

# Modelagem de Software



---

Prof. Dr. Ronaldo C. de Oliveira

[ronaldo.co@ufu.br](mailto:ronaldo.co@ufu.br)

[www.facom.ufu.br/~ronaldooliveira](http://www.facom.ufu.br/~ronaldooliveira)

FACOM - 2019



---

# Estudo de Viabilidade

# Estudo de Viabilidade

- **Após o levantamento inicial das informações do sistema deve-se verificar as seguintes viabilidades:**
  - **Econômica:** a avaliação do custo de desenvolvimento confrontada com a renda ou benefício últimos derivados do sistema desenvolvido;
  - **Técnica:** estudo das funções, do desempenho e das restrições que possam afetar a capacidade de se conseguir um sistema aceitável;
  - **Legal:** determinação de qualquer interface, violação ou responsabilidade legal que possa resultar do desenvolvimento do sistema;
  - **Alternativas:** avaliação das abordagens alternativas ao desenvolvimento do sistema;



# Estudo de Viabilidade

---

- O estudo de viabilidade deve ser realizado para todo e qualquer sistema, independentemente do tamanho ou da necessidade do mesmo.
- Impasse:  
→ → → **Prosseguir** / **Cancelar o projeto** ← ← ←



# Estudo de Viabilidade Econômica

---

- Análise de custo-benefício;
- Estratégias de renda corporativa longo prazo;
- Impacto sobre outros centros de lucro ou produtos;
- Custo dos recursos necessários ao desenvolvimento;
- Crescimento em potencial de mercado.



# Estudo de Viabilidade Técnica

---

- **Riscos do desenvolvimento:** O sistema pode ser projetado de forma que a função e o desempenho necessário sejam obtidos dentro das restrições reveladas durante a análise?
- **Disponibilidade de recursos:** Existe um pessoal competente à disposição para desenvolver o sistema em questão? Os demais recursos necessários (hardware e software) estão à disposição para a construção do sistema?
- **Tecnologia:** A tecnologia progrediu para um estado que suporte o sistema?



# Estudo de Viabilidade Legal

---

- Contratos comerciais;
- Responsabilidade legal;
- Violações;
- Direitos autorais;
- Infinitude de outras armadilhas muitas vezes desconhecidas do pessoal técnico.



# Análise de Custo Benefício

---

- Delineia os custos para o desenvolvimento do projeto e compara-os com os benefícios tangíveis (diretamente mensuráveis em \$) e intangíveis de um sistema.
- Dificuldades encontradas:
  - critérios que variam de acordo com as características do sistema a ser desenvolvido;
  - tamanho relativo do projeto;
  - retorno sobre o investimento esperado.



# Benefícios Associados aos Sistemas

---

- Automação de processos
- Controle e integridade das informações
- Busca e acesso rápido as informações
- Capacidade de análise e simulações;
- Redução de pessoal
- Aumento de lucratividade
- .....

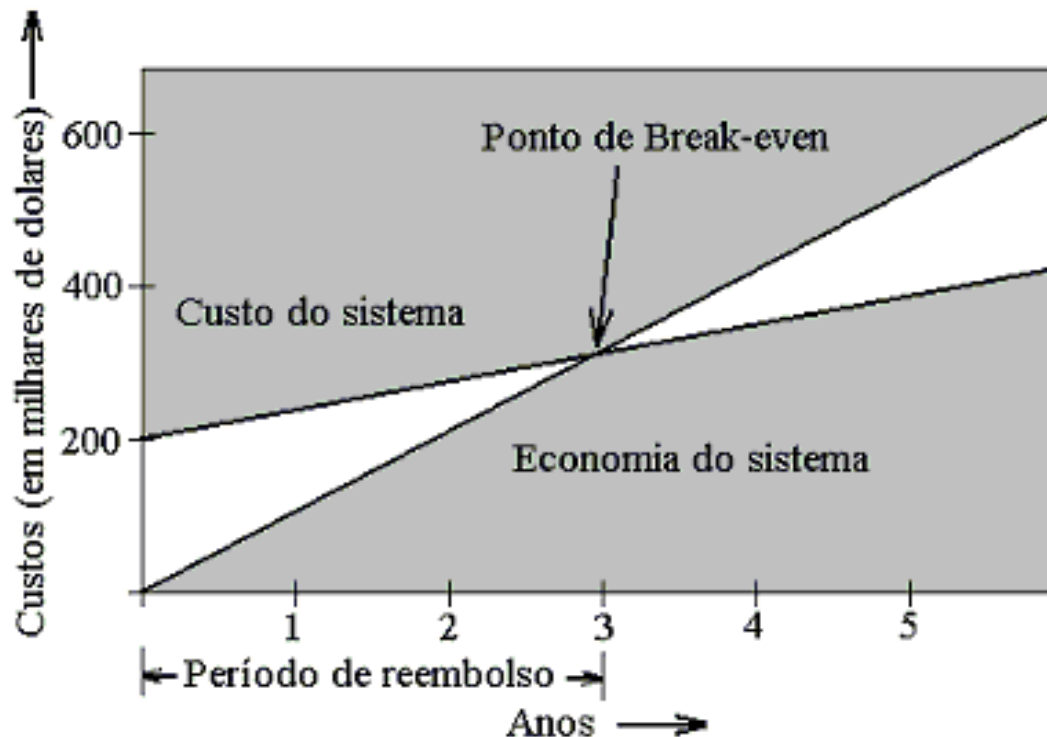
# Custos Associados aos Sistemas

---

- **Custos de aquisição:** consultoria, licenças e compra de hardware ...
- **Custos de iniciação:** instalação, contratação e treinamento de pessoal, mudança cultural ...
- **Custos relacionados ao projeto:** adequações do sistema, coleta de dados, implementação do sistema, migração de dados, preparação de documentação, gerenciamento e desenvolvimento...
- **Custos permanentes:** manutenção, depreciação do hardware, evolução do sistema, pessoal envolvido na administração e operação do sistema ...
- .....

# Análise do Custo/Benefício

- Determinação do Ponto Break-even





---

# Análise e Projeto Orientados a Objetos

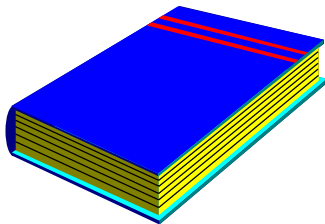
# Análise e Projeto Orientados a Objetos

## ■ O que é Análise e Projeto?

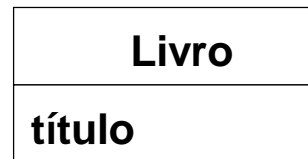
<b>Análise — “o quê”</b> <i>Investigação do problema e dos requisitos</i>	<b>Projeto — “como”</b> <i>Descrição de uma solução lógica</i>
Requisitos Casos de uso Cenários de Casos de Uso Objetos Modelo Conceitual Diagrama de Sequência Restrições Vocabulário ....	Classes de objetos Diagrama de Classes Diagrama de Colaboração Arquitetura Diagrama de Atividades Modelo de Dados Implantação e Operação Interface do usuário ....

# Análise e Projeto Orientados a Objetos

Conceito de domínio



Representação na análise



Representação no projeto



Representação no código

```
public class Livro
{
    public void imprimir();
    private String titulo;
}
```

**Ex : Representação de um Conceito na APOO do objeto “LIVRO” em um sistema de biblioteca**

# Análise e Projeto Orientados a Objetos – Exemplo

---

- Um Exemplo — Jogo de Dados
  - Objetivo: ganha o jogador que rolar dois dados e tirar sete
  - Análise Orientada a Objetos
    - Casos de uso
    - Modelo Conceitual
  - Projeto Orientado a Objetos
    - Diagrama de Colaboração
    - Diagrama de Classes

# Análise e Projeto Orientados a Objetos – Exemplo

- Análise Orientada a Objetos - Jogo de Dados
  - Casos de uso: Descrições narrativas de processos do domínio no formato de prosa estruturada.

Ex.:



**Caso de uso:** Jogar Dados

**Atores:** Jogador

**Descrição:** Este caso de uso começa quando o jogador rola os dados. Se o total dos dados for sete, o jogador ganha; do contrário, ele perde.



# Análise e Projeto Orientados a Objetos – Exemplo

- Modelo conceitual: Conceitos, atributos, e associações que são considerados importantes no domínio da aplicação

Ex.:

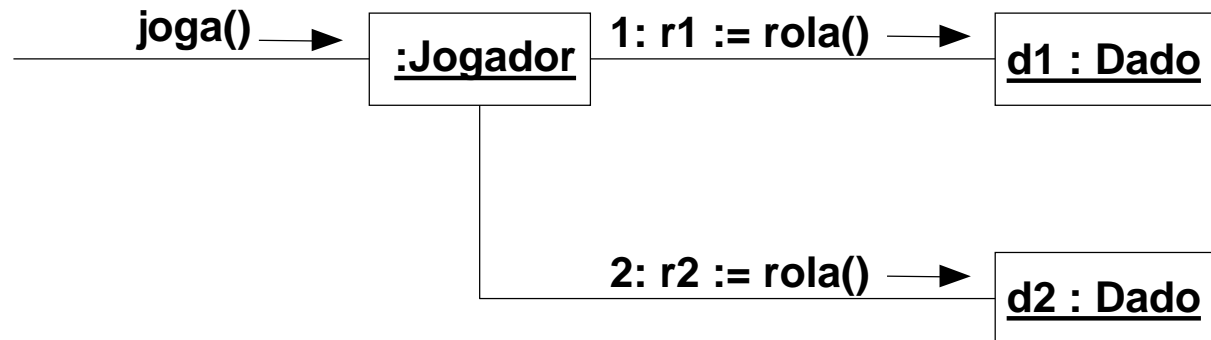


- Um modelo conceitual descreve conceitos do mundo real, não componentes de software!

# Análise e Projeto Orientados a Objetos – Exemplo

- Projeto Orientado a Objetos – Jogo de Dados
  - Diagramas de colaboração
    - Alocação de responsabilidades para objetos ilustrando como eles interagem via mensagens
    - Mostram o fluxo de mensagens entre instâncias e a invocação de métodos

Ex.:

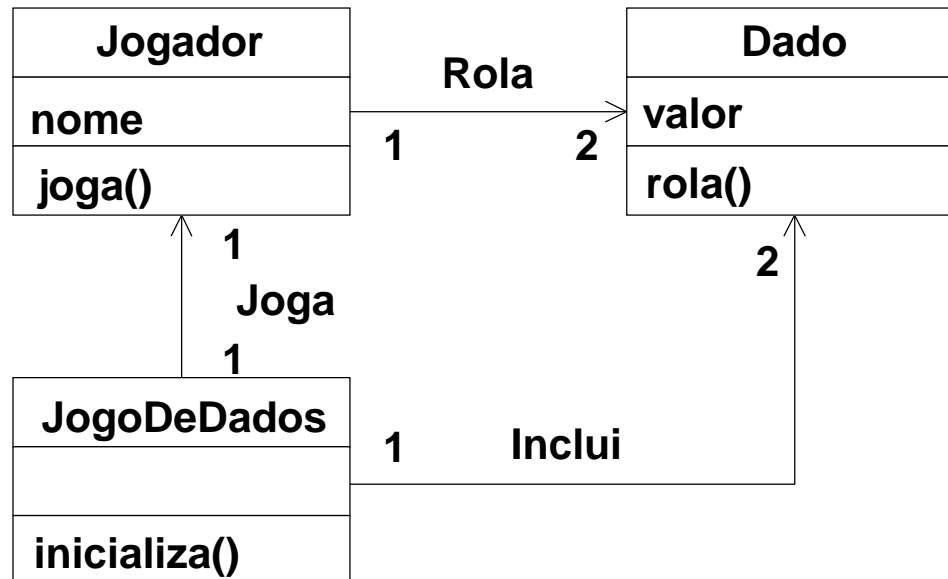


# Análise e Projeto Orientados a Objetos – Exemplo

## ■ Modelagem na APOO (cont.)

- Diagramas de classes de projeto: Como os objetos (de software) se conectam e quais são os métodos de uma classe?

Ex.:



# UML – Linguagem Unificada de Modelagem



---

Visão geral da UML para  
programadores

# Linguagem Unificada de Modelagem - UML

---

- A UML é a linguagem padrão de diagramação para visualizar os resultados da análise e projeto;
- A notação (a própria UML) é relativamente trivial;
- Muito mais importante: habilidade para modelar objetos. Só aprender a notação UML não resolve;

# Linguagem Unificada de Modelagem - UML

---

- Diversas notações para descrição da modelagem orientado a objetos foram proposta nos anos 80 e 90.
- A *Unified Modeling Language* (UML) é uma integração das principais notações.
- Ela define notações diferentes para descrever modelos diversos, produzidos durante a Análise e o Projeto OO.

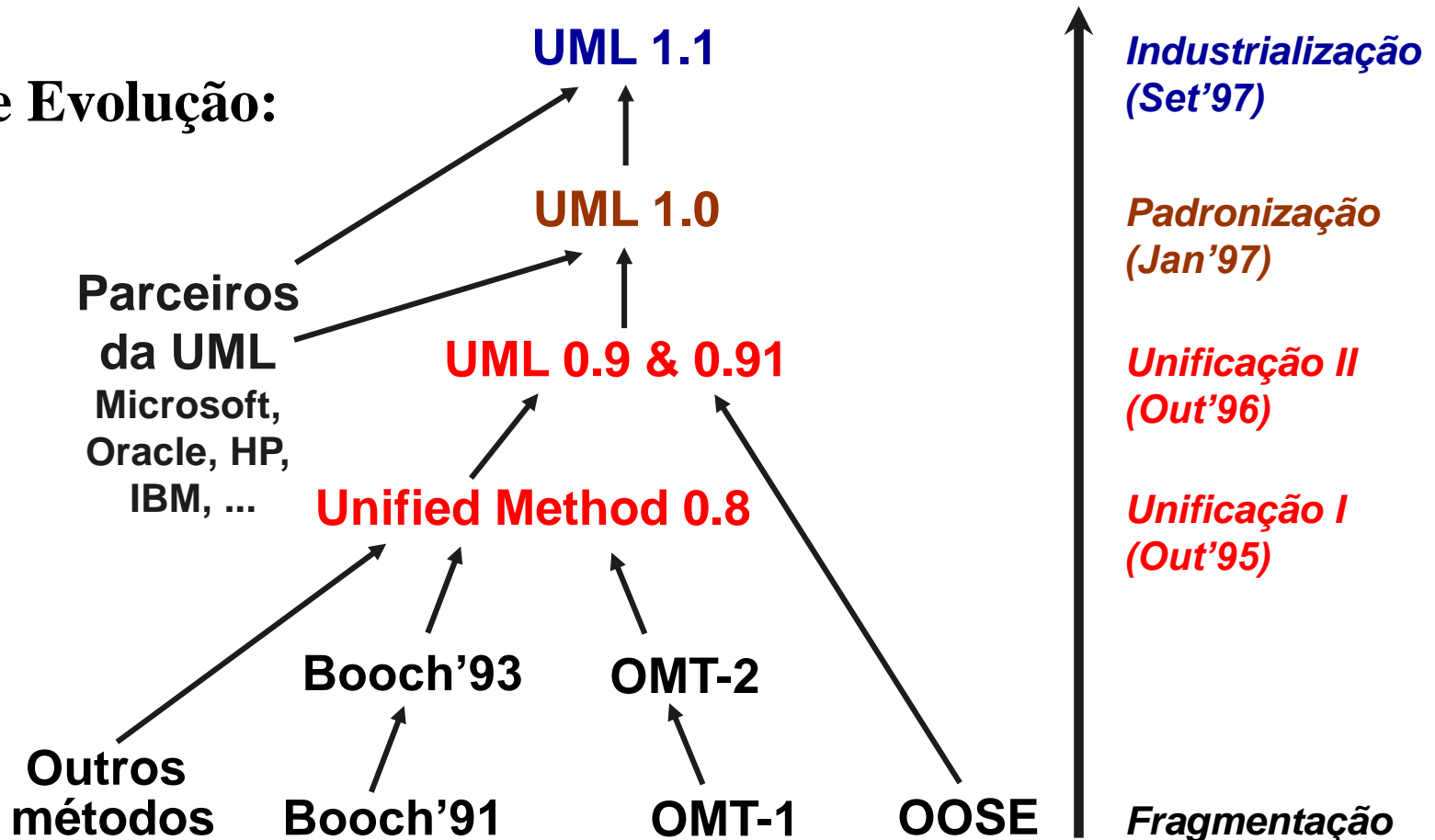
# Linguagem Unificada de Modelagem - UML

---

- O que é UML:
  - É uma padronização da modelagem Orientada a Objetos;
  - Pode ser usada para especificação, construção, visualização e documentação de sistemas de software;
  - Pode ser usado durante todo o ciclo de vida de um software;
  - Pode ser usado com diferentes tecnologias de implementação.

# Linguagem Unificada de Modelagem - UML

Origem e Evolução:





# Linguagem Unificada de Modelagem - UML

---

- Diagramas básicos:
  - **Diagramas de caso de uso:** representação dos processo do ambiente;
  - **Diagrama de classes:** estrutura o relacionamento dos objetos do sistema;
  - **Diagrama de estados:** como o objeto responde aos estímulos, dependendo do estado em que se encontra;
  - **Diagrama de atividades:** representa quando quem faz o quê dentro do sistema;

# Linguagem Unificada de Modelagem - UML

---

- Diagramas básicos:
    - **Diagramas de seqüência:** como os objetos trabalham juntos para realizarem os processos do ambiente;
    - **Diagrama de colaborações:** mostra a interação organizada em torno dos objetos e suas ligações uns com os outros;
- OBS: Estes dois diagramas representam os **diagramas de interação.**

# Linguagem Unificada de Modelagem - UML



- Diagramas básicos:
  - **Diagrama de Pacotes:** representação dos pacotes desenvolvidos e suas relações;
  - **Diagrama de componentes:** documentação da organização física do software, mostrando a estrutura do software;
  - **Diagrama de implantação/distribuição:** especificação do sistema que inclui todos os elementos de processamento, incluindo hardware e software.

# Linguagem Unificada de Modelagem - UML

- Os diagramas da UML

- Três tipos de diagramas: estáticos, dinâmicos e físicos

- Diagrama de Classes → **Modela aspectos estáticos**
  - Diagrama de Casos de Uso
  - Diagrama de Seqüência
  - Diagrama de Colaboração } **Diagramas de Interação**
  - Diagrama de Atividades
  - Diagrama de Estados
  - Diagrama de Pacotes
  - Diagrama de Componentes
  - Diagrama de Utilização } **Diagramas físicos**
- Modelam aspectos dinâmicos**

# Linguagem Unificada de Modelagem - UML

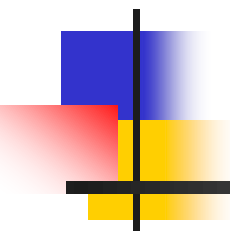
---

- **Análise Orientada a Objetos com UML:**
  - Diagrama de Caso de Uso (com descrições detalhadas);
  - Modelo conceitual (diagrama de classe simplificado);
  - Diagrama de seqüência;
  - Diagrama de Estados de Objetos.

# Linguagem Unificada de Modelagem - UML

---

- Projeto Orientada a Objetos com UML:
  - Diagrama de Classe completo;
  - Diagrama de Colaboração entre objetos;
  - Diagrama de Pacotes;
  - Diagrama de Componentes;
  - Diagrama de Atividades;
  - Diagrama de Implantação/distribuição.



---

# Ferramentas de Modelagem Orientados a Objetos

# Ferramentas Modelagem Orientadas a Objetos

---

- Rational ROSE - [www.rational.com](http://www.rational.com)
- IDE Eclipse (JAVA) - [www.eclipse.org](http://www.eclipse.org)
  - Plug-in UML - [www.omondo.com](http://www.omondo.com)
- STARUML (Open Source) – [www.staruml.com](http://www.staruml.com)
- Enterprise Architect - [www.sparxsystems.com.au](http://www.sparxsystems.com.au)
- Visual Paradigm - [www.visual-paradigm.com](http://www.visual-paradigm.com)
- Argo UML - [argouml.tigris.org](http://argouml.tigris.org)
- Outros.





---

# Diagrama de Caso de Uso

# UML

## Diagrama de Caso de USO

---

- Permitem especificar graficamente quais são os processos do ambiente de negócios que estamos modelando e como eles interagem entre si;
- É muito comum que os iniciantes confundam aspectos do diagrama de casos de uso com Diagrama de Fluxo de Dados (muito usado em modelagem de sistemas estruturados);

# UML

## Diagrama de Caso de USO

---

- O que é um caso de uso:
  - É um serviço que o sistema fornece externamente do ponto de vista do ambiente de negócios;
  - Um caso de uso tem valor agregado em relação ao ambiente. Como consequência, o caso de uso é realizado através de diversas interações entre objetos no ambiente;

# UML

## Diagrama de Caso de USO

---

- Caso de Uso:
  - Descrições narrativas de processos do domínio da aplicação;
  - Documentam a seqüência de eventos de um ator (um agente externo) usando o sistema para completar, do início ao fim, um determinado processo;
  - Representação em UML:



Agendar  
Consulta

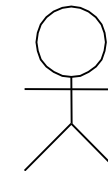
# UML

## Diagrama de Caso de USO

### ■ Atores

- Entidades externas ao sistema que de algum modo participam da estória do caso de uso
  - Estimulam o sistema com eventos de entrada, ou recebem alguma coisa dele
  - Designados pelo papel que desempenham no sistema
    - Ex.: Cliente, Operador, etc.

- Representação em UML:



Cliente

# UML

## Diagrama de Caso de USO

---

### ■ Atores

- Um ator pode ser humano ou artificial.
- Atores podem ser identificados pelos eventos externos ao sistema.
- Um ator pode realizar vários casos de uso.
- Um caso de uso pode ser realizado por vários atores.
- Pode haver hierarquia entre atores.

# UML

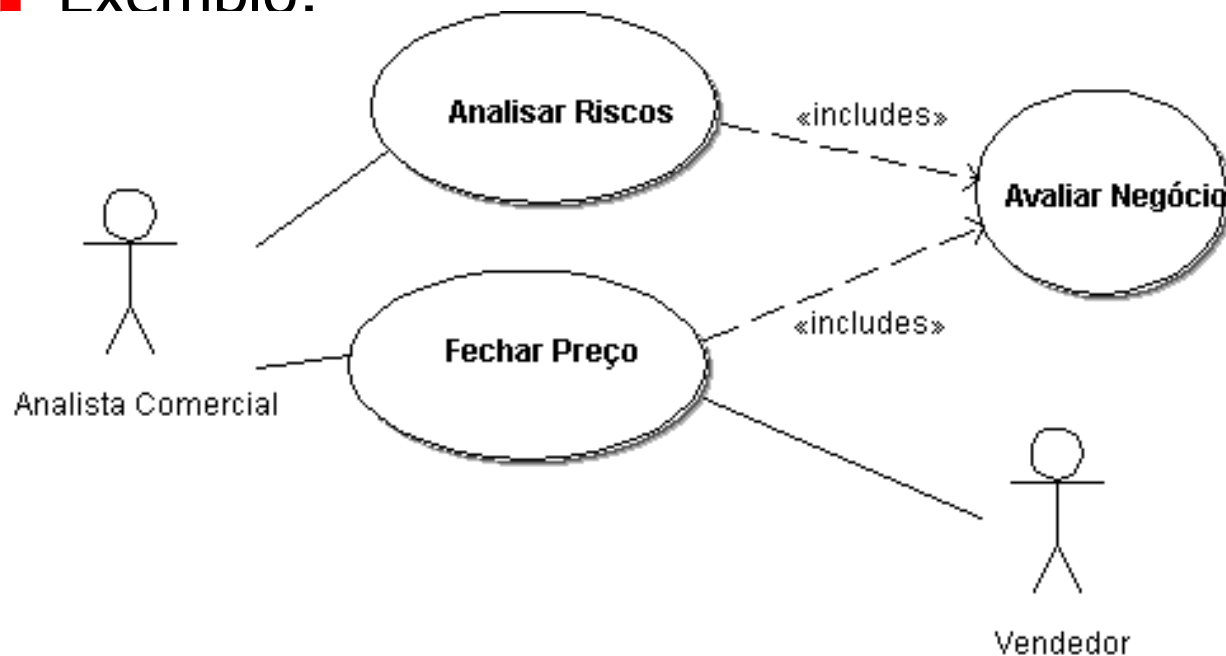
## Diagrama de Caso de USO

---

- Relacionamento
  - Representado por linhas.
  - Corresponde à associação que um UC tem com outro.
  - *Tipos*
    - *Include* (reutilização)
    - *Generalization* (similaridade)
    - *Extend* (caso específico)

# UML Diagrama de Casos de Uso – Relacionamento

- **Inclui (reutilização)**
  - Comportamento semelhante que acontece em mais de um caso de uso
  - Exemplo:





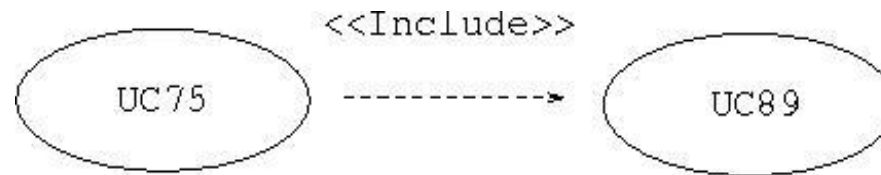
# UML Diagrama de Casos de Uso – Relacionamento

---

- **Include**
  - Tanto **Analisar Riscos** quanto **Fechar Preço** possui em uma parte da descrição do cenário a avaliação do negócio. Portanto o caso de uso **Avaliar Negócio** é incluído nos dois casos de uso anteriores.

# UML Diagrama de Casos de Uso – Relacionamento

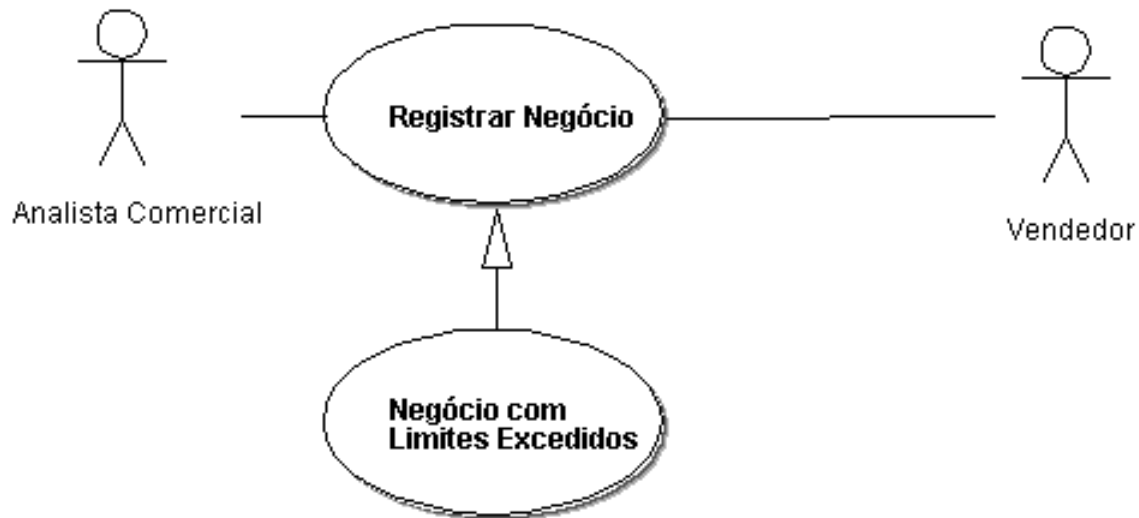
- Dicas para o diagrama



- Use *Include* quando houver “muita” repetição de um mesmo comportamento.
  - Vários passos se repetem em diferentes UCs.
  - UC75 e UC89 sempre são realizados.

# UML Diagrama de Casos de Uso – Relacionamento

- Generalization (similaridade)
  - Caso de uso que descreve um pouco mais que outro caso de uso
  - Exemplo:

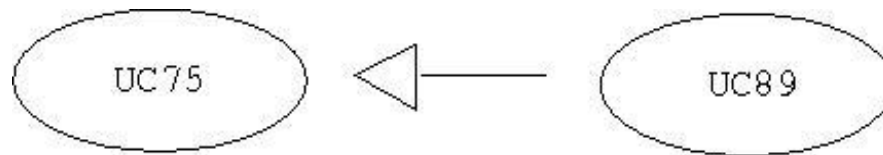


# UML Diagrama de Casos de Uso – Relacionamento

- Generalization (similaridade)
  - No exemplo, o caso de uso básico é Registrar Negócio, no qual tudo funciona bem. Se o limite for excedido, acontece uma alternativa do caso de uso. Esta situação poderia ser simplesmente uma alternativa do caso de uso, porém é diferente o suficiente para merecer um caso de uso separado. O caso de uso especializado pode sobre-escrever qualquer parte do caso de uso básico.

# UML Diagrama de Casos de Uso – Relacionamento

- Dicas para o diagrama



- Use *generalization* quando uma variação de um cenário “normal” ocorre ocasionalmente.
  - Se quer dar um destaque para um cenário alternativo de um UC.

# UML Diagrama de Casos de Uso – Relacionamento

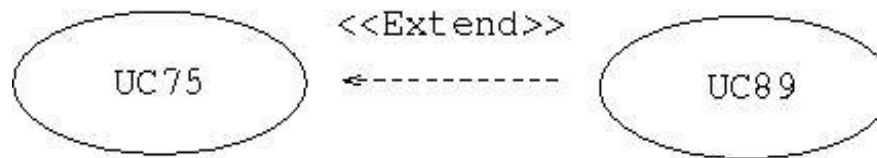
## ■ Extensão

- Semelhante à generalização, com declaração dos pontos de extensão. Um caso de uso pode ter vários pontos de extensão, que são indicados na linha entre os casos de uso do diagrama
- Exemplo:



# UML Diagrama de Casos de Uso – Relacionamento

- Dicas para o diagrama



- Use *Extend* quando uma variação de um cenário “normal” ocorre ocasionalmente.
  - Os UCs são similares, mas o mais específico é “ativado” ocasionalmente.
  - UC75 é realizado e UC89 pode ser realizado também ou não.

# UML

## Diagrama de Caso de USO

---

- O objetivo de um diagrama de caso de uso é descrever diversas visões que um ou outro observador pode ter do ambiente;
- A modelagem do sistema pode incluir vários diagramas de caso de uso;
- Um mesmo caso de uso pode aparecer em diversos diagramas de caso de uso;



# UML

## Diagrama de Caso de USO

---

- Atores e Casos de Uso:
  - Um caso de uso possui um *ator iniciador* que gera o estímulo inicial, e possivelmente vários *atores participantes*
    - O ator iniciador deve ser indicado explicitamente na descrição do caso de uso
  - Algumas categorias típicas de atores incluem:
    - papéis exercidos por pessoas
    - sistemas de computação
    - dispositivos elétricos e mecânicos

# UML

## Diagrama de Caso de USO

---

- Identificando Casos de Uso:
  - Normalmente não são eventos ou passos individuais, mas um processo completo ligado as regras do negócio
    - Erro mais comum!
  - Método baseado em atores
    1. Identificar os atores relacionados com o sistema ou organização
    2. Para cada ator, identificar os processo que eles iniciam ou participam

# UML

## Diagrama de Caso de USO

---

- Identificando Casos de Uso:
  - Método baseado em Requisitos Funcionais
    1. Identificar os RF aos quais o sistema deve responder;
    2. Dar um nome abreviado ao Caso de Uso de acordo com o RF;
    3. Considere que cada resposta ao RF será realizada por um caso de uso;
    4. Relacionar os eventos a atores e casos de uso;
    5. Descreva textualmente os cenários.

# UML

## Diagrama de Caso de USO

---

- Exemplo:
  - Considere um consultório médico. Um evento importantíssimo a se observar é quando um paciente deseja marcar uma consulta.
  - EVENTO:  
**Cliente deseja marcar uma consulta.**
  - Defina o cenário do caso de uso listando todas as respostas que o sistema fornece, tentando definir quem são os atores que participam do caso de uso e quais são as funcionalidades ligadas ao caso de uso.

# UML

## Diagrama de Caso de USO

### ■ Exemplo: cenário do evento

Observe que um ator inicia o caso de uso.

### Cliente deseja marcar uma consulta

- O cliente comunica-se e solicita o agendamento de uma consulta;
- A secretária obtém na agenda o próximo horário disponível e o informa ao cliente;
- Se o cliente aceitar, a secretária solicita o nome e telefone ao cliente e efetua o agendamento;

Diversos objetos compõem o cenário

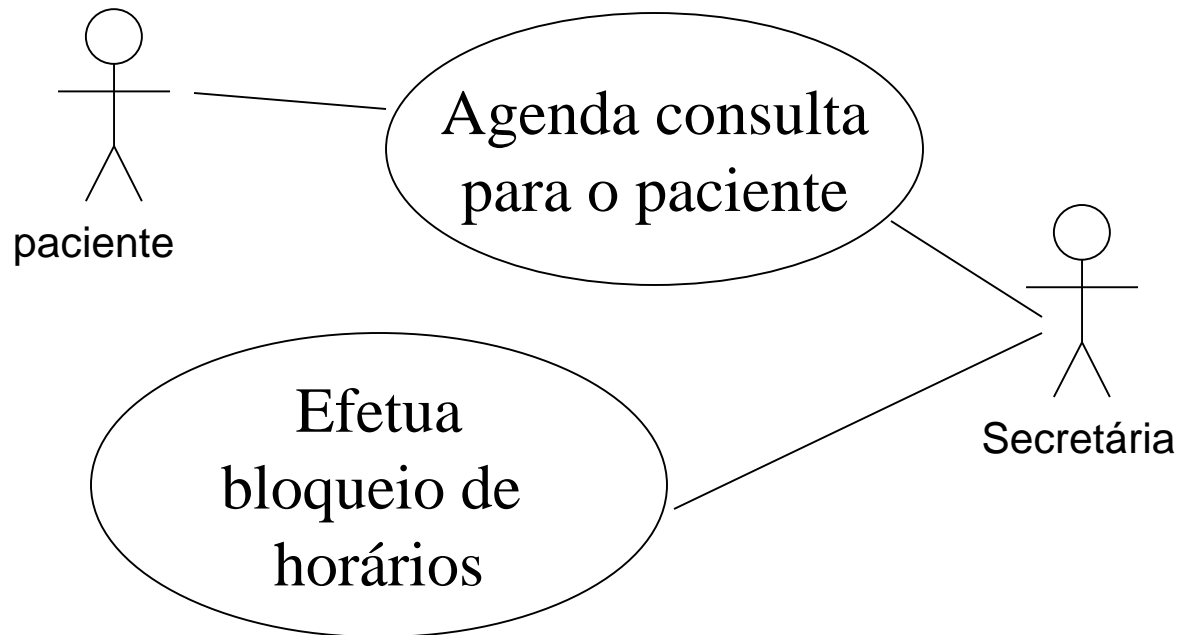
Existem ações bastante ligadas aos objetos.

Eventualmente podemos identificar propriedades de objetos no cenário.

# UML

## Diagrama de Caso de USO

- Conseqüências no modelo



# UML

## Diagrama de Caso de USO

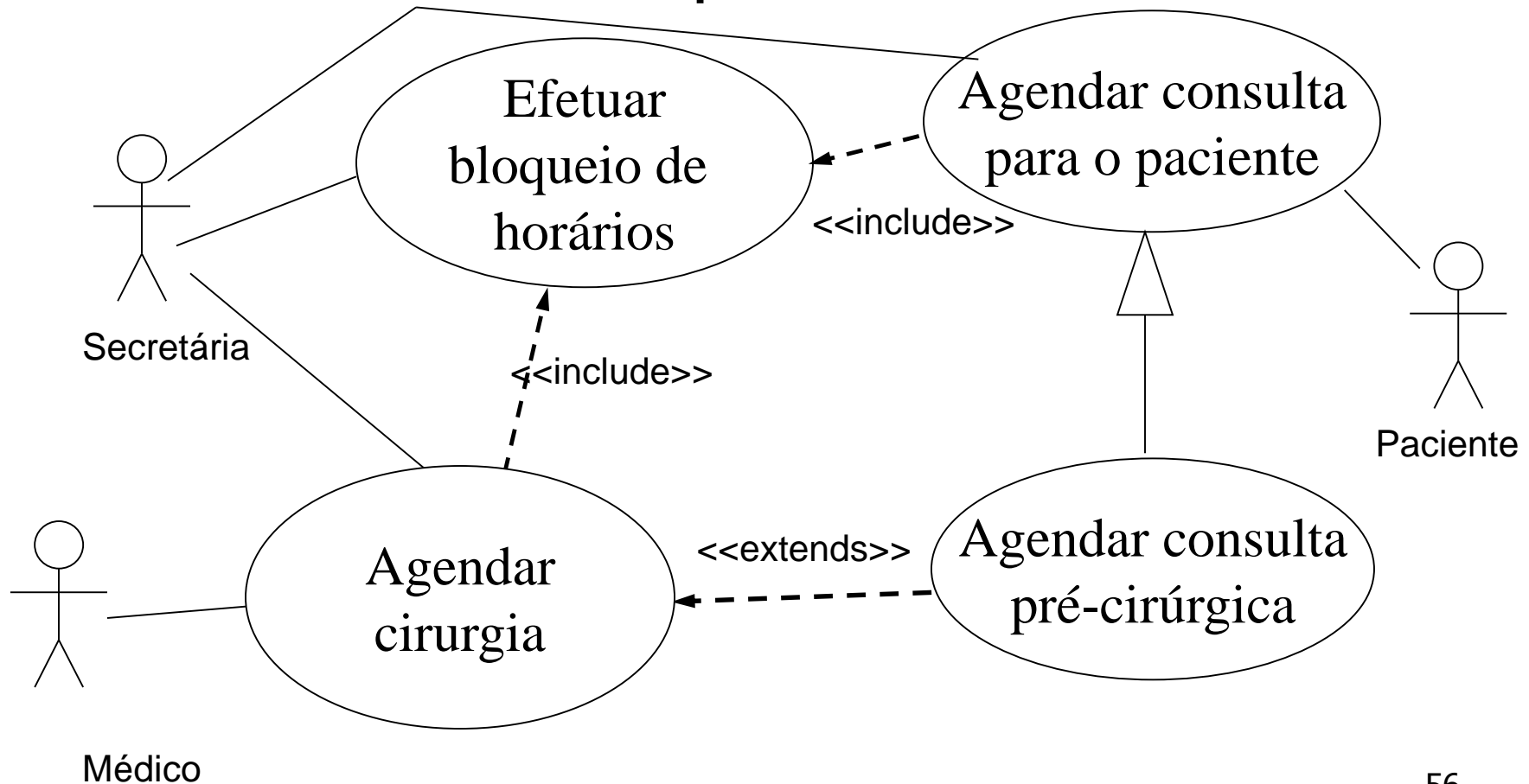
---

- Relações entre casos de uso
  - Imagine que como consequência de uma consulta ou mesmo por uma emergência, o médico identifique a necessidade de realizar uma cirurgia. O agendamento da cirurgia tem várias implicações:

# UML

## Diagrama de Caso de USO

- Possíveis conseqüências no modelo





# UML

## Diagrama de Caso de USO

---

- A inclusão do estereótipo <<include>> no relacionamento dos casos de uso indica que será obrigatório que o caso de uso “agendar consulta pré-cirurgia” acione o comportamento expresso pelo caso de uso “efetuar bloqueio de horários”;
- Se o estereótipo <<extends>> for utilizado, irá indicar um comportamento opcional por um caso de uso, ou seja, o caso de uso “agendar cirurgia” poderá ou não utilizar o “agendar consulta pré-cirurgia”.

# UML

## Diagrama de Caso de USO

---

- Tipos de Casos de Uso:
  - Primário
    - Representam os processos principais ou mais comuns (ex.: *Comprar Itens*)
  - Secundário
    - Representam processos menos importantes ou mais raros (ex.: *Cadastrar Operadores*)
  - Opcional
    - Representam processos que podem ser ignorados ou incluídos em futuras versões do sistema (ex.: *Solicitar Estoque para um Novo Produto*)

# UML

## Diagrama de Caso de USO

---

- Formato dos Casos de Uso:
  - Alto-nível
    - Breve descrição de um processo, normalmente em duas ou três frases, e deliberadamente vago em decisões de projeto
    - Criados na fase inicial de requisitos
  - Expandido
    - Descrição passo a passo dos eventos de um processo
    - Durante a fase de requisitos, apenas os casos de uso mais importantes devem ser escritos nesse formato
  - Real
    - Descrição de um processo em termos de seu projeto real, comprometido com tecnologias de desenvolvimento, interfaces de entrada e saída, etc.



# Casos de Uso – Exemplo

---

- Caso de Uso – Alto Nível
  - Vender item (venda on-line)
    - *“The customer browses the catalog and adds desired items to the shopping basket. When the customer wishes to pay, the customer describes the shipping and credit card information and confirms the sale. The system checks the authorization on the credit card and confirms the sale both immediately and with a follow-up email.”*

# Casos de Uso – Exemplo

## ■ Caso de Uso – Expandido

1. Customer browses through **catalog** and **selects** items to **buy**
2. Customer goes to **check out**
3. Customer fills in **shipping information** (address; **next-day** or **3-day delivery**)
4. System presents full **pricing information**, including shipping
5. Customer fills in **credit card information**
6. System **authorizes purchase**
7. System **confirms sale immediately**
8. System sends **confirming email to customer**

### *Alternative: Authorization Failure*

At step **6**, system fails to **authorize** credit purchase

Allow customer to **re-enter credit card information** and **re-try**

### *Alternative: Regular Customer*

3a. System displays **current shipping information**, **pricing information**, and last **four digits** of credit card information

3b. Customer may **accept** or **override** these defaults

Return to primary scenario at step **6**

# Casos de Uso - Modelo Expandido

---

- Formulário para descrição de caso de uso:
  - Id (código e nome)
  - Descrição
  - Pré-condições
  - Atores
  - Cenário Principal
  - Cenário(s) Alternativo(s)

# Casos de Uso - Modelo Expandido

---

## **UC07 Realizar Locação (RF07)**

### Descrição

Realiza a locação de um ou mais títulos a um cliente.

### Pré-condições

Cliente deve estar cadastrado (UC05).

Título deve estar cadastrado (UC01).

Atores: Cliente, Atendente

# Casos de Uso - Modelo Expandido

---

## UC07 Realizar Locação (RF07)

### Cenário Principal

- 1) O cliente entrega títulos para o atendente.
- 2) O atendente verifica se não existem reservar para s(s) titulo(s) requisitados
- 2) O atendente identifica o cliente no sistema ou o seu dependente
- 3) O atendente cria uma locação para o cliente, com as cópias dos títulos selecionados por este.
- 4) Os títulos locados são colocados como indisponíveis para outras locações
- 5) O sistema estabelece uma data de devolução para cada título, conforme a classificação (lançamento, museu etc) do mesmo.
- 6) O sistema emite comprovante de locação.



# Casos de Uso - Modelo Expandido

---

## UC07 Realizar Locação (RF07)

Cenário Alternativo: cliente com pendências

2.1 Cliente com pendência de multas.

2.1.1 O sistema informa o valor da multa a ser paga.

2.1.2 Se o cliente pagar a multa,  
ir para o passo 3 do Cenário Principal.

Caso Contrário, o UC é encerrado.

# Casos de Uso - Modelo Expandido

---

## UC07 Realizar Locação (RF07)

Cenário Alternativo: cliente com pendências (cont.)

2.2 Cliente com mais de 50% de títulos sem devolução

2.2.1 Emitir mensagem informando que o cliente possui títulos não devolvidos

2.2.2 Encerra o UC.

# UML

## Diagrama de Caso de USO

---

- **Recomendações de Notação**
  - Crie nomes para os casos de uso sempre começando com um verbo
  - Inicie a seqüência de eventos com
    1. **Este caso de uso começa quando <Ator> <inicia evento>**
  - Descreva detalhadamente as ações que serão executadas pelo caso de uso na seção Cenário Principal.
  - Use a seção *Cenário Alternativo* para representar desvios para seqüências de eventos incomuns ou excepcionais;

# UML

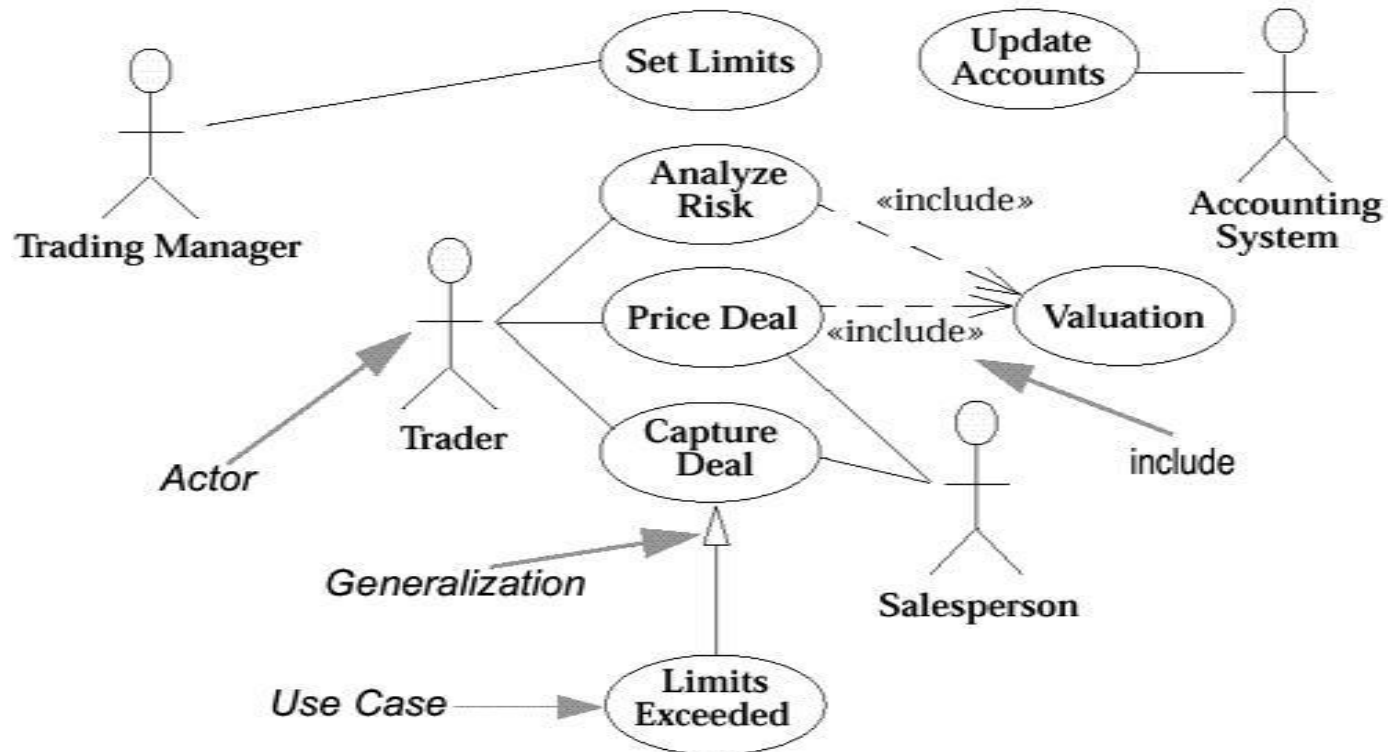
## Diagrama de Caso de USO

---

- Identificando os elementos do cenário
  - Identifique os **objetos/classes** pelos **atores** e **substantivos** que são objetos das ações;
  - Identifique os **serviços dos objetos** pelas **ações** que estão bastante relacionadas a eles (diretamente solicitados ou que promovem a sua manipulação);

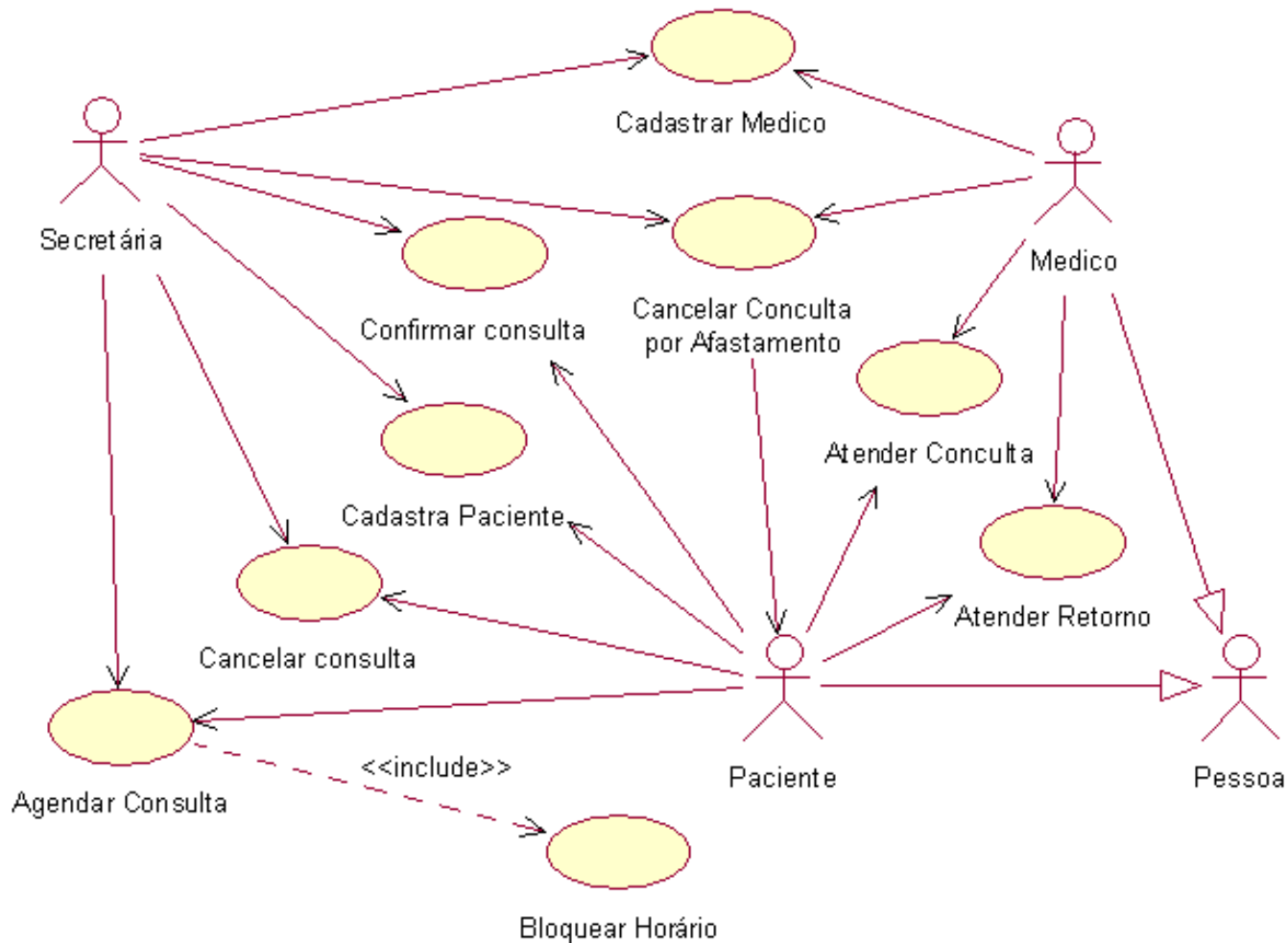
# Diagrama de Caso de Uso Exemplo

- Um exemplo de diagrama



# Diagrama de Caso de Uso

## Exemplo – Diagrama completo





---

# Exemplo de Diagrama de Caso de Uso – Sistema de Locadora de Games

# Sistema de Locadora de Games

- Você foi contratado para desenvolver um sistema para GG Locadora de Games. Nessa locadora, os títulos são classificados em vários gêneros de game. Os títulos estão gravados apenas em DVD e alguns títulos são constituídos por mais de um disco. Um título pode possuir várias cópias. O preço de aluguel de um título varia conforme sua classificação: lançamento, catálogo, clássico e museu. O sistema deve guardar um histórico de títulos por cliente e vice-versa. Um cliente possui um cadastro com seus dados (nome, endereço, ...) e o nome de seus dependentes. Os clientes podem fazer reserva de títulos. Um cliente não tem limite para empréstimo de filmes, mas ele só pode pegar novos filmes após a devolução de 50% dos títulos correspondente à locação anterior. Dependendo da classificação do título, o mesmo possui um prazo para locação (os lançamentos devem ser devolvidos em 24 horas. O sistema deve oferecer informação sobre o aluguel de títulos por gênero, quantas vezes um título foi alugado por cópia. O sistema deve informar também se um título já se pagou. A locação de um filme deve informar quem foi o atendente que a fez. Os títulos em uma locação podem ter diferentes datas de devolução. O sistema deve ser capaz de informar os títulos que estão com atraso de devolução. O sistema deve informar também qual a arrecadação diária da locadora. O sistema deve também controlar a venda de produtos da loja, tais como: balas, chicletes, ...



# Sistema de Vídeo Locadora

## ■ Requisitos Funcionais

1. Administrar\* os títulos do acervo;
2. Administrar\* a carteira de clientes;
3. Administrar\* os funcionários da locadora;
4. Gerenciar gênero dos títulos
5. Gerenciar a classificação dos títulos (preço e prazo de locação);
6. Gerenciar cópias de títulos do acervo
7. Associar vários dependentes a um cliente (titular);
8. Realizar a locação de cópias de títulos a um cliente;
9. Permitir a reserva de títulos por um cliente;
10. Realizar a devolução cópias de títulos por um cliente;
11. Visualizar a data de devolução das cópias dos títulos alugados;
12. Visualizar o histórico dos títulos alugados por um cliente;
13. Visualizar o histórico dos clientes que alugaram um título;
14. Gerar relatório de faturamento diário
15. Administrar\* os produtos da loja
16. Realizar venda de um produto da loja para um cliente
17. ...

# Sistema de Vídeo Locadora

- Requisitos Não Funcionais
  1. O sistema deve ser utilizado via Web.
  2. O sistema deve ser utilizado pelo usuário por meio de um navegador Web (Firefox, IExplorer ou outro compatível com esses navegadores).
  3. O sistema deve possuir um banco de dados relacional (PostgreSQL ou MySQL)
  4. A interface do sistema deve ser fácil de ser utilizada, e o cliente deve ser capaz de aprender a usar o sistema com 2 dias de treinamento
  5. Somente será feita novas locações se o cliente tiver devolvido no mínimo 50% dos títulos da locação anterior
  6. Lançamentos deverão ser devolvidos em 24 horas
  7. Os títulos de DVD´s podem ter mais de um disco

# Sistema de Vídeo Locadora

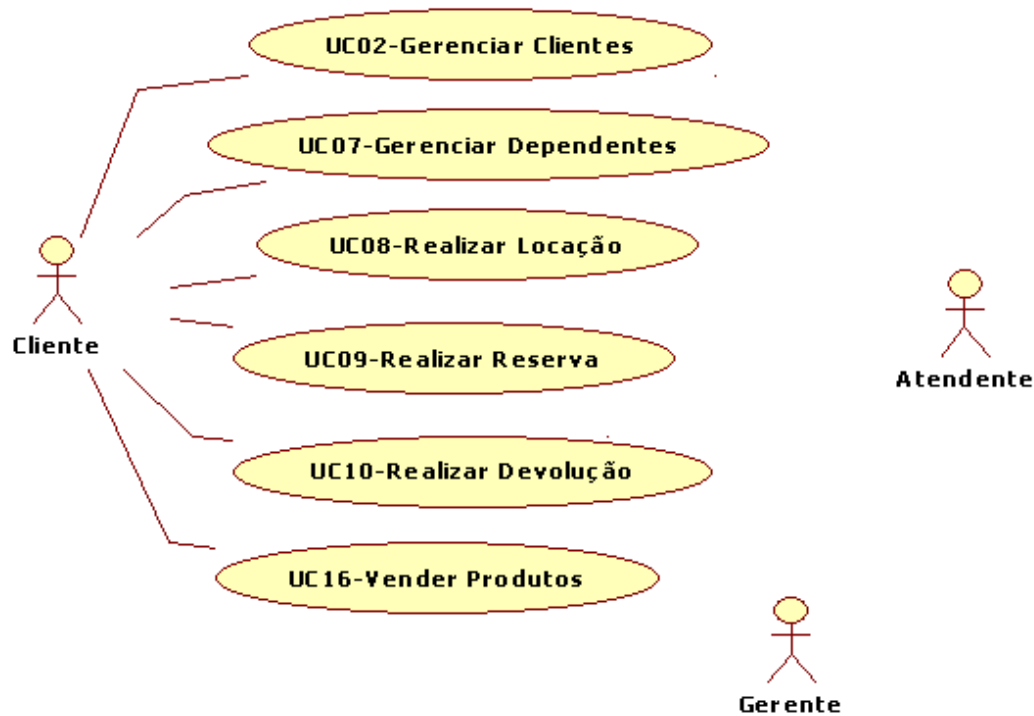
## Identificando Atores do Sistema

---



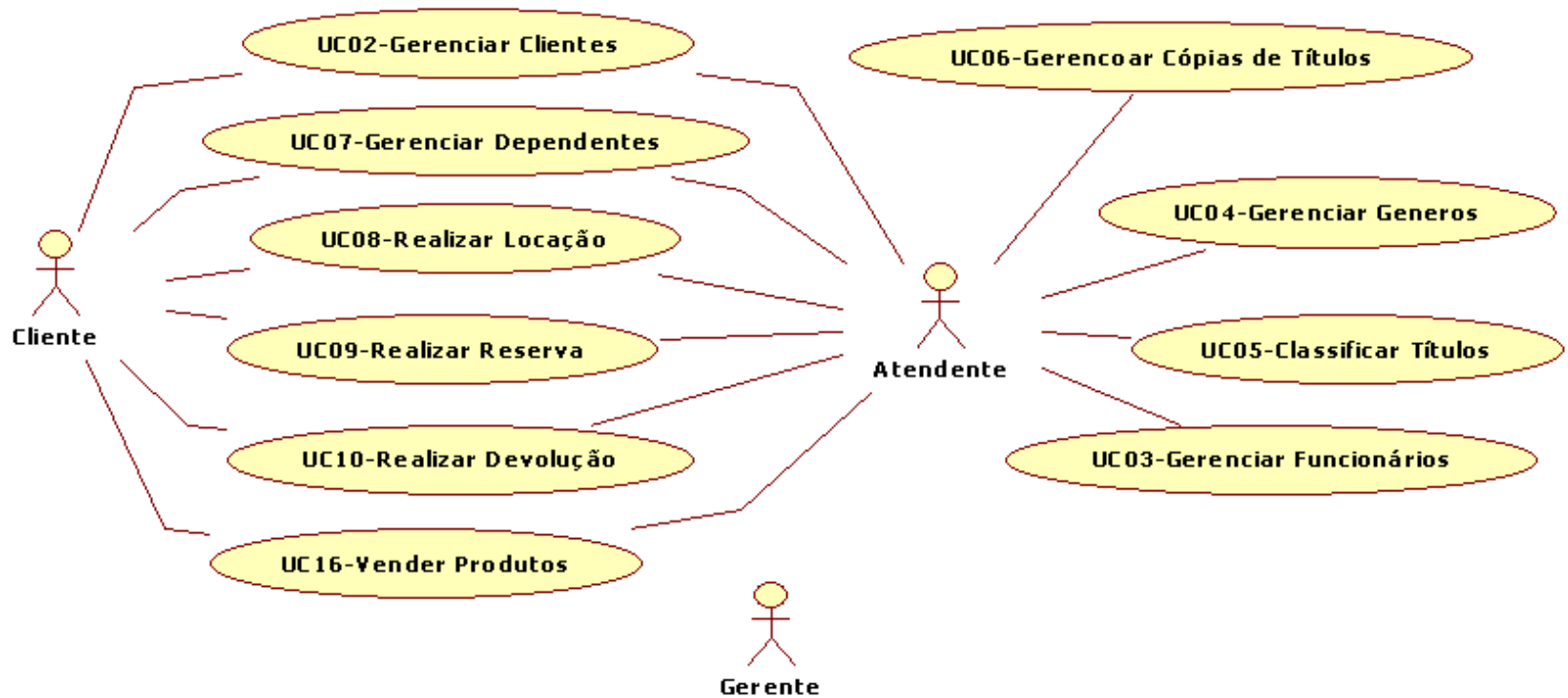
# Sistema de Vídeo Locadora

## UC realizado pelo cliente



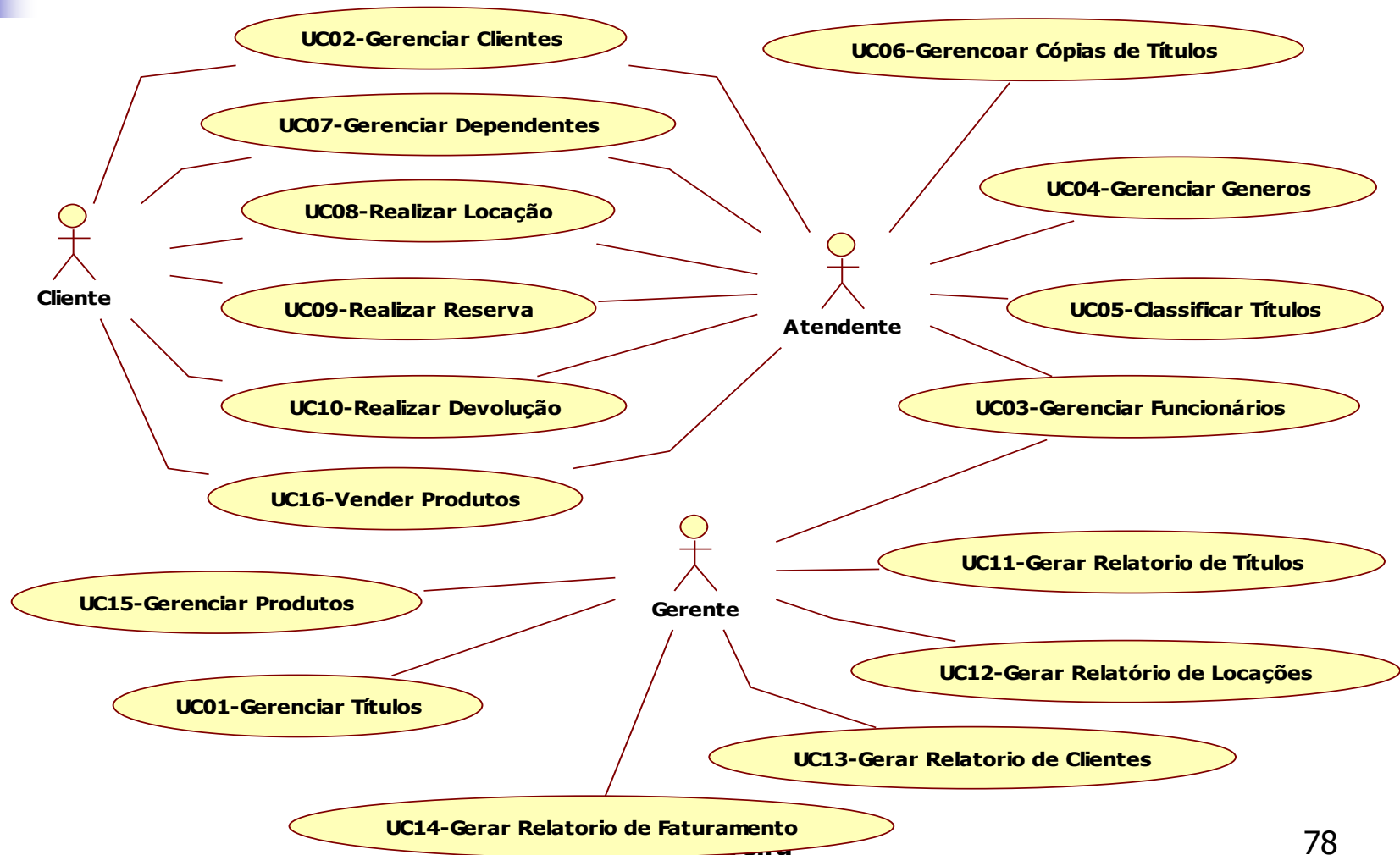
# Sistema de Vídeo Locadora

## UC Realizado pelo Atendente



# Sistema de Vídeo Locadora

## UC realizado pelo Gerente



# Sistema de Vídeo Locadora

## UC01 Gerenciar Títulos (RF01)

### Descrição

Realiza a gerência do acervo da locadora quanto a inserção, remoção de títulos e modificação dados referentes aos mesmos.

**Pré-condições:** ---

**Atores:** Gerente

### **Cenário Principal**

(Inserção)

1. O gerente solicita a inserção de um novo título
2. O sistema lista todos os títulos cadastrados
3. Informar dados de um novo título
4. Validar dados do título
5. Confirmar inserção do título no sistema

(Modificação)

1. O gerente solicita a alteração dos dados de um título
2. O sistema lista todos os títulos cadastrados
3. Informar o código do título a ser modificado
4. Busca o título no sistema e apresenta na tela
5. Modificar os dados necessários para o título
6. Confirma a modificação do título

(Remoção)

1. O gerente solicita a exclusão de um título
2. O sistema lista todos os títulos cadastrados
3. Informar o código do título a ser excluído
4. Busca o título no sistema e apresenta na tela os dados
5. Solicita a confirmação de exclusão do título
6. Confirma a exclusão do título
7. Exclui o título e todas os registros de outras tabelas relacionados a ele

**Cenário Alternativo:** modificação de um título

- 2 Se o código do título não corresponder a um título cadastrado, dar oportunidade para o usuário apresentar novo título ou desistir do caso.

# Sistema de Vídeo Locadora

## **UC08 Realizar Locação (RF08)**

### **Descrição**

Realiza a locação de um ou mais títulos a um cliente.

### **Pré-condições**

Cliente deve estar cadastrado (UC05).

Título deve estar cadastrado (UC01).

**Atores:** Cliente, Atendente

### **Cenário Principal**

1. O cliente apresenta as cópias dos títulos a serem locados.
2. O atendente verifica as pendências do cliente.
3. O atendente cria uma locação para o cliente.
4. O atendente insere cada cópia de cada título como item de locação a essa locação.
5. O sistema estabelece a data de devolução de cada título, conforme classificação e o respectivo tempo de locação.
6. O sistema coloca as cópias locadas como indisponíveis para outras locações
7. O sistema emite comprovante de locação.

**Cenário Alternativo:** cliente com pendências

- 2.1 Se o cliente tiver multa por atraso, o atendente deve solicitar o pagamento da mesma, antes de realizar a locação. Se o cliente não pagá-la, cancelar a locação.
- 2.2 Se o cliente não tiver devolvido 50% dos títulos da última locação, cancelar a nova locação.

**Cenário Alternativo:** Cópia com reserva

- 4.1 Se a título que esta sendo inserida na locação estiver reservado, para a data específica, o sistema informa o atendente e a cópia é retirada da locação.