



# Laboratório de Engenharia de Software I

Turma 01: Profa. Lucia Vilela Leite Filgueiras

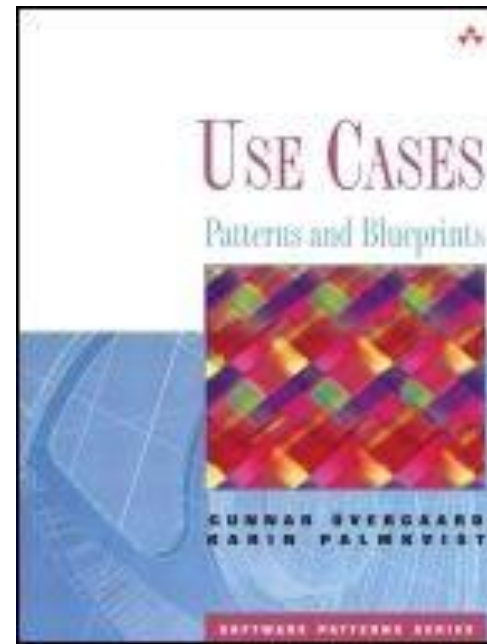
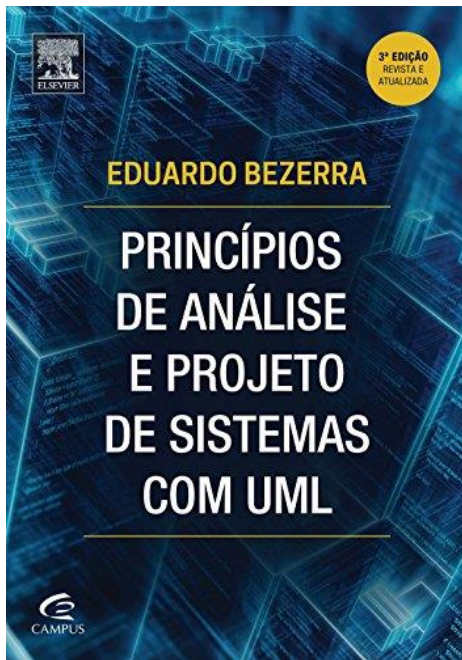
Turma 02: Profa. Maria Alice Grigas Varela Ferreira

Aula 02

Modelo de casos de uso

# Bibliografia para esta aula

BEZERRA, Eduardo. **Princípios de Análise e Projeto de Sistema com UML**. Elsevier Brasil, 2015.



Disponível em versão digital na Amazon

<http://proquest.safaribooksonline.com/0131451340>

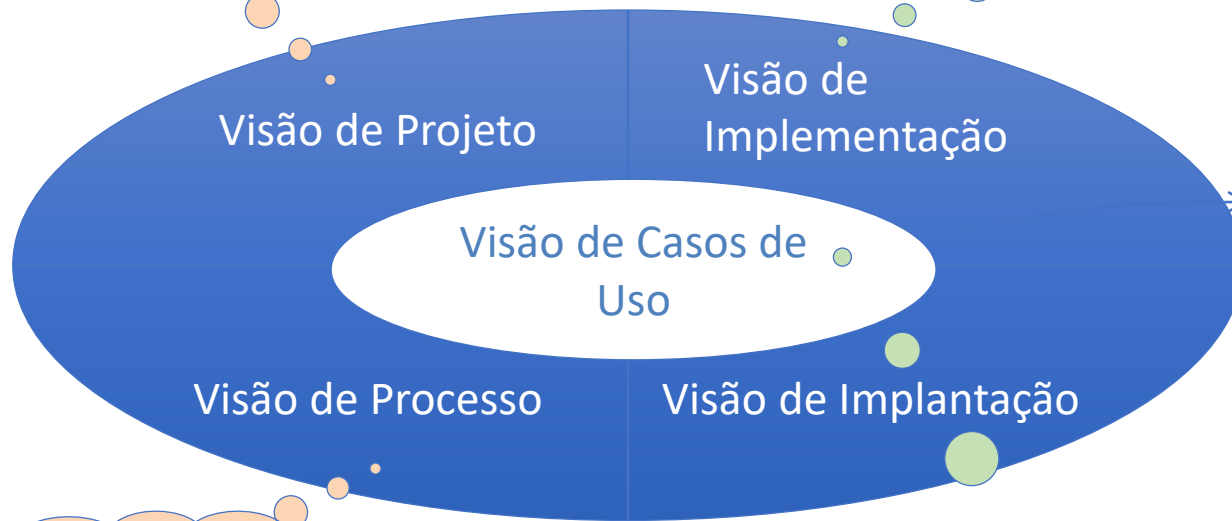
# Roteiro da aula

Horário	Atividade
08:20-08:50	Dicas para elaboração do modelo de casos de uso
08:50-09:20	Apresentação da ferramenta CASE para elaboração do modelo de casos de uso.
09:20-10:00	Desenho do diagrama de casos de uso.
10:00-11:00	Descrição dos casos de uso
11:00-12:00	Apresentação do diagrama e de casos de uso selecionados
Para casa	Revisar e complementar descrição dos casos de uso

# Dicas para elaboração do modelo de casos de uso

características  
estruturais e  
comportamentais

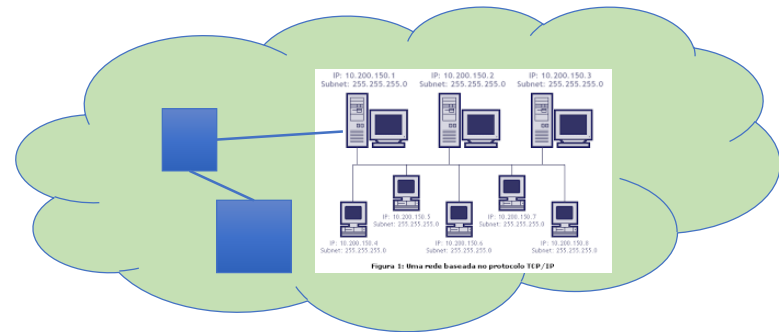
versões e seus  
componentes



interação  
entre  
agentes e o  
sistema



características  
não  
funcionais



# Dicas

1. Casos de uso essenciais x reais
2. Padrões de reuso dos casos de uso
  1. CRUD
  2. Inclusões
  3. Extensões
3. Erros comuns dos casos de uso

Dica 1 - Casos de uso  
essenciais x reais

# Casos de uso essenciais x reais (exemplo do livro do Bezerra)

## Essencial

- Independente de tecnologia.
- Vale a regra dos 100 anos!

- 1) Cliente fornece sua identificação.
- 2) Sistema identifica o usuário.
- 3) Sistema fornece opções disponíveis para movimentação da conta.
- 4) Cliente solicita o saque de uma determinada quantia.
- 5) Sistema requisita o valor da quantia a ser sacada.
- 6) Cliente fornece o valor da quantia que deseja sacar.
- 7) Sistema fornece a quantia desejada.
- 8) Cliente retira dinheiro e recibo e o caso de uso termina.

## Real

- Leva em conta decisões de implementação
- 1) Cliente insere seu cartão no caixa eletrônico.
  - 2) Sistema apresenta solicitação de senha.
  - 3) Cliente digita senha.
  - 4) Sistema valida a senha e exibe menu de operações disponíveis.
  - 5) Cliente indica que deseja realizar um saque.
  - 6) Sistema requisita o valor da quantia a ser sacada.
  - 7) Cliente fornece o valor da quantia que deseja sacar.
  - 8) Sistema fornece a quantia desejada e imprime o recibo para o Cliente
  - 9) Cliente retira a quantia e o recibo, e o caso de uso termina.



Dica 2 - CRUD



# Reuso de casos de uso

---

Maria Alice G. V. Ferreira  
2015



# Reutilização ou Reuso

---

- Reutilização de software é o uso repetido e sistemático de elementos relacionados com software:
  - Produtos: requisitos, arquitetura, componentes, modelos, casos de teste, documentos
  - Processos: desenvolvimento, aquisição, gerência
- Pode-se considerar que o conhecimento e a experiência são também reutilizáveis.



# Reusing Experience

---

“When developing or building a construction of some sort, such as a house or a machine, we all rely on previous experience from similar work. In this way, we will be more productive and will therefore be able to complete a better construction in less time than an inexperienced person. We will know what solutions work, and can thus avoid pitfalls, and we will know or have qualified guesses about possible ways to tackle arising problems [...]”

*Use Cases Patterns and Blueprints  
Gunnar Övergaard; Karin Palmkvist*

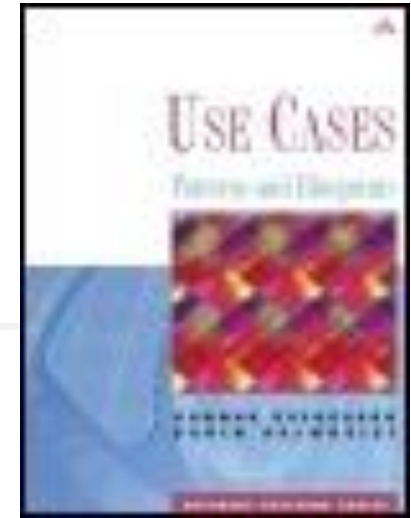


# Principais formas de reuso

---

- Modelos de análise e projeto
  - *Design patterns* – padrões de projeto
  - Componentes (vários modelos: .Net, Java Beans)
- Bibliotecas (ponto flutuante, várias de linguagens)
- Outros Modelos de Padrões
  - Casos de Uso
  - Interfaces padronizadas
- Requisitos

# Padrões para casos de uso



- Use Cases: Patterns and Blueprints  
by Gunnar Övergaard; Karin Palmkvist  
Publisher: Addison Wesley Professional, 2004.
- Acessível pelo Portal de Livros (Safari) com conexão via USP  
Fonte:  
<http://proquest.safaribooksonline.com/0131451340>
- "... the book presents a collection of use-case patterns proven useful when developing maintainable and reusable use-case models. These patterns focus on designs and techniques used in high-quality models, and not on how to model specific usages. Each chapter presents and discusses a group of related patterns...".
- Disponível também através da Biblioteca Genesys.

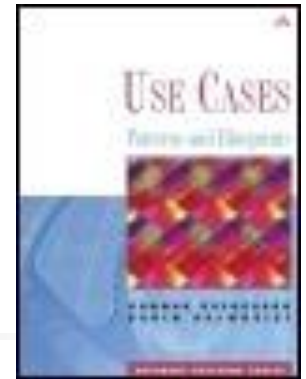


## *Patterns (padrões) e Blueprints*

---

- Um padrão de caso de uso é um desenho (*design*) em geral aprovado em um modelo de casos de uso, juntamente com a descrição do contexto no qual é usado e que consequências sua aplicação provocará no modelo.
- Um *blueprint* (não sei a tradução em português) de caso de uso é um fragmento de um modelo de casos de uso, modelando um uso recorrente do sistema.
- Padrões são descritos de forma padronizada, fazendo uso de uma série de campos pré-estabelecidos.

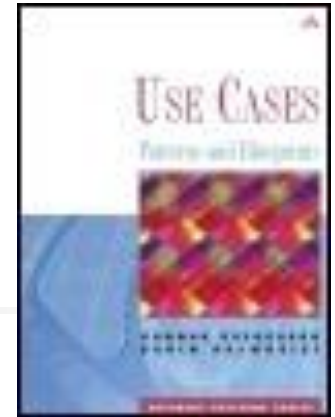
# Padrões para casos de uso: organizado por grupos



- Nome do Grupo – um nome característico do grupo em umas poucas palavras (ex: CRUD – *create, retrieve, update, delete*)
- Intenção – captura o que se pretende ao aplicar um padrão desse grupo
- Características – informa se o padrão é simples ou complexo, comum ou pouco frequente.
- Palavras-chave – uma lista de palavras-chave que caracterizam os padrões.
- **Descrição dos padrões.**
- Discussão – uma discussão compreensível dos padrões do grupo
- Exemplo de uso – um exemplo onde um ou mais padrões são aplicados, incluindo a descrição do caso de uso.
- Modelo de Análise – Um modelo de classes, independente de plataforma, que apresenta a realização do caso de uso.

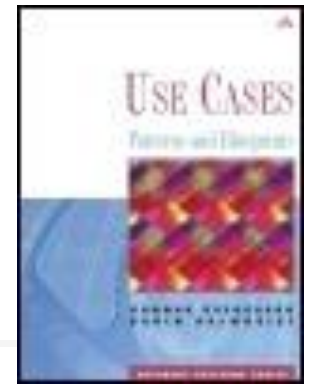


# Padrões para casos de uso: descrição de um padrão

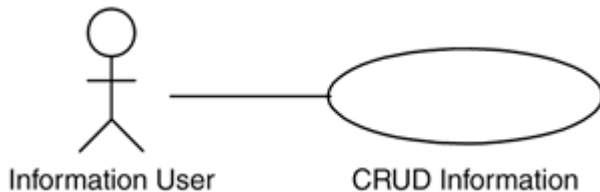


- Para cada padrão:
  - Nome – um nome descritivo para o padrão
  - Modelo – Um modelo de caso de uso do padrão
  - Descrição – do modelo de caso de uso
  - Aplicabilidade – estados em que o padrão pode ser usado
  - Tipo – indica se o padrão afeta a estrutura do modelo de casos de uso ou a descrição de um caso de uso.

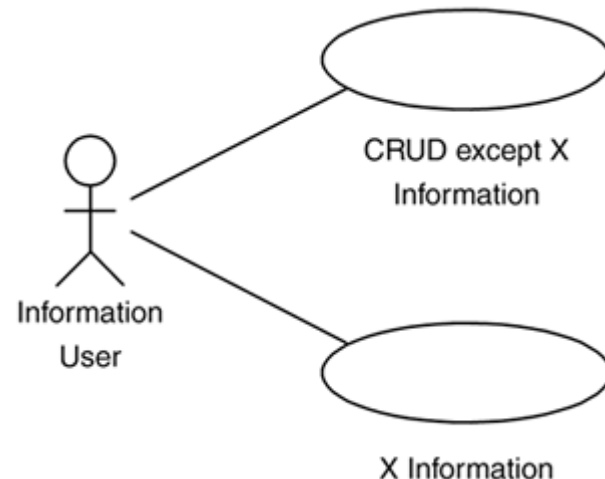
# Exemplo: CRUD



- Intercala casos de uso simples, como Criação (create), Recuperação (retrieve), Atualização (update) e Exclusão (delete) de informação, num único caso de uso, formando uma unidade conceitual.
- Modelo de casos de uso

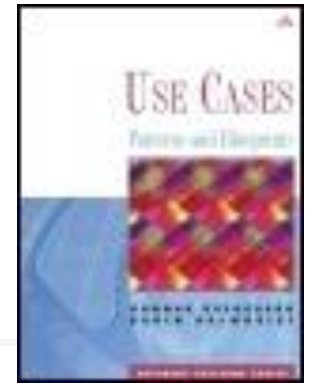


**CRUD Completo**



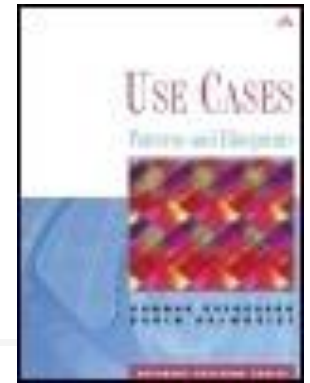
**CRUD Parcial**

# Exemplo: CRUD



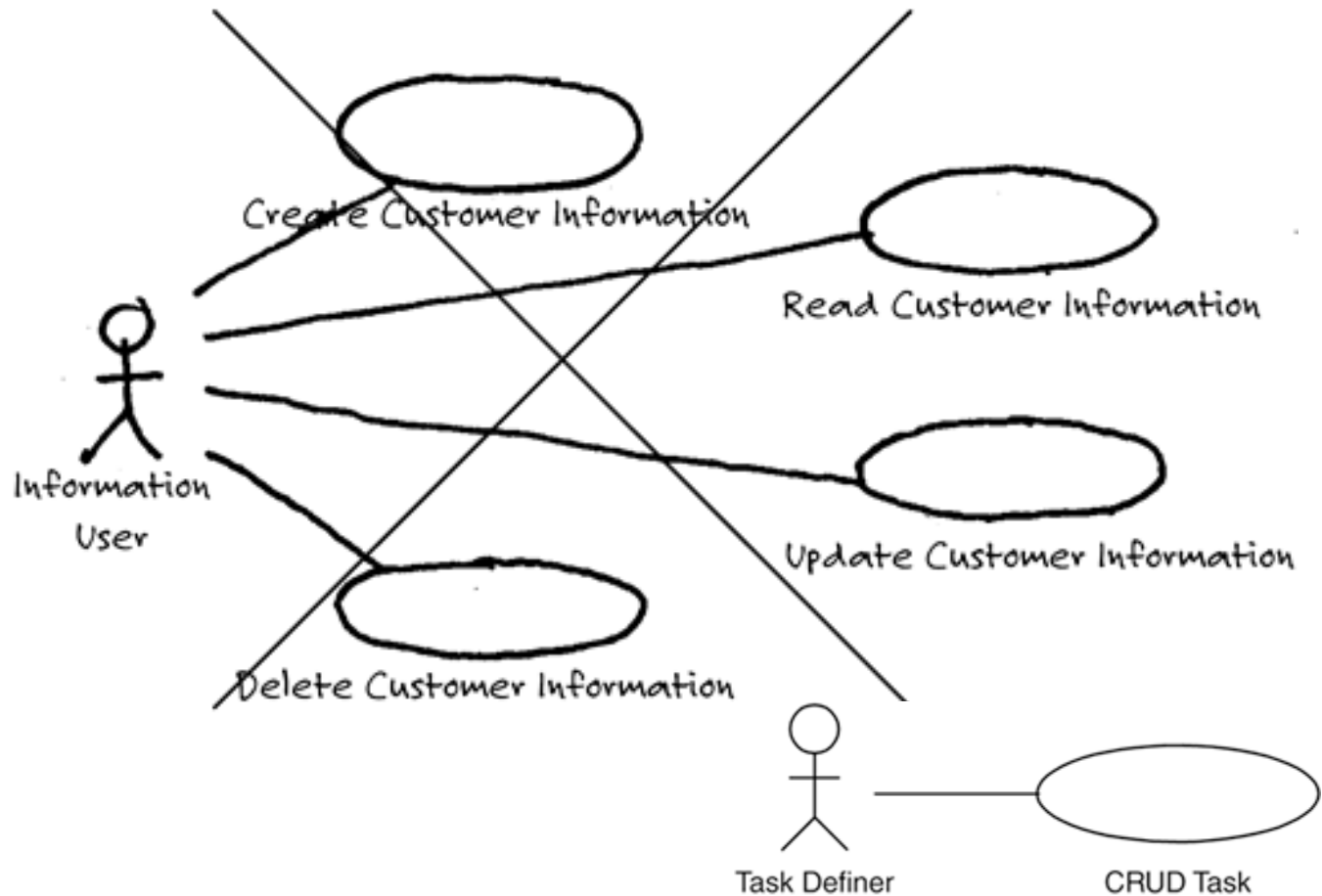
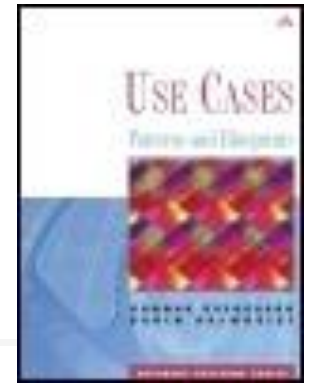
- **Aplicabilidade:**
  - **Completo:** Esse padrão pode ser usado quando todos os fluxos contribuem com o mesmo valor de negócio e todos são curtos e simples.
  - **Parcial:** Esse padrão é preferível quando uma das alternativas do caso de uso é mais significativa, mais extensa ou muito mais complexa do que as demais alternativas.
- **Tipo:** Estrutural (o termo estrutural tem a conotação de estar ligado ao diagrama de classes → a estrutura de classes).

# Exemplo: CRUD

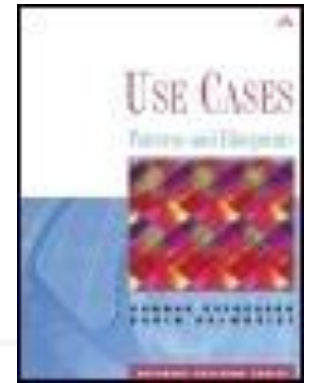


- **Discussão:** Frequentemente, os sistemas manipulam informação que, do ponto de vista do sistema, é facilmente criada no sistema. Após uma simples verificação de sintaxe ou verificação de tipo de dado, e talvez, alguns cálculos triviais ou verificação de regras de negócio, a informação é armazenada. Nenhum cálculo ou verificação mais avançado necessitam ser efetuados. A descrição dos fluxos é realizada em poucas sentenças, sem necessidade de mais do que um ou dois fluxos alternativos. A recuperação, atualização e exclusão da informação são operações igualmente simples.
  - Mesmo sendo operações simples, devem ser incluídas no modelo de casos de uso
  - O valor de cada operação é muito pequeno; é o conjunto delas que lhes confere valor.

# Exemplo: CRUD

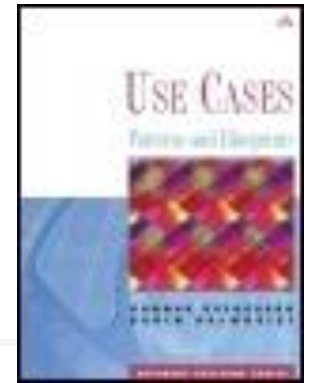


# Exemplo: CRUD



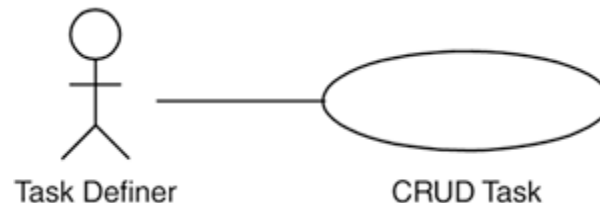
- Um caso de uso CRUD pode incluir outros fluxos básicos como por exemplo a busca por um item ou a realização de algum cálculo simples sobre algum item.
- É importante não inserir operações complexas. Essas operações devem permanecer como casos de uso isolados porque certamente serão desenvolvidos, revisados, projetados e implementados separadamente.

# Exemplo: CRUD

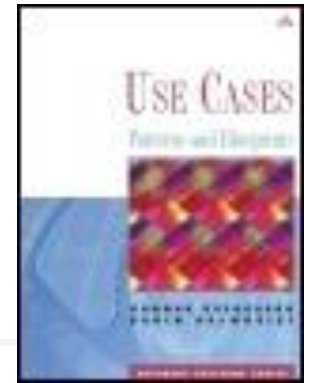


- **Exemplo:** Nessa seção se apresenta um exemplo de uso de CRUD Completo. Deseja-se modelar o registro de uma nova tarefa a ser realizada dentro de um esquema de projeto. Deseja-se também modificar tarefas registradas mas ainda não executadas, cancelar tarefas, bem como visualizar a lista de tarefas ainda não realizadas ou que falharam na sua execução.
- Como se pode ver na descrição completa do caso de uso, as quatro diferentes alternativas são muito simples e podem ser expressas como quatro fluxos básicos, nenhum sendo superior aos demais.

- **Modelo:**



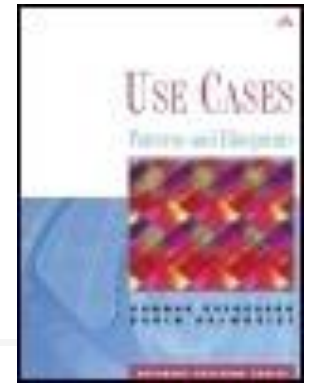
# CRUD Tarefa



- **Descrição:** O caso de uso registra, modifica ou cancela informação referente a uma tarefa, conforme solicitado pelo Task Definer (Task Definer é o papel encarregado de estipular na empresa o que cada participante deverá fazer → chefe) .
- **Fluxo normal:** o caso de uso tem quatro fluxos: registrar tarefa, modificar tarefa existente, cancelar tarefa e visualizar as tarefas que falharam.
  - **Registrar tarefa**
    - 1) o sistema apresenta ao Task Definer uma lista de tipos de tarefas
    - 2) Task Definer seleciona o tipo da tarefa, fornece um nome para ela e a data em que deve ser realizada.
    - 3) O caso de uso verifica a validade da data e a unicidade do nome fornecido.
    - 4) A tarefa é registrada no sistema.
    - 5) O caso de uso é encerrado.

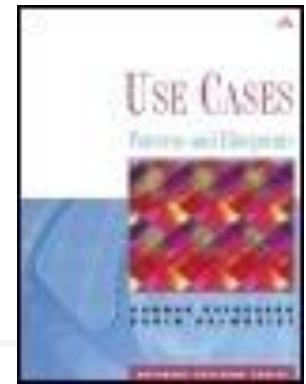


# CRUD Tarefa



- Modificar tarefa existente
  - 1) Task Definer decide modificar uma tarefa existente
  - 2) O sistema recupera e exibe o nome de todas as tarefas que não estão marcadas como ativas.
  - 3) Task Definer seleciona uma das tarefas e o sistema recupera a informação sobre ela e a exibe ao Task Definer
  - 4) Task Definer altera as informações apresentadas; o nome não pode ser alterado
  - 5) Task Definer aceita a informação. O sistema verifica a validade da data e armazena a informação no repositório.
  - 6) O caso de uso é encerrado.

# CRUD Tarefa



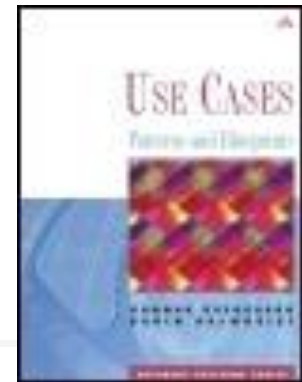
- Cancelar Tarefa

- 1) Task Definer decide cancelar uma tarefa.
- 2) O sistema visualiza todas as tarefas armazenadas e que não se encontram ativas.
- 3) Task Definer seleciona uma tarefa e o sistema apresenta a informação sobre ela.
- 4) Task Definer confirma seu cancelamento;
- 5) O caso de uso remove a tarefa do repositório.
- 6) O caso de uso termina;

- Ver tarefas que falharam

- 1) Task Definer solicita a lista das tarefas que falharam.
- 2) O caso de usos seleciona todas as tarefas que falharam e exhibe seus nomes.
- 3) O caso de uso é encerrado.

# CRUD Tarefa

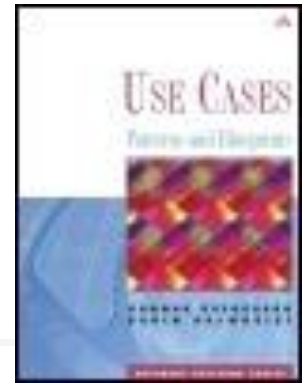


- Fluxos alternativos:
  - **Operação cancelada:** Task Definer decide cancelar a operação a qualquer tempo; a operação é cancelada e os dados fornecidos são descartados. O caso de uso é encerrado.
  - **Nome incorreto e tempo inválido:** Se o nome fornecido para a tarefa não é único ou a data é inválida, Task Definer é notificado e solicita-se que ele forneça uma nova data ou nome. Task Definer reentra essa informação e o fluxo recomeça na posição onde se deu o erro.
- **Modelo de análise:** não será abordado no momento, mas pode ser encontrado no livro.



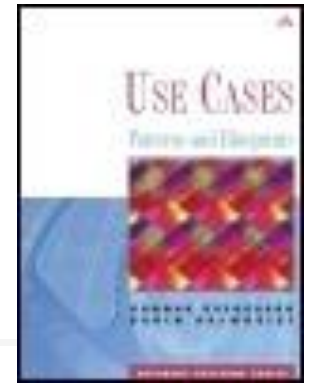
# *Blueprints*

---



- São organizados de forma parecido ao dos padrões.
- Exemplos:
  - Controle de Acesso: O sistema deve incluir algum tipo de segurança de acesso. O acesso à informação no sistema ou aos serviços do sistema é ditado por direitos de acesso específicos concedidos ao usuário individual.
  - Sistemas legados: O sistema deve incluir ou fazer uso de um sistema já existente.
  - Login e Logout: Os usuários necessitam se registrarem ou se identificarem antes de usarem os serviços oferecidos pelo sistema.

# Antipadrões (*mistakes*)

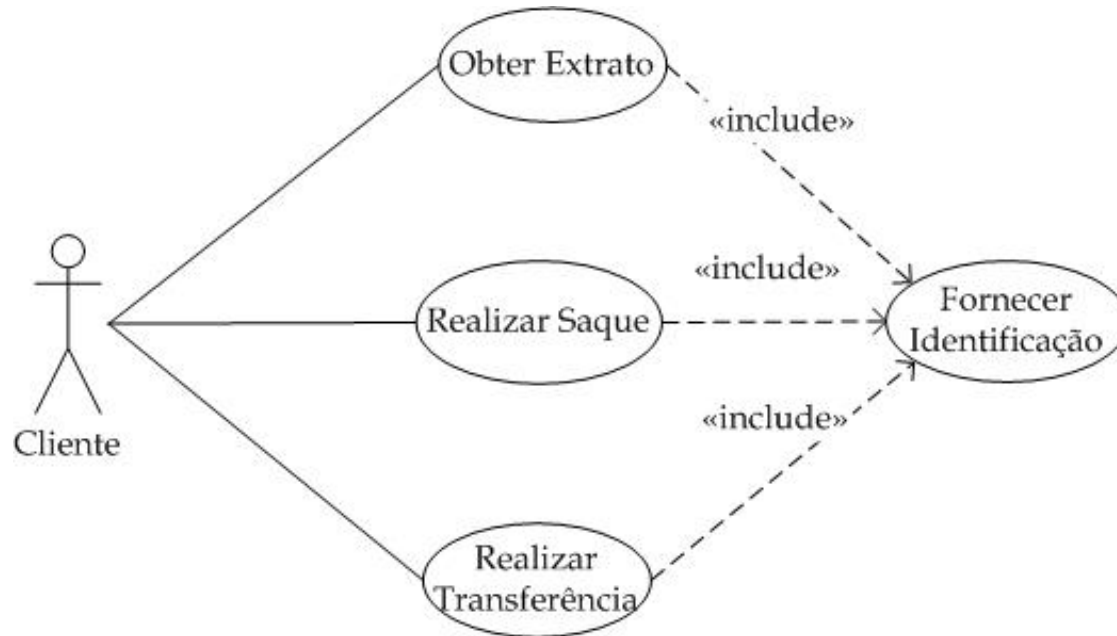


- Situações que devem ser evitadas e que podem ser usadas para detectar alguma coisa que pode estar errada num modelo. Apesar de raros, sempre existem boas razões para aparecerem erroneamente nos modelos. Os antipadrões auxiliam a correção dos modelos.
- Exemplos:
  - Fluxos alternativos modelados como extensões – até o nome (em português) auxilia a fazer essa confusão
  - Modelos de negócio modelados como casos de uso.

# Dica 3 – Inclusões e extensões

# Inclusão (include)

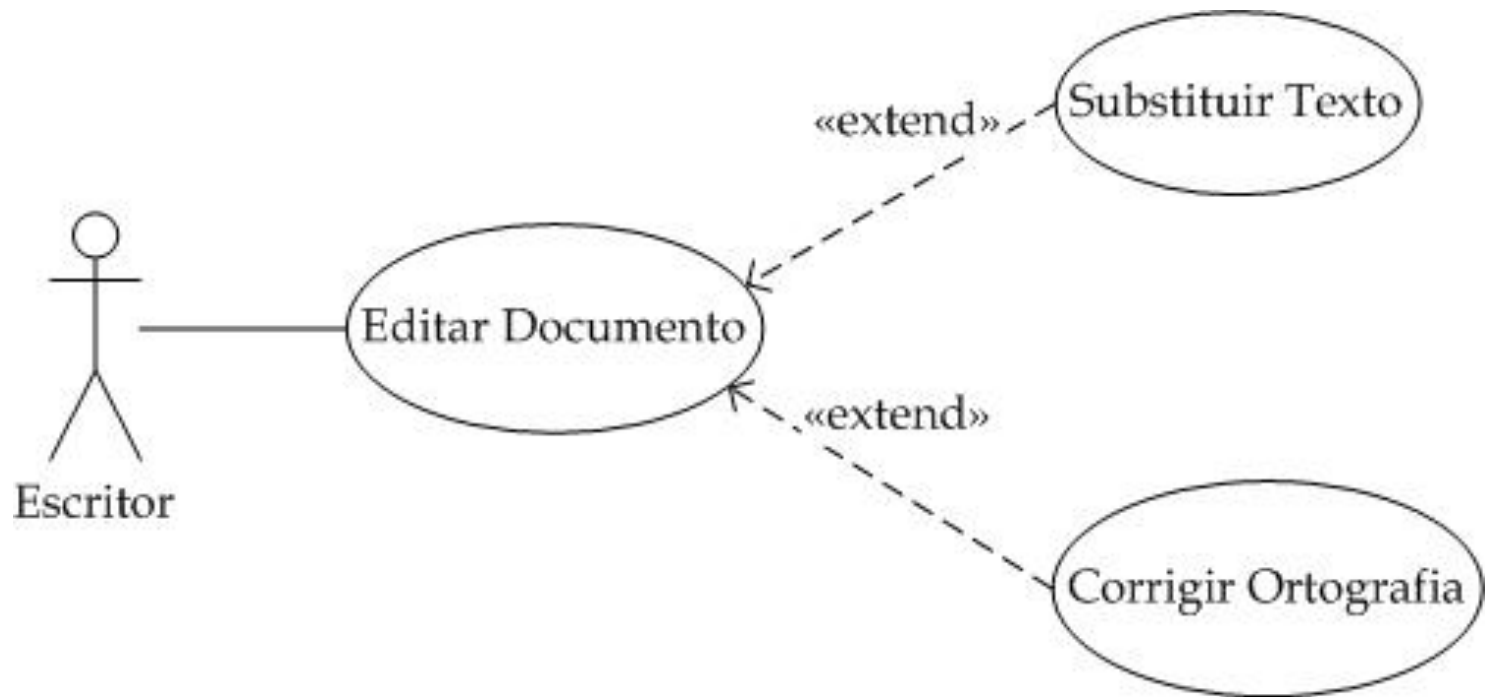
Exemplo:



Referência no texto do caso de uso inclusor:

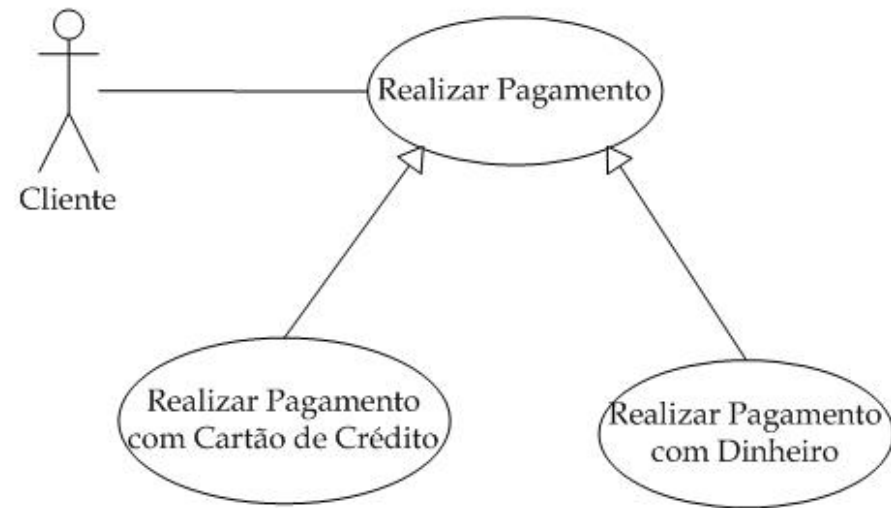
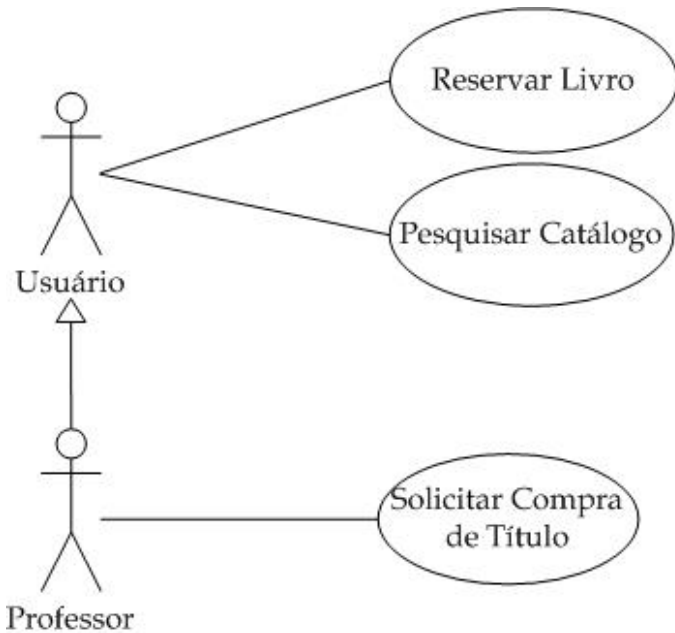
***Include(Fornecer Identificação)***

# Extensão (extend)





# Generalização



# Dica 4: Erros comuns em casos de uso

(ver documento no STOA)