

Projeto de Software

# Contrato de Operações

**Lesandro Ponciano**

2024

# Objetivos

- Apresentar o conceito de contrato de operações
- Contextualizar contrato de operações no contexto de projeto de sistemas
- Apresentar exemplos

# Contratos de Operação

- Contratos de operações podem ajudar a definir o comportamento do sistema
- Eles descrevem a saída da operação do sistema em termos das mudanças de estado dos objetos do domínio
- Estão associados aos casos de uso
  - Fornecem mais detalhes de análise sobre o efeito das operações do sistema implícito ao caso de uso

# Registrando Operações de Sistema

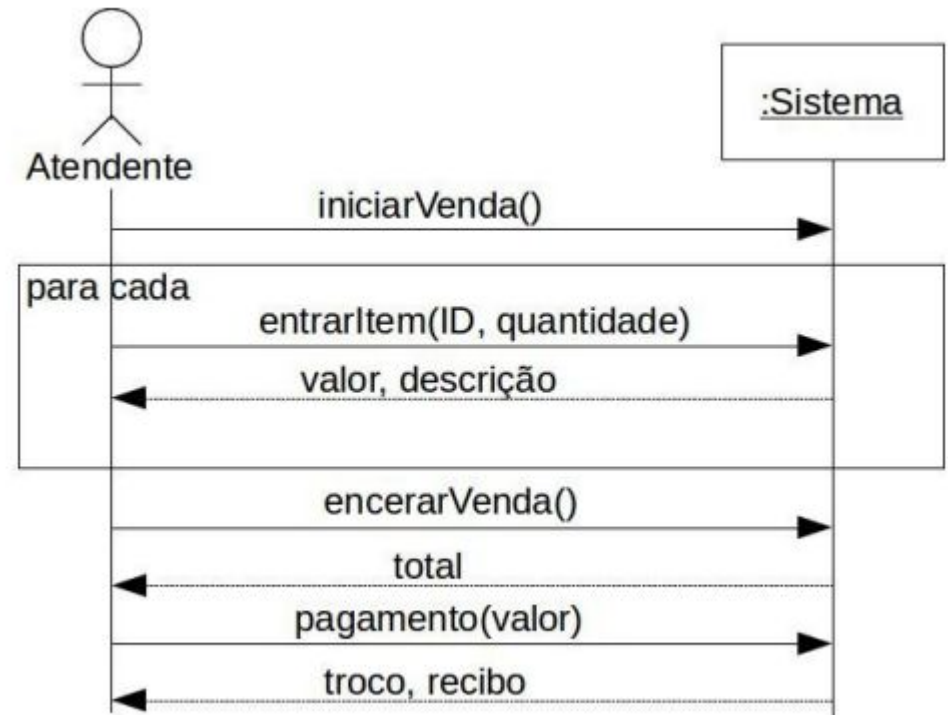
- As **operações do sistema** podem ser identificadas quando se esboça o DSS
  - sistema como uma caixa preta
- Operações de sistema têm o mesmo nome e parâmetros dos eventos que acionam o sistema
- Por exemplo, um evento *entrarItem(ID, quantidade)* corresponde a uma operação do sistema com nome *entrarItem* e com os parâmetros *ID* e *quantidade*.

# Definição do Contrato

- Um contrato é definido para uma **operação do sistema**
- Uma operação do sistema pode ser definida como uma operação que o sistema oferta na sua interface pública para responder a um evento de sistema
  - No caso, o sistema deve ser considerado como uma caixa preta
- O **conjunto de todas as operações de um sistema define a interface pública** do sistema

# Exemplo de Operação

- Eventos de sistema invocam operações de sistema
- O evento de sistema *iniciarVenda()* invoca a operação de sistema *iniciarVenda()*
- E assim respectivamente



# Seções de um Contrato

**Contrato** <<nome da operação>>

- ▣ **Operação:** nome da operação e parâmetros
- ▣ **Referências cruzadas:** Caso de uso onde ocorre
- ▣ **Pré-condições:** Estado dos objetos do domínio do sistema antes da execução da operação
- ▣ **Pós-condições:** Estado dos objetos do domínio do sistema após a execução da operação

# Exemplo de Contrato

## Contrato entrarItem

- ▣ **Operação:** entrarItem(ID: itemID, quantidade: inteiro)
- ▣ **Referências cruzadas:** Caso de uso processar Venda
- ▣ **Pré-condições:** Existe uma venda ativa
- ▣ **Pós-condições:**
  - ▣ Uma instância “iv” de *ItemDeVenda* foi criada;
  - ▣ “iv” foi associada à *Venda* corrente;
  - ▣ “quantidade” foi atribuída a “iv.quantidade”;
  - ▣ “iv” foi associada a um *Produto* através do “ID”.



# Pós-condições

- São observações sobre os objetos do modelo de domínio que se tornam verdadeiras ao término da operação
- Descrevem modificações no estado dos objetos do modelo de domínio, exemplo:
  - Instâncias criadas, associações (ligações UML) formadas ou desfeitas, atributos modificados
- Não são ações a serem executadas
- Apoiam detalhe de granularidade fina e precisão sobre qual deve ser o resultado da operação

# Como escrever uma pós-condição?

- No pretérito perfeito
  - Deixa claro que são observações que resultaram da operação, não uma ação a acontecer
- Exemplos ruins
  - Criar uma *LinhaDeItemDeVenda*
  - Uma *LinhaDeItemDeVenda* é criada
- Exemplo bom
  - Uma *LinhaDeItemDeVenda* foi criada

# Metáfora: "O palco e a Cortina"

- Imagine que o sistema e os objetos estão num palco de teatro
  - 1) Antes da operação ser executada tire uma foto do palco
  - 2) Feche as cortinas e aplique a operação
  - 3) Abra novamente as cortinas e tire uma nova foto
  - 4) Compare as duas fotos e expresse as mudanças no estado do palco

# Como Escrever Contratos?

- 1) Identifique as operações do sistema a partir dos DSSs
- 2) Construa um contrato para as operações do sistema complexas e talvez sutis em seus resultados ou que não estão claras no caso de uso
- 3) Para descrever as pós-condições, use as seguintes categorias
  - Instâncias criadas e excluídas
  - Atributos modificados
  - Associações formadas e desfeitas

# Exemplo: *criarNovaVenda*

<b>Operação:</b>	<code>criarNovaVenda()</code>
<b>Referências Cruzadas:</b>	Casos de Uso: Processar Venda
<b>Pré-Condições:</b>	Nenhuma.
<b>Pós-Condições:</b>	<ul style="list-style-type: none"><li>– Foi criada uma instância <code>v</code> de Venda (criação de instância).</li><li>– <code>v</code> foi associada com Registradora (associação formada).</li><li>– Os atributos de <code>v</code> foram iniciados.</li></ul>

# Exemplo: *entrarItem*

<b>Operação:</b>	entrarItem(idItem: IdItem, qtdade: inteiro)
<b>Referências Cruzadas:</b>	Casos de Uso: Processar Venda
<b>Pré-Condições:</b>	Existe uma venda em andamento.
<b>Pós-Condições:</b>	<ul style="list-style-type: none"><li>– Foi criada uma instância liv da LinhaDeItemDeVenda (criação de instância).</li><li>– liv foi associada com a venda corrente (associação formada).</li><li>– liv.quantidade tornou-se quantidade (modificação de atributo).</li><li>– liv foi associada a uma DescriçãoDoProduto com base na correspondência do idItem (associação formada).</li></ul>

# Exemplo: *finalizarVenda*

<b>Operação:</b>	finalizarVenda()
<b>Referências Cruzadas:</b>	Casos de Uso: Processar Venda
<b>Pré-Condições:</b>	Existe uma venda em andamento.
<b>Pós-Condições</b>	– Venda.estáCompleta tornou-se verdadeira (modificação de atributo).

# Exemplo: *fazerPagamento*

<b>Operação:</b>	fazerPagamento(quantia: Moeda)
<b>Referências Cruzadas:</b>	Casos de Uso: Processar Venda
<b>Pré-Condições:</b>	Existe uma venda em andamento.
<b>Pós-Condições</b>	<ul style="list-style-type: none"><li>– Foi criada uma instância p de Pagamento (criação de instância).</li><li>– p.quantiaFornecida tornou-se quantia (modificação de atributo).</li><li>– p foi associada com a Venda corrente (associação formada).</li><li>– A Venda corrente foi associada com a Loja (associação formada); (para acrescentá-la ao registro histórico de vendas completadas).</li></ul>



# Referências

LARMAN, C.; Utilizando UML e Padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo, Porto Alegre, Bookman, 2007. (Capítulo 11)

BEZERRA, E.; Princípio de Análise e Projeto de Sistemas com UML, Rio de Janeiro, Elsevier, 2007.

## Sistemas Operacionais

**Prof. Dr. Lesandro Ponciano**

<https://orcid.org/0000-0002-5724-0094>