



# Desenvolvimento de Sistemas Software

## Modelação do Requisitos Funcionais (*Use Cases*)



# Onde estamos...

## Planeamento

- Decisão de avançar com o projecto
- Gestão do projecto

## Análise

- Análise do domínio do problema
- Análise de requisitos

## Concepção

- Concepção da Arquitectura
- Concepção do Comportamento

## Implementação

- Construção
- Teste
- Instalação
- Manutenção



# Tipos de requisitos

- **Requisitos funcionais** - O que o sistema deve fazer
  - Descrevem as interacções entre o Sistema e o seu ambiente
    - Descrições independentes da implementação
    - Ambiente: Utilizadores e outros sistemas (externos)
- **Requisitos não funcionais** - Como o sistema deve fazê-lo
  - Aspectos do sistema não directamente relacionados com o seu comportamento funcional
    - *Usability; Dependability* (*reliability, robustness, safety*); *Performance*; *Maintainability; Portability*; e ainda: de implementação, de interface (com outros sistemas), de operação, de modo de entrega, legais.



# Validação dos requisitos

- Os requisitos são **validados com o cliente**
- Validação deve considerar:
  - **Completude** - todos os aspectos relevantes foram considerados
  - **Consistência** - não existem contradições entre requisitos
  - (ausência de) **Ambiguidade** - nenhum requisitos pode ser interpretado de formas diferentes
  - **Correcção** - os requistes descrevem correctamente o que o cliente pretende e o que a equipa de desenvolvimento se propõe fazer

A especificação de requisitos deve ainda ser...

- **Realista** - não se deve prometer o que não podemos realizar
- **Verificável** - temos que poder saber se cumprimos os objectivos!
- **Rastreável** - porque é que cada requisito lá está?



# Definição de requisitos (funcionais)

## Visão orientada aos *Use Case*

1. Identificar **Cenários** - descrição informal, mas concreta e focada, de uma interacção com o Sistema do ponto de vista de um utilizador
  - Ajudam a análise pois são compreensíveis para os clientes
2. Identificar e especificar os ***Use Cases*** (Casos de Uso) - descrições de fluxos de interacção com o Sistema para suportar os cenários
  - Cenários são instâncias dos *Use Case*
3. Identificar **Actores** - entidades externas que interagem com o sistema
4. Identificar relações entre Actores e *Use Cases*
5. Identificar componentes de suporte à realização dos *Use Case* - como vai a funcionalidade necessária ser implementada

## *Vantagens:*

- Não há trabalho desnecessário.
- O Sistema de Informação suporta as tarefas do cliente.
- As fronteiras do Sistema ficam bem definidas.



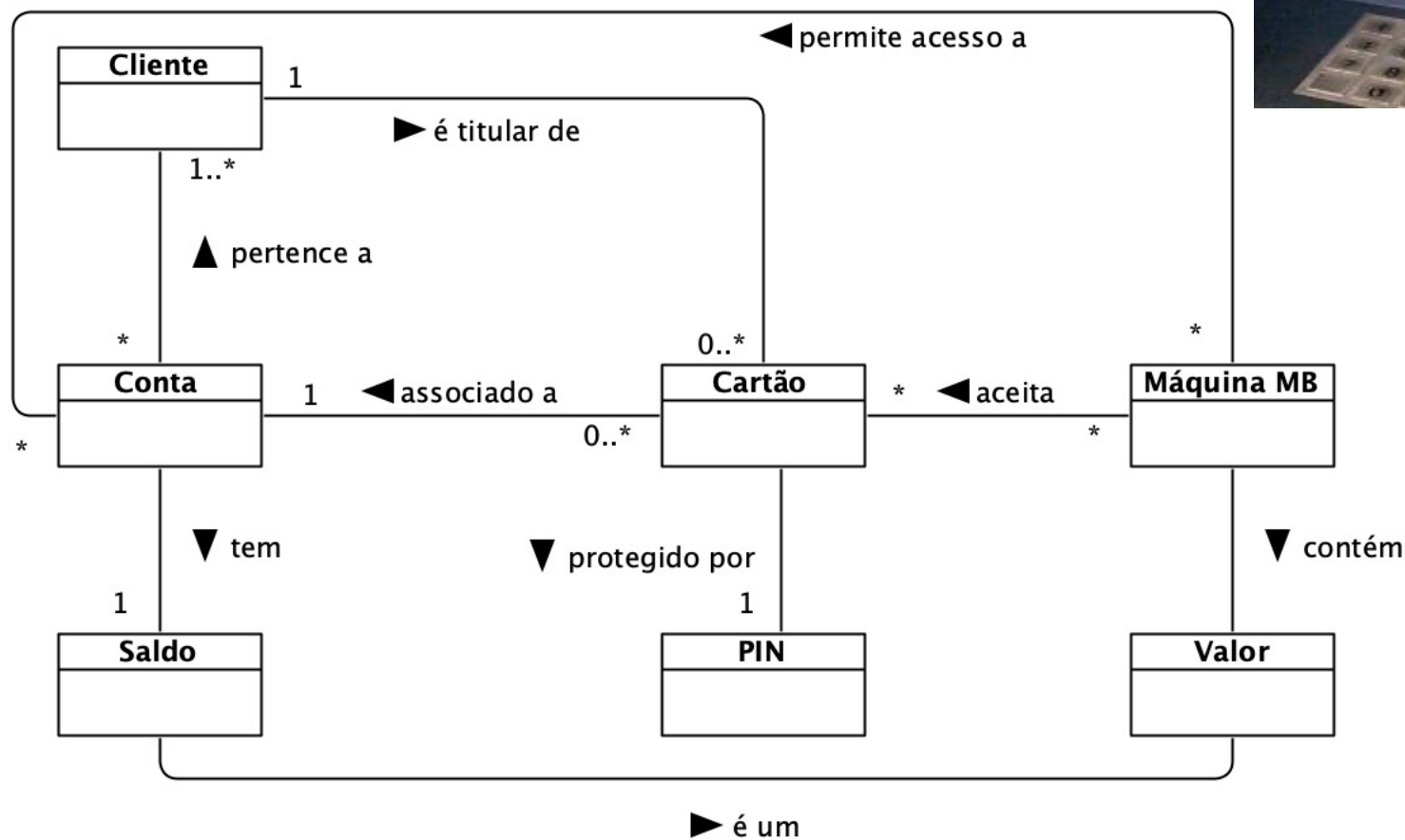
## Definição de *Use Case*

- Descreve como os Actores atingem objectivos (realizam os *Use Cases*) utilizando o sistema
  - Definem relação entre *inputs* dos Actores e comportamento do Sistema
- Especificação deve incluir o comportamento tipicamente esperado, bem como variantes:
  - Comportamentos alternativos que ainda levam ao sucesso
  - Comportamentos de insucesso (Excepções)
  - vamos também definir as pré-condições e pós-condições de cada use case (cf. *design by contract*).



# Exemplo - Máquina Multibanco

- Modelo de Domínio



# Exemplo - Máquina Multibanco

- **Cenários**

1. O João levanta €60 com cartão  
O João vai viajar e dirige-se a uma máquina MB para levantar dinheiro para a viagem. Introduce o cartão e o código PIN quando tal é solicitado pela máquina. No menu escolhe a opção de levantamento de €60. A máquina pergunta ao João se pretende um talão e ele responde que não. A máquina disponibiliza então o cartão e o valor pedido, que o João retira.
2. O João levanta €10 com MB way  
...
3. A Maria paga a conta da luz  
...
4. O Rui transfere a mesada para a conta da filha  
...
5. A Joana abastece a máquina com notas  
...
6. ...







# Exemplo - Máquina Multibanco

- **Cenários**

1. O João levanta €60 com cartão
2. O João levanta €10 com MB way
3. A Maria paga a conta da luz
4. O Rui transfere a mesada para a conta da filha
5. A Joana abastece a máquina com notas
6. ...

- **Use Cases**

- Levantar €
- Pagar serviço
- Efectuar transferência
- Carregar máquina

**Sempre um verbo!**

*Fazer algo...*

(Use Case descreve como fazê-lo)

- **Actores**

- Cliente (cf. João, Maria, Rui)
- Bancário (cf. Joana)



# Definição de *Use Case* - Especificação

- A UML não especifica como descrever *Use Cases*
  - *Tem que ser definido por cada organização ou projecto*
- *Muitas abordagens são possíveis/comuns*
  - *Desde descrições textuais até especificações via diagramas*
  - *Mais ou menos verbosas e detalhadas*
- Em DSS vamos utilizar uma notação semi-estruturada:
  - **Use Case:** nome do use case
    - **Descrição:** breve descrição do use case
    - **Cenários:** cenários que originam o use case
    - **Pré-condição:** o que deve ser verdade **no Sistema** para que executar o use case seja válido
    - **Pós-condição:** condição de sucesso do use case
    - **Fluxo normal:**
      - Fluxo de eventos mais comum
    - **Fluxos alternativos:**
      - Especificação dos modos alternativos de completar o use case
    - **Fluxos de excepção:**
      - Especificação de situações relevantes em que o use case não tem sucesso

(cf. Verificável!)



# Exemplo - Máquina Multibanco

- **Use Case:** *Levantar €*
  - **Descrição:** *Cliente levanta quantia da máquina*
  - **Cenários:** *O João levanta €60 com cartão*
  - **Pré-condição:** *Sistema tem notas*
  - **Pós-condição:** *Cliente tem quantia desejada e saldo da conta foi actualizado*
  - **Fluxo normal:**
    1. *Cliente apresenta cartão e PIN*
    2. *Máquina MB pede operação*
    3. *Cliente indica que pretende levantar dada quantia*
    4. *Máquina MB pergunta se quer talão*
    5. *Cliente responde que não*
    6. *Máquina MB devolve cartão e fornece notas*
    7. *Cliente retira cartão e notas*

- **Fluxo Alternativo 1:**

O João vai viajar e dirige-se a uma máquina MB para levantar dinheiro para a viagem. Introduce o cartão e o código PIN quando tal é solicitado pela máquina. No menu escolhe a opção de levantamento de €60. A máquina pergunta ao João se pretende um talão e ele responde que não. A máquina disponibiliza então o cartão e o valor pedido, que o João retira.
- **Fluxo Alternativo 2:**

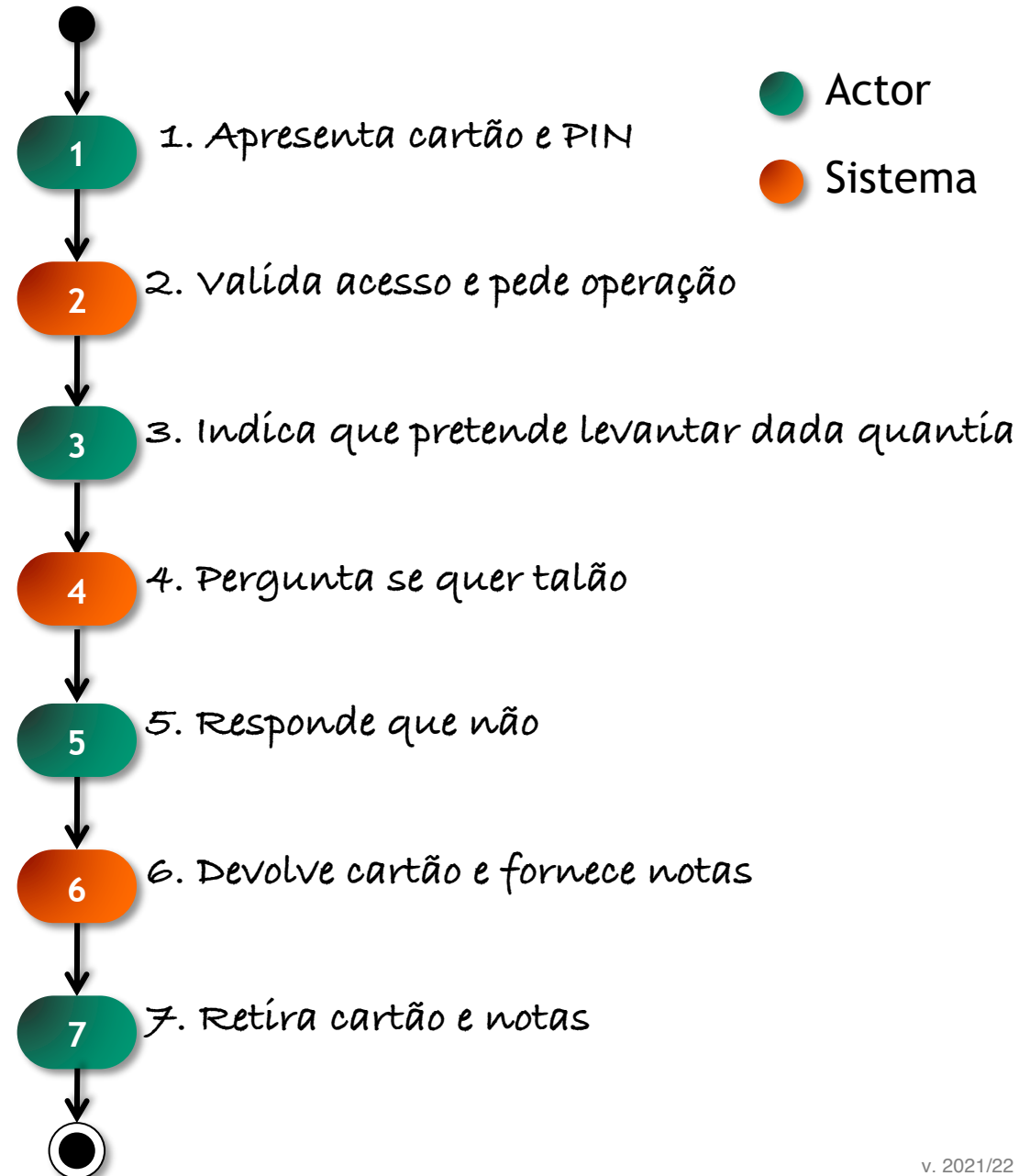


# Exemplo - Máquina Multibanco

- **Use Case: Levantar €**
  - **Descrição:** Cliente levanta quantia da máquina
  - **Cenários:** O João levanta €60 com cartão
  - **Pré-condição:** Sistema tem notas
  - **Pós-condição:** Cliente tem quantia desejada e saldo da conta foi actualizado
  - **Fluxo normal:**
    1. Cliente apresenta cartão e PIN ✓
    2. Máquina MB pede operação ✓
    3. Cliente indica que pretende levantar dada quantia ✓
    4. Máquina MB pergunta se quer talão ✓
    5. ~~Cliente responde que não~~
    6. ~~Máquina MB devolve cartão e fornece notas~~
    7. ~~Cliente retira cartão e notas~~
  - **Fluxo alternativo (1):** [cliente quer talão] (passo 5)
    - 5.1. Cliente responde que ~~sim~~ ✓
    - 5.2. Máquina MB devolve cartão, notas e talão ✓
    - 5.3. Cliente retira cartão, notas e talão ✓
  - **Fluxo de excepção (2):** [PIN inválido] (passo 6)
    - 6.1. Máquina MB avisa sobre PIN inválido e fornece cartão
    - 6.2. Cliente retira cartão



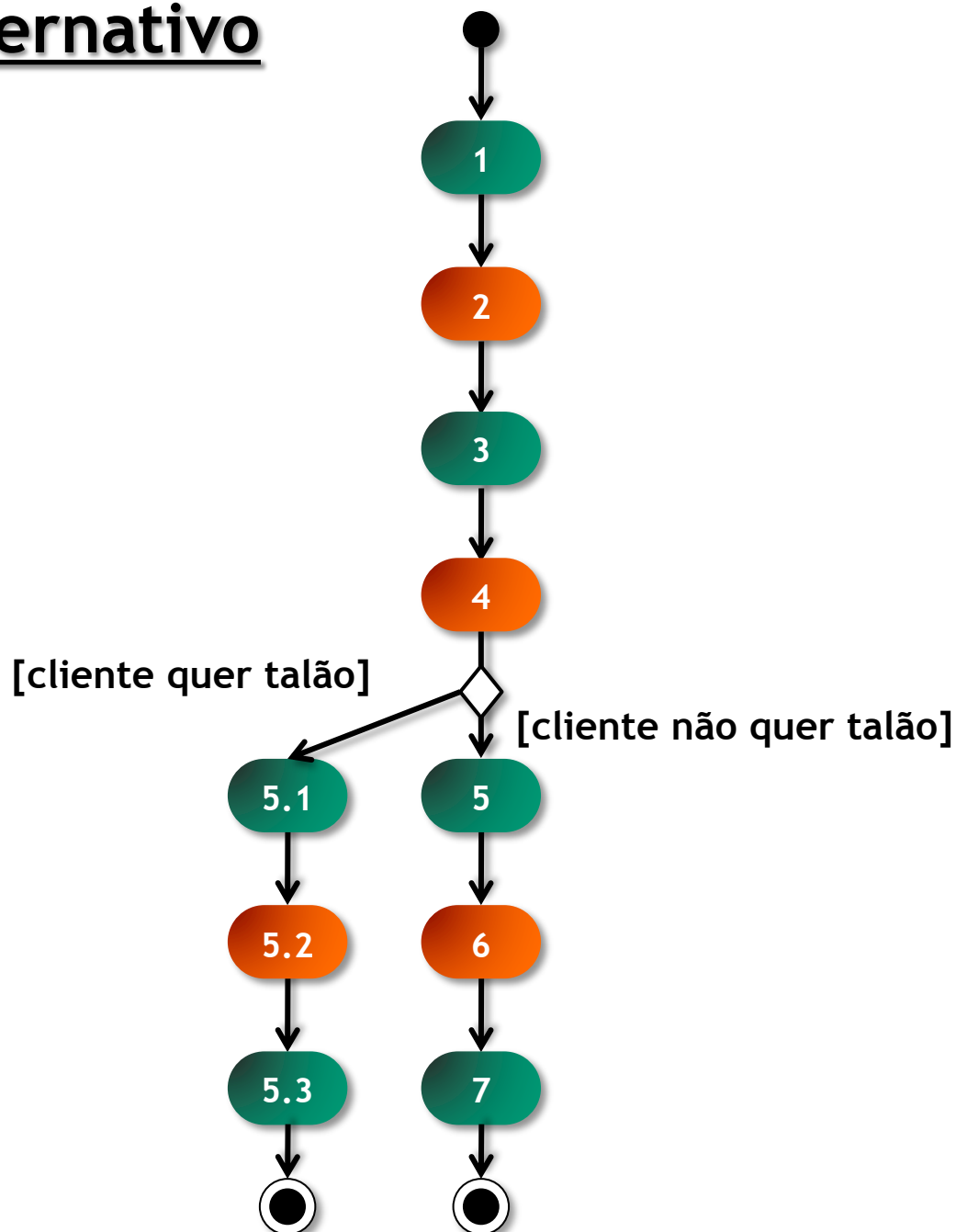
# Fluxo normal





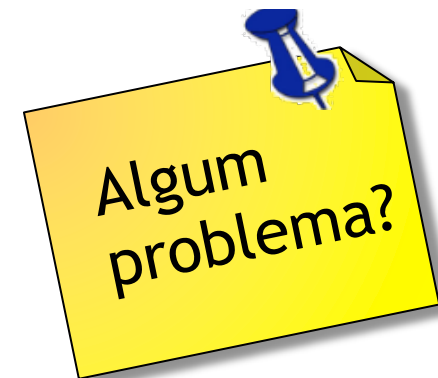
# Fluxo alternativo

● Actor  
● Sistema



# Exemplo - Máquina Multibanco

- Use Case: *Levantar €*
  - Descrição: *Cliente levanta quantia da máquina*
  - Cenários: *O João levanta €60 com cartão*
  - Pré-condição: *Sistema tem notas*
  - Pós-condição: *Cliente tem quantia desejada e saldo da c*
  - Fluxo normal:
    1. Cliente apresenta cartão e PIN ✓
    2. Máquina MB pede operação ✓
    3. Cliente indica que pretende levantar dada quantia ✓
    4. Máquina MB pergunta se quer talão ✓
    5. Cliente responde que não ✓
    6. ~~Máquina MB devolve cartão e fornece notas~~
    7. ~~Cliente retira cartão e notas~~
  - Fluxo alternativo (1): [cliente quer talão] (passo 5)
    - 5.1. Cliente responde que sim
    - 5.2. Máquina MB devolve cartão, notas e talão
    - 5.3. Cliente retira cartão, notas e talão
  - Fluxo de exceção (2): [PIN inválido] (passo 6)
    - 6.1. Máquina MB avisa sobre PIN inválido e fornece cartão ✓
    - 6.2. Cliente retira cartão ✓



**74526**

live.voxvote.com

Nickname: nº aluno!



# Fluxo de exceção

Quer talão e PIN inválido?  
(qualidade do *Use Case*)

Algun problema?

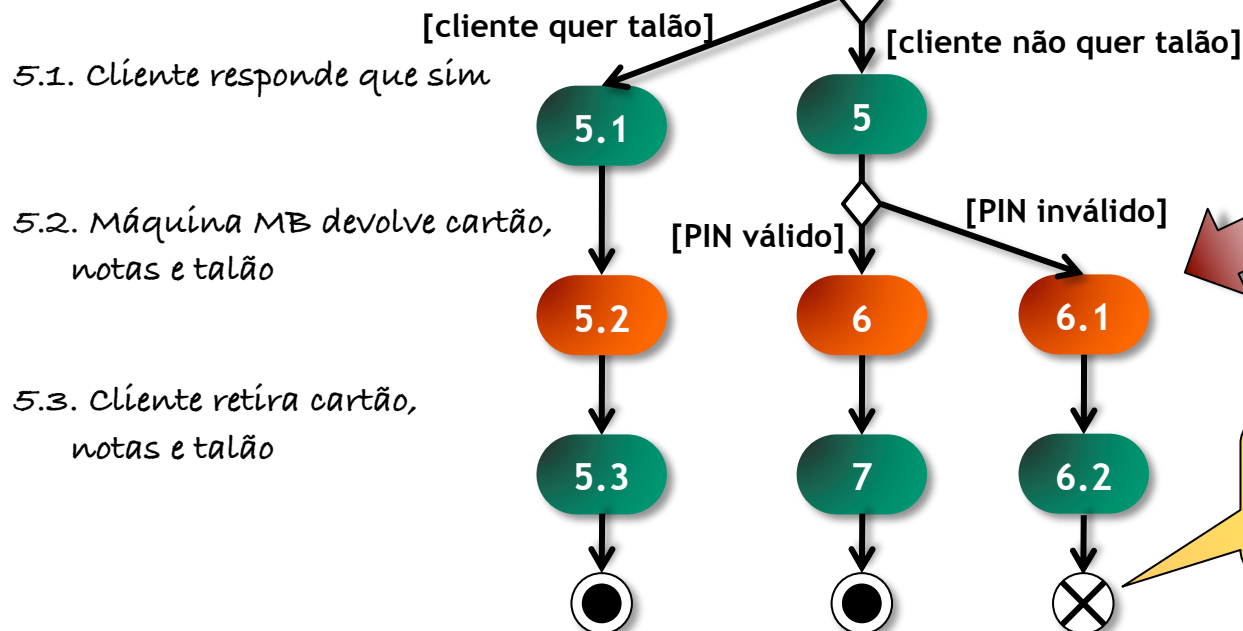
● Actor  
● Sistema



74526

live.voxvote.com

Nickname: n° aluno!



Só avisa aqui?  
(qualidade do Sistema)

Não termina com sucesso.  
Pós-condição não se verifica



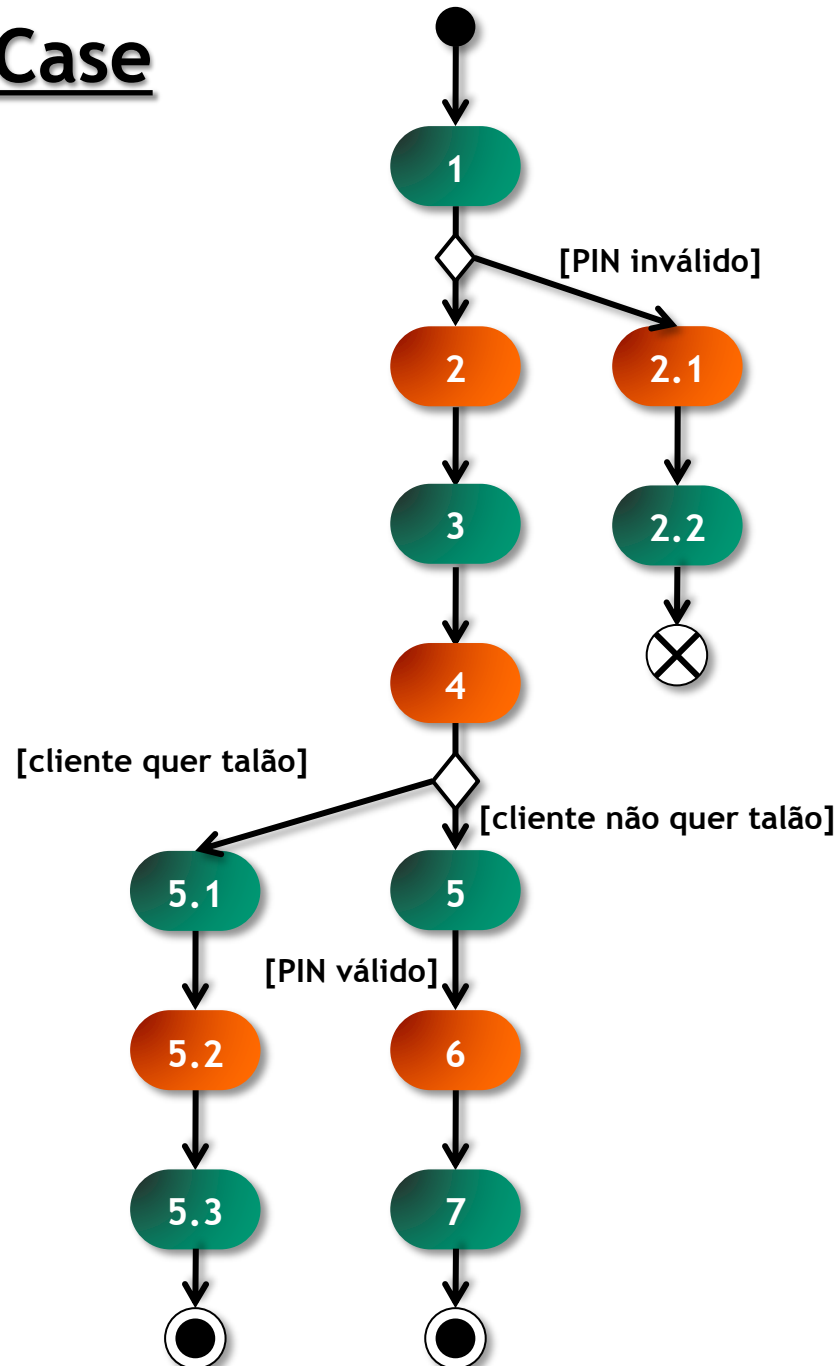


# Exemplo - Máquina Multibanco

- **Use Case:** *Levantar €*
  - **Descrição:** *Cliente levanta quantia da máquina*
  - **Cenários:** *O João levanta €60 com cartão*
  - **Pré-condição:** *Sistema tem notas*
  - **Pós-condição:** *Cliente tem quantia desejada e saldo da conta foi actualizado*
  - **Fluxo normal:**
    1. *Cliente apresenta cartão e PIN*
    2. *Máquina MB **válida acesso e** pede operação*
    3. *Cliente indica que pretende levantar dada quantia*
    4. *Máquina MB pergunta se quer talão*
    5. *Cliente responde que não*
    6. *Máquina MB devolve cartão e fornece notas*
    7. *Cliente retira cartão e notas*
  - **Fluxo alternativo (1):** *[cliente quer talão] (passo 5)*
    - 5.1. *Cliente responde que sim*
    - 5.2. *Máquina MB devolve cartão, notas e talão*
    - 5.3. *Cliente retira cartão, notas e talão*
  - **Fluxo de excepção (2):** *[PIN inválido] (**passo 2**)*
    - 2.1. *Máquina MB avisa sobre PIN inválido e fornece cartão*
    - 2.2. *Cliente retira cartão*



# Melhorar o Use Case



● Actor  
● Sistema



# Exemplo - Máquina Multibanco

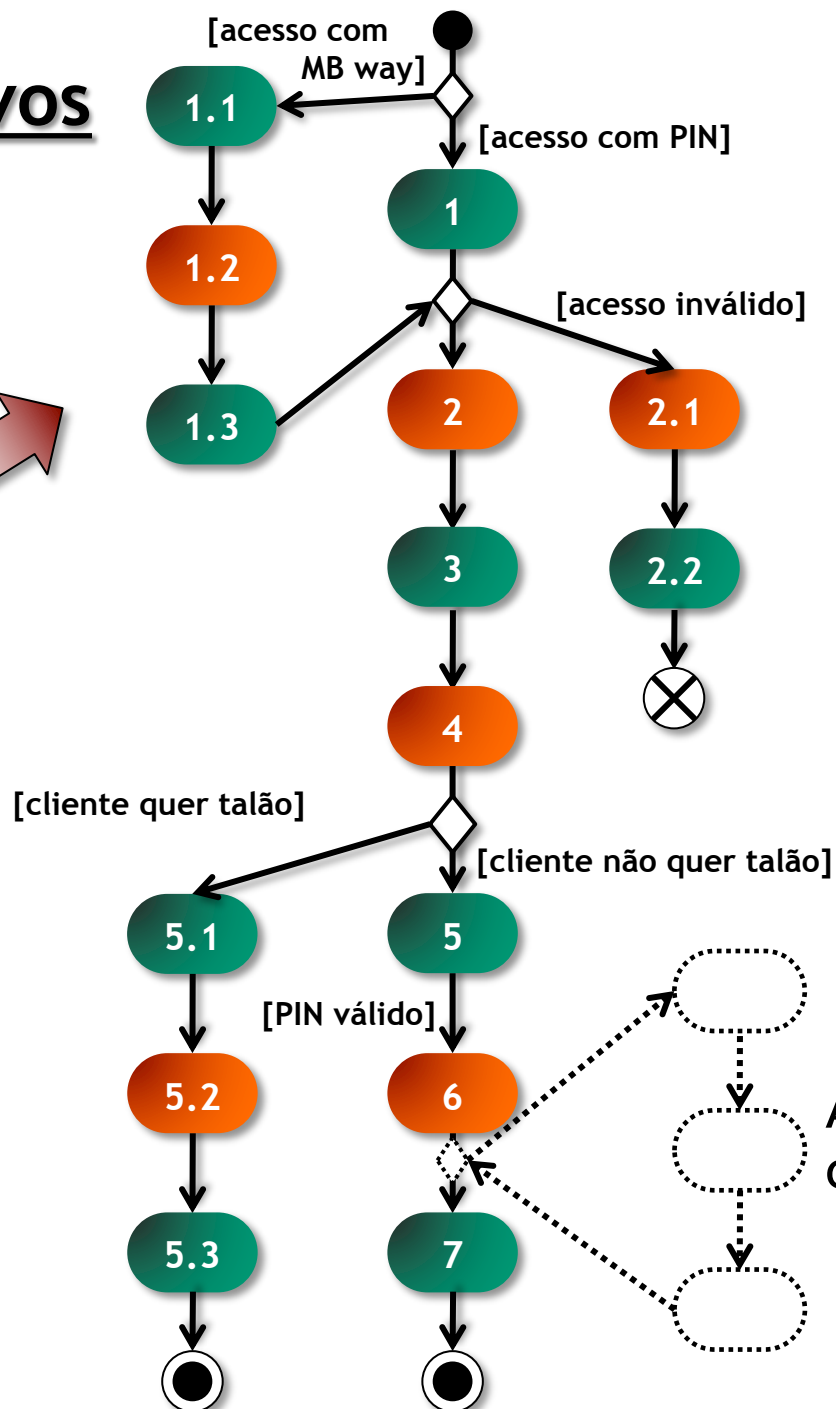
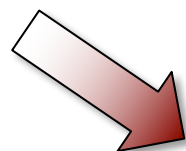
- Use Case: Levantar €
  - Descrição: Cliente levanta quantia da máquina
  - Cenários: O João levanta €60 com cartão; O João levanta €10 com MB way
  - Pré-condição: Sistema tem notas
  - Pós-condição: Cliente tem quantia desejada e saldo da conta foi actualizado
  - Fluxo normal:
    1. ~~Cliente apresenta cartão e PIN~~
    2. Máquina MB valida acesso e pede operação
    3. Cliente indica que pretende levantar dada quantia
    4. Máquina MB pergunta se quer talão
    5. Cliente responde que não
    6. Máquina MB devolve cartão e fornece notas
    7. Cliente retira cartão e notas
  - Fluxo alternativo (1): [cliente quer talão] (passo 5)
    - ...
  - Fluxo de excepção (2): [PIN inválido] (passo 2)
    - ...
  - Fluxo alternativo (3): [cliente autentica-se com MB way] (passo 1)
    - 5.1. Cliente escolhe acesso MB way
    - 5.2. Máquina MB pede Código MB way
    - 5.3. Cliente indica Código MB way
    - 5.4 Regressa a 2

MB way?! Temos que actualizar o Modelo de Domínio!!



# Fluxos alternativos

Substituem  
comportamento



● Actor  
● Sistema

Adiciona  
comportamento



# Exemplo - Máquina Multibanco

- **Use Case:** *Levantar €*
    - **Descrição:** *Cliente levanta quantia da máquina*
    - **Cenários:** *O João levanta €60 com cartão; O João levanta €10 com MB way*
    - **Pré-condição:** *Sistema tem notas*
    - **Pós-condição:** *Cliente tem quantia desejada e saldo da conta foi actualizado*
    - **Fluxo normal:**
      1. *Cliente apresenta cartão e PIN*
      2. *Máquina MB valida acesso e pede operação*
      3. *Cliente indica que pretende levantar dada quantia*
      4. *Máquina MB pergunta se quer talão*
      5. *Cliente responde que não*
      6. *Máquina MB devolve cartão e fornece notas*
      7. *Cliente retira cartão e notas*
    - **Fluxo alternativo (1):** *[cliente quer talão] (passo 5)*
      - 5.1. *Cliente responde que sim*
      - 5.2. *Máquina MB devolve cartão, notas e talão*
      - 5.3. *Cliente retira cartão, notas e talão*
    - **Fluxo de excepção (2):** *[PIN inválido] (passo 2)*
      - 2.1. *Máquina MB avisa sobre PIN inválido e fornece talão*
      - 2.2. *Cliente retira cartão*
    - **Fluxo alternativo (3):** *[cliente autentica-se com MB way]*
      - 5.1. *Cliente prime escolhe acesso MB way*
      - 5.2. *Máquina MB pede Código MB way*
      - 5.3. *Cliente indica Código MB way*
      - 5.4. *Regressa a 2*
- Cartão ilegível?
  - € insuficiente na máquina?
  - Valor diário excedido?
  - Quantia impossível com notas existentes?
  - Cliente quer desistir?
  - Ligação ao servidor cai? (!)
  - Dispensador de notas encrava? (!)
  - etc., etc., etc.



# Definição de *Use Case* - Tipos de fluxos

- Em cada especificação de um *Use Case* podem/devem existir diferentes fluxos de controlo (sequências de eventos, comportamentos)
- Podemos caracterizá-los em três tipos:

- Fluxo Normal (ou Principal)

O fluxo mais comum. Representa uma situação perfeita em que nada corre mal. A pós-condição é satisfeita no final (se pré-condição também o é no início).

- Fluxos Alternativos

Fluxos válidos mas menos comuns.

A pós-condição é satisfeita (se pré-condição também o é no início)

- Fluxos de Excepção

Condições de erro suficientemente importantes para serem capturadas no modelo.

A pós-condição NÃO é satisfeita.



# Identificação *de Use Cases*

Etapas a cumprir (com o auxílio de **cenários de utilização** do sistema):

1. Identificar actores (quem utiliza o sistema)
2. Identificar *use cases* (o que se pode fazer no sistema)
3. Identificar associações (quem pode fazer o quê)

## *Identificar actores*

- Quem vai utilizar o sistema?
- Neste caso: Cliente, Bancário, Técnico de Manutenção?, Impressora?, Servidor do banco?

## *Identificar Use Cases*

- Objectivos dos utilizadores/actores?
- Resposta a estímulos externos.



# Use Cases - Especificação

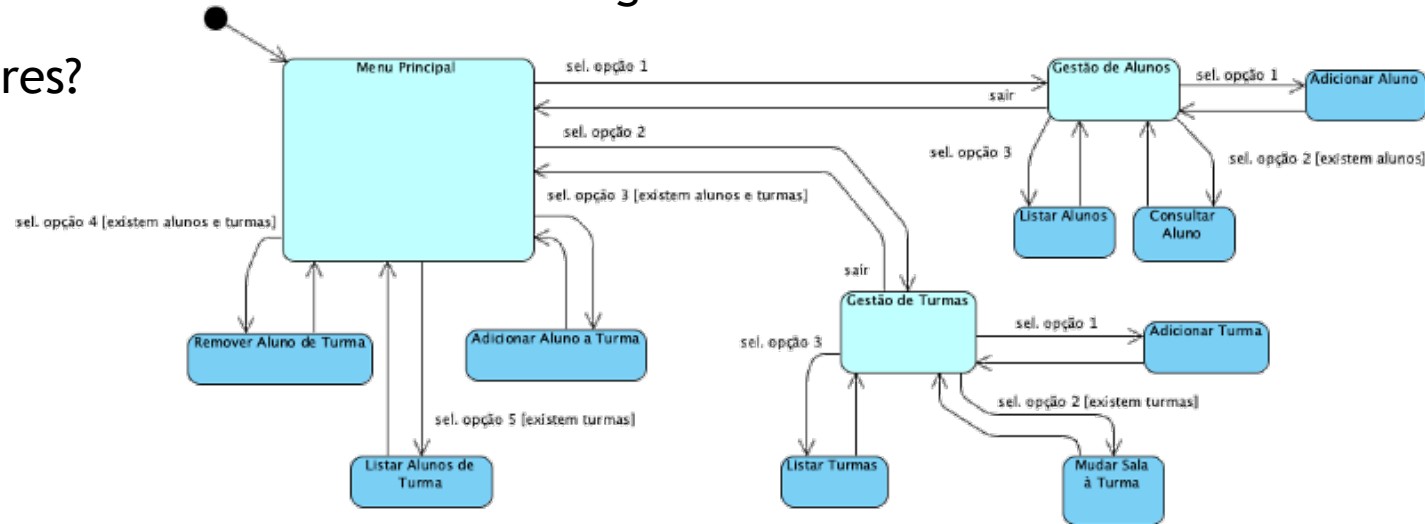
- Não escrever *Use Cases* demasiados longos
  - Focar no que é essencial garantir
- Entidades referidas no Use Case devem estar presentes no Modelo de Domínio
  - Modelo de Domínio descreve o contexto do problema
  - Modelo de Use Case descreve uma solução
  - Conceitos têm que ser os mesmos!
- Deve ser expresso ao nível dos requisitos dos Actores (utilizadores/sistemas)
  - Não devem especificar a interface com o utilizador!!





# Exercício: o exemplo das turmas...

- Que *Use Cases* fazem sentido num sistema de gestão de turnos.
- Quem são os actores?



```

*** Gestão de Turmas ***
1 - Adicionar Turma
2 - Mudar Sala à Turma
3 - Listar Turmas
0 - Sair
Opção: 0
  
```

```

*** Menu ***
1 - Operações sobre Alunos
2 - Operações sobre Turmas
3 - Adicionar Aluno a Turma
4 - Remover Aluno de Turma
5 - Listar Alunos de Turma
0 - Sair
Opção: 1
  
```

```

*** Gestão de Alunos ***
1 - Adicionar Aluno
2 - Consultar Aluno
3 - Listar Alunos
0 - Sair
Opção:
  
```



# *Use Cases*

*“Good use cases are balanced, describing essential system behavior while providing only the necessary details about the interactions between system and its users”*

S. Adolph & P. Bramble (2002) *Patterns for Effective Use Cases*.  
The Agile Software Development Series. Addison-Wesley Professional.