pelo mecanismos de especialização, ou alternativamente, permite definir casos mais abstratos a partir de casos mais concretos.

- **Notação**:

Relação de generalização entre atores



- Um Empregado de Balcão é um (is a) Funcionário
- O Empregado de Balcão herda as associações (de comunicação com casos de utilização) do Funcionário



- Um cliente empresarial é um (is a) cliente
- O cliente empresarial herda as associações (de comunicação com casos de utilização) do cliente genérico

Proposta de Metodologia

- A metodologia proposta consiste na aplicação sucessiva dos seguintes passos:
  1. Identificar os atores do sistema, ou seja, o perfil dos indivíduos e de outros sistemas que interatuam com o sistema original.
  2. Identificar, para cada ator, os seus casos de utilização principais. Note-se que podem existir casos que envolvam a participação de mais que um ator.
  3. Com base nos casos de utilização originais, identificar, fatorizar e colocar em evidência casos de utilização que sejam recorrentes em mais que um dos casos originais. Nesta situação, cria-se o novo caso de utilização (em geral um caso abstrato) e os casos originais envolvidos estabelecem uma relação de inclusão com o dito caso. Repetir o processo até não se conseguir identificar qualquer outro caso a reutilizar.

Proposta de Metodologia

1. Para tratar casos de utilização que pretendam ser flexíveis e versáteis, definir pontos de extensão e conjuntamente definir um ou mais casos de utilização (abstratos) que os permitam estender nesses pontos. Nesta situação, cria-se uma relação de extensão do caso abstrato para o caso estendido.
2. Especificar textualmente cada caso de utilização segundo um determinado formato previamente definido. Não esquecer nesta especificação textual a explicitação dos pontos de extensão e de inclusão identificados.

Descoberta de Atores

- Definir as entidades interessadas em usar e interagir com o sistema.
- Podemos depois colocar-nos no lugar do ator e tentar identificar os seus requisitos para o sistema e quais os use cases necessários.
- Perguntas que ajudam a descobrir atores:
  - Quem vai usar a funcionalidade principal do sistema (atores principais)?
  - Quem precisa do sistema para o apoiar nas suas tarefas diárias?
  - Quem fará a administração, manutenção e manterá o sistema a funcionar (atores secundários)?
  - Que dispositivos de hardware vão ser controlados pelo sistema?
  - Com que outros sistemas vai este sistema interagir?
  - Quem tem interesse nos resultados (valores) que o sistema produz?

Descoberta de Use Cases

- O ponto de partida é a lista de atores inicial. Para cada um fazem-se as perguntas:
  - Que funções necessita do sistema?
  - O que é que o ator precisa de fazer?
  - O ator necessita de ler, criar, destruir, modificar ou armazenar algum tipo de informação no sistema?
  - O ator deve ser notificado de quaisquer eventos no sistema ou deve o ator notificar o sistema acerca de algum evento?
  - O que representam esses eventos em termos de funcionalidade?
  - Pode melhorar-se, tornando mais simples ou mais eficiente, o trabalho do dia-a-dia do ator através de novas funcionalidades acrescentadas ao sistema?

NORTE2020 PORTUGAL 2020 UNIÃO EUROPEIA Fundo Social Europeu

Descoberta de Use Cases

- Outras questões que não envolvem nenhum dos atores atuais e para as quais deve depois encontrar-se um ator (um use case deve estar sempre associado a, pelo menos, um ator):
  - Quais são os "inputs/outputs" necessários?
  - De onde vêm e para onde vão?
  - Quais são os principais problemas com a versão atual do sistema (provavelmente manual)?

Exercício

- Considere o sistema de uma equipa de futebol constituído pelos seguintes atores:
  - jogador,
  - treinador,
  - atacante,
  - guarda-redes,
  - médio,
  - defesa,
  - presidente.
- Desenhe o respetivo diagrama de casos de utilização.
- Sugestão: considere por exemplo os seguintes casos:
  - jogar,
  - treinar,
  - defender a baliza,
  - pagar ao jogador,
  - pagar ao treinador,
  - vender jogador,
  - contratar jogador,
  - contratar treinador,
  - despedir treinador.