

# ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO

# P.PORTO

**CTeSP DWDM**

Análise e Arquitetura de Sistemas

UML: Diagramas de Atividades

# Diagrama de Atividades

- Fluxograma mostrando o fluxo de controlo entre atividades
- Usado para modelação de aspetos dinâmicos do sistema
- Insere-se no conjunto de artefactos que permitem a modelação comportamental dos requisitos.
- Os diagramas de atividades permitem ainda:
  - descrever conjuntos de atividades que se desenvolvem em paralelo
  - atribuir a uma classe responsabilidade pela execução de uma atividade

**Cada um dos diagramas propostos pela UML permite modelar um aspeto específico do sistema e deve ser utilizado em complemento com outros diagramas**

# Diagrama de Atividades - utilização

- Usado para descrever processos (dos mais variados tipos):
  - processos de negócio
  - processo de software como um workflow (fluxo de trabalho) por meio de uma serie de ações
  - etapas de um caso de uso
  - operação/método de uma classe
  - protocolo de software
  - algoritmos de software





# Exemplo

Considerando o exemplo da PhonePizza, e pensando no use case “Processar Encomenda na Pizzaria”:


“O cliente dirige-se ao balcão e pede ao funcionário um conjunto de produtos que pretende. O funcionário vai tomando nota do pedido, verificando se o produto está na lista de produtos comercializados e se existe em stock. No caso do produto não existir, informa o cliente. Se for detetada uma rotura de stock, é enviada uma mensagem ao Gestor da Loja para encomendar o produto em falta e o funcionário sugere um produto alternativo. Se o produto solicitado não pertencer à lista dos que são vendidos na pizzaria, o funcionário sugere igualmente um produto alternativo. Após o cliente ter concluído a sua encomenda, é determinado o valor da mesma e solicitado o pagamento. Se o pagamento for válido, a encomenda é entregue ao cliente. Caso contrário, a encomenda é cancelada.”



# Componentes Básicos

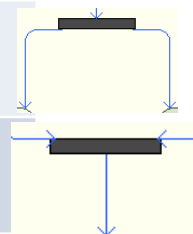
Elemento	Descrição	
<b>Pistas de responsabilidade</b>	<ul style="list-style-type: none"><li>○ <i>Swimlanes</i></li><li>○ São separadas por linhas contínuas</li><li>○ Cada pista tem o nome da unidade organizacional, entidade ou objeto responsável pelas ações e atividades aí localizadas</li></ul>	
<b>Atividade</b>	<ul style="list-style-type: none"><li>○ Descrever um conjunto de ações realizadas</li></ul>	
<b>Atividade inicial</b>	<ul style="list-style-type: none"><li>○ Pode ser virtual - identificar o início do diagrama</li><li>○ Pode corresponder a uma atividade operacional do sistema</li><li>○ Só pode haver <b>uma</b></li><li>○ Representada por um círculo preenchido a negro</li></ul>	
<b>Atividade de término</b>	<ul style="list-style-type: none"><li>○ Pode existir <b>mais do que uma</b></li><li>○ Representada por um círculo a negro, limitado por uma circunferência</li></ul>	
<b>Transição entre atividades</b>	<ul style="list-style-type: none"><li>○ Descrever a sequência pela qual as atividades se realizam.</li><li>○ Pode listar eventos, ações e condições.</li><li>○ Representada por uma seta</li></ul>	

# Componentes Básicos

Elemento	Descrição
<b>Comportamento condicional</b>	Definição de caminhos alternativos
<b>1. Guardas</b>	<ul style="list-style-type: none"><li>Expressões booleanas que têm de ser verificadas para se realizar a transição para uma nova atividade</li><li>limitadas por parêntesis retos [ ] <span>[x é verdade]</span></li></ul>
<b>2. Diamantes de decisão</b>	<ul style="list-style-type: none"><li>Para representar caminhos alternativos baseados numa expressão condição.</li><li>Devem ser utilizados para aumentar a legibilidade do diagrama.</li><li>Podem ser usados para descrever uma divergência (<b>branch</b>) ou uma convergência (<b>merge</b>) </li></ul>



# Componentes Básicos

Elemento	Descrição
<b>Processamento paralelo</b>	<ul style="list-style-type: none"><li>• Representar fluxos de atividades que se desenvolvem em paralelo</li><li>• Útil na descrição de processos organizacionais<ul style="list-style-type: none"><li>• Porque ajuda a identificar oportunidades para aumentar a eficiência do processo</li><li>• Através da realização de atividades em paralelo</li></ul></li><li>• <u>Uma barra de divergência deve ser compensada com uma barra de convergência</u></li></ul>
<b>1. Fork</b>	<ul style="list-style-type: none"><li>• É um <b>ponto de divergência</b> a partir do qual duas ou mais tarefas se podem iniciar em paralelo</li></ul>
<b>2. Join</b>	<ul style="list-style-type: none"><li>• Permite sincronizar tarefas que têm de estar concluídas para que se inicie uma nova tarefa (<b>ponto de convergência</b>).</li></ul>

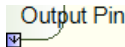





# Componentes Básicos

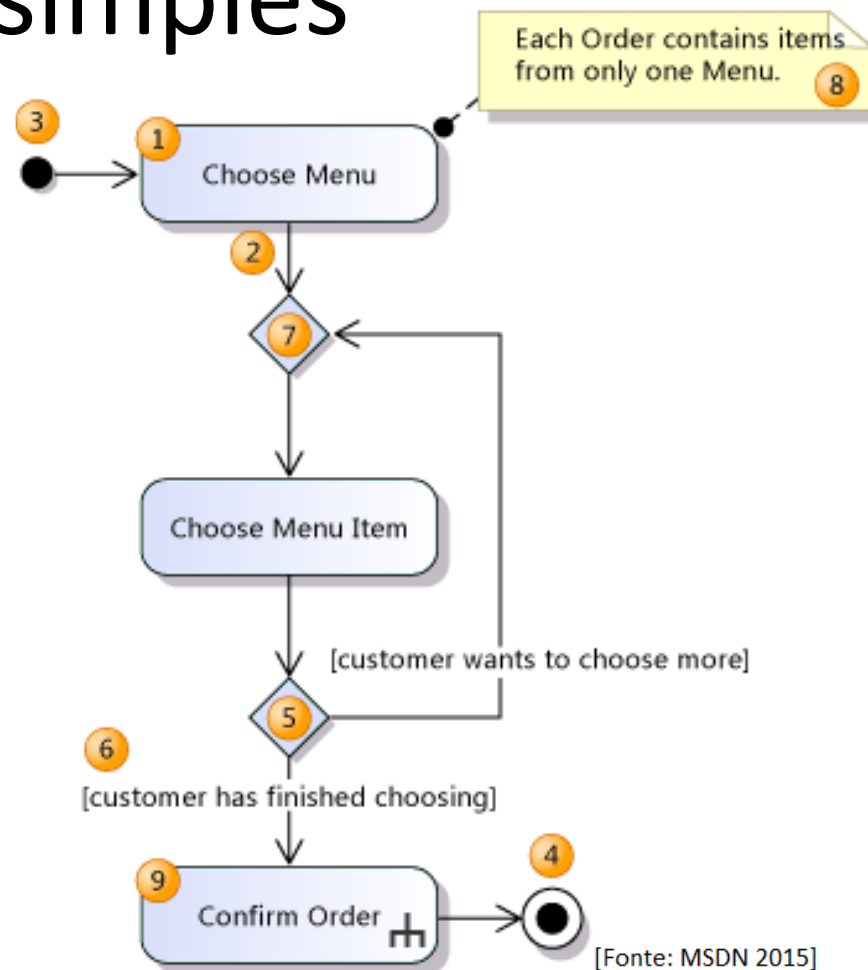
Elemento	Descrição
<b>Eventos</b>	
<b>1. Envio de Sinal</b>	
<b>2. Receção de Sinal</b>	

# Uso de diagramas de atividades

- Descrição do controlo do fluxo de informação
- Descrever decisões e loops
- Descrever fluxo de dados
  - existe a intervenção de objetos para a realização das atividades. Pode ser representada colocando estes objetos nos diagramas e ligando-os à atividade através do símbolo de dependência
  - utilização dos seguintes elementos:
    - output pins 
    - input pins 
- Descrever fluxos de dados concorrentes (executados em simultâneo)
  - Uso de forks e joins
- Descrever sinais e eventos

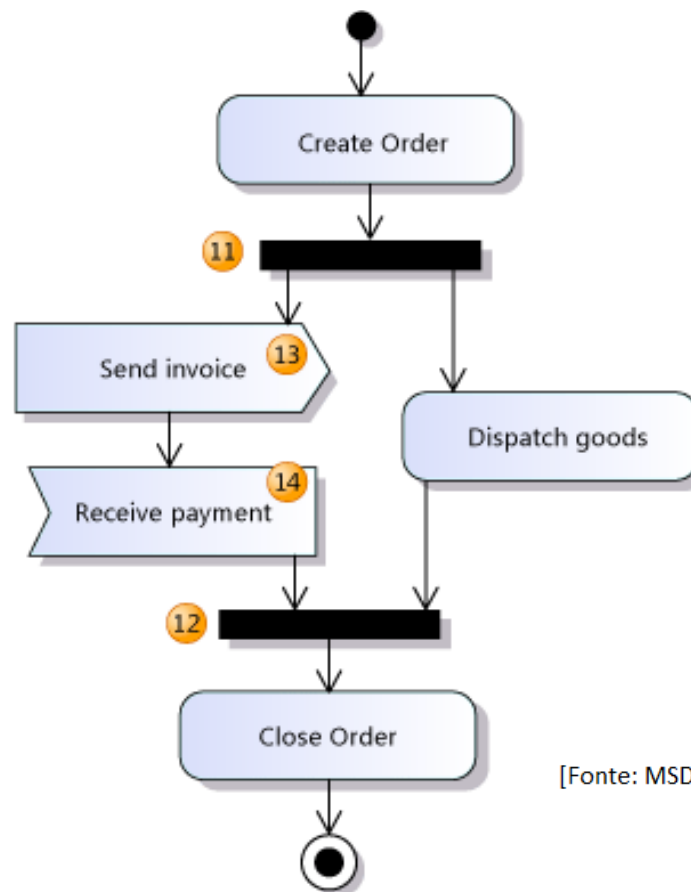
# Fluxo simples

1. Ação (atividade)
2. Controlo de fluxo
3. Nó inicial
4. Nó de término
5. Nó de decisão
6. Condição de guarda
7. Nó de junção
8. Comentário
9. Chamada de Nó de atividade (agrupa um conjunto de ações).



# Concorrência

- 11. Fork divide um fluxo simples em fluxos concorrentes
- 12. Join combina os fluxos concorrentes num fluxo simples
- 13. Envio de sinal envia uma mensagens ou sinal para uma outra atividade ou para um thread concorrente na mesma atividade. É possível enviar dados na mensagem (sinal)
- 14. Receção de sinal



[Fonte: MSDN 2015]

# Fluxo de dados

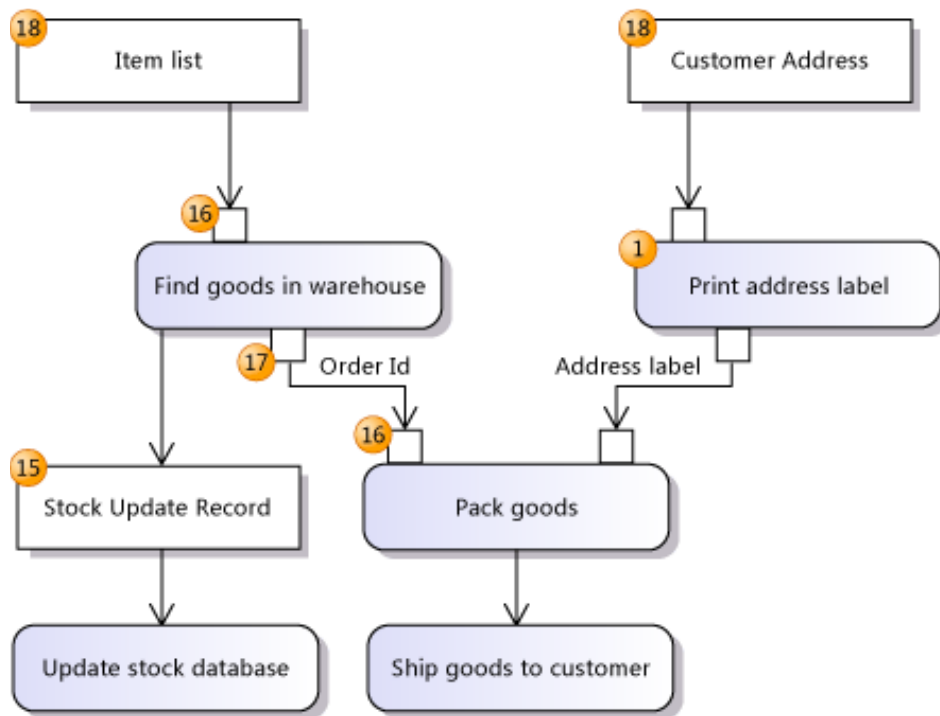
**15. Nó de objeto:** representa informação passada ao longo do fluxo. Pode incluir os seguintes atributos:

- **Ordering** - Quantos são armazenados?
- **Selection** - Invoca o processo que pode estar definido num outro diagrama.
- **Upper Bound** - Se 0 a informação deverá passar diretamente ao longo do fluxo corrente. Se \* a informação poderá ser armazenada no fluxo.
- **Type** - tipo de objetos armazenados e transmitidos

**16. Input Pin** - representa informação que uma ação pode receber quando executada

**17. Output Pin** - representa informação que uma ação pode produzir quando executada

**18. Parâmetro** - um nó do tipo objeto através do qual pode ser recebida e produzida informação



[Fonte: MSDN 2015]