

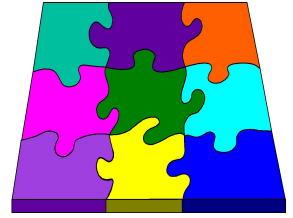
# **Princípios de Análise e Projeto de Sistemas com UML**

3ª edição (2015)

Eduardo Bezerra  
[ebezerra@cefet-rj.br](mailto:ebezerra@cefet-rj.br)



# Tópicos



- Introdução
- Diagrama de casos de uso
- Identificação dos elementos do MCU
- Construção do MCU
- Documentação suplementar ao MCU
- O MCU em um processo de desenvolvimento iterativo e incremental

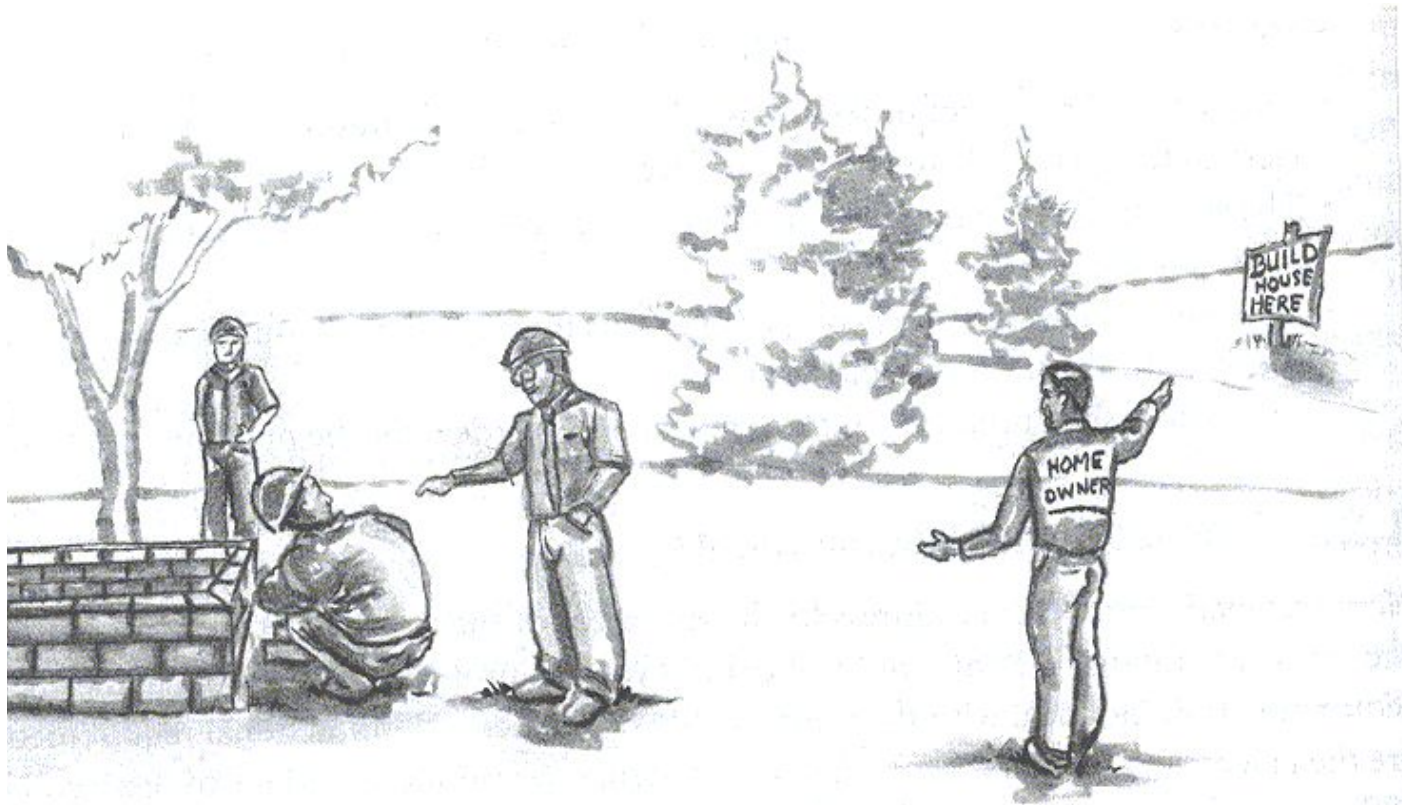
# Introdução

- O *modelo de casos de uso* é uma representação das *funcionalidades* externamente observáveis do sistema e dos *elementos externos* ao sistema que interagem com o mesmo.
- Esse modelo representa os *requisitos funcionais* do sistema.
- Também direciona diversas das atividades posteriores do ciclo de vida do sistema de software.
- Além disso, força os desenvolvedores a moldar o sistema de acordo com as **necessidades** do usuário.

# Utilidade dos Casos de Uso

- Equipe de clientes (**validação**)
  - aprovam o que o sistema deverá fazer
  - entendem o que o sistema deverá fazer
- Equipe de desenvolvedores
  - Ponto de partida para refinar requisitos de software.
  - Podem seguir um desenvolvimento dirigido a casos de uso.
  - Designer (projetista): encontrar classes
  - Testadores: usam como base para **casos de teste**

# Utilidade dos Casos de Uso



*Homeowner: "Hey, I wanted that foundation laid over there!"*

# Composição do MCU

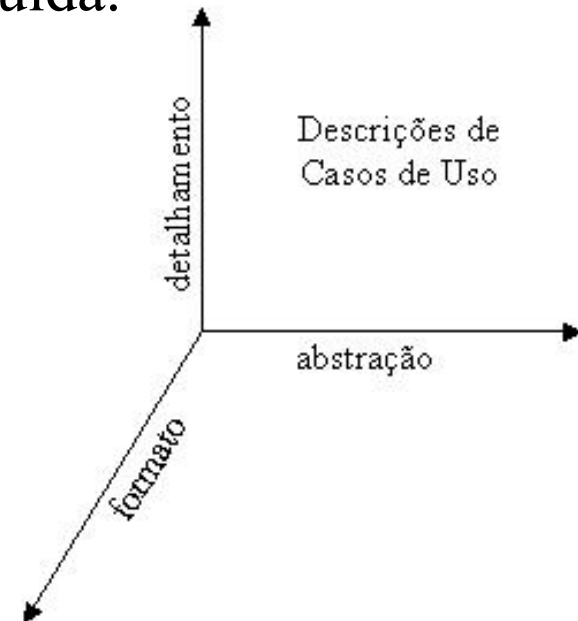
- O modelo de casos de uso de um sistema é composto de duas partes, uma **textual**, e outra **gráfica**.
- O diagrama da UML utilizado na modelagem de gráfica é o *diagrama de casos de uso*.
  - Este diagrama permite dar uma visão global e de alto nível do sistema.
  - É também chamado de diagrama de contexto.
- Componentes: casos de uso, atores, relacionamentos entre os elementos anteriores.

# Casos de uso

- Um caso de uso é a especificação de uma seqüência de interações entre um sistema e os agentes externos.
- Define parte da funcionalidade de um sistema, *sem revelar a estrutura e o comportamento internos deste sistema*.
- Um modelo de casos de uso típico é formado de vários casos de uso.
- Cada caso de uso é definido através da **descrição textual** das interações que ocorrem entre o(s) elemento(s) externo(s) e o sistema.
- Há várias “dimensões de estilo” para descrição de casos de uso: Grau de abstração; Formato; Grau de detalhamento.

# Dimensões para Descrições Textuais

- Um caso de uso é definido através da descrição textual das interações entre o(s) elemento(s) externo(s) e o sistema.
- Entretanto, a UML não define nada acerca de como essa descrição textual deve ser construída.
- Por conta disso, há várias dimensões independentes sob as quais a descrição textual de um caso de uso pode variar:
  - Grau de abstração (essencial ou real)
  - Formato (contínua, tabular, numerado)
  - Grau de detalhamento (sucinta ou expandida)





# Formato

- Exemplo de descrição contínua

Este caso de uso inicia quando o Cliente chega ao caixa eletrônico e insere seu cartão. O Sistema requisita a senha do Cliente. Após o Cliente fornecer sua senha e esta ser validada, o Sistema exibe as opções de operações possíveis. O Cliente opta por realizar um saque. Então o Sistema requisita o total a ser sacado. O Cliente fornece o valor da quantidade que deseja sacar. O Sistema fornece a quantia desejada e imprime o recibo para o Cliente. O Cliente retira a quantia e o recibo, e o caso de uso termina.

# Formato

- Exemplo de descrição numerada

- 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.

# Formato

- Exemplo de descrição tabular

Cliente	Sistema
<p>Insere seu cartão no caixa eletrônico.</p> <p>Digita senha.</p> <p>Solicita realização de saque.</p> <p>Fornece o valor da quantia que deseja sacar.</p> <p>Retira a quantia e o recibo.</p>	<p>Apresenta solicitação de senha.</p> <p>Valida senha e exibe menu de operações disponíveis.</p> <p>Requisita quantia a ser sacada.</p> <p>Fornece a quantia desejada e imprime o recibo para o Cliente</p>

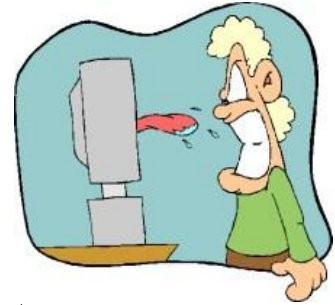
# Grau de Abstração

- Exemplo de descrição essencial (e numerada):

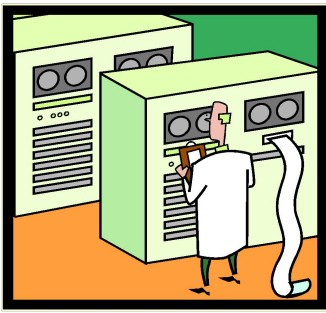
- 1) Cliente fornece sua identificação.
- 2) Sistema identifica o usuário.
- 3) Sistema fornece opções disponíveis para movimentação da conta.
- 4) Cliente solicita o saque de uma determinada quantia.
- 5) Sistema requisita o valor da quantia a ser sacada.
- 6) Cliente fornece o valor da quantia que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

•*Dica: regra dos 100 anos*

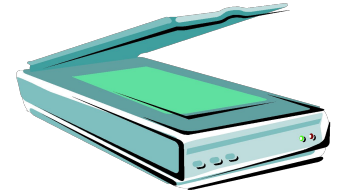
# Atores



- Elemento externo que interage com o sistema.
  - “externo”: atores não fazem parte do sistema.
  - “interage”: um ator troca informações com o sistema.
- Casos de uso representam uma seqüência de interações entre o sistema e o ator.
  - no sentido de troca de informações entre eles.
- Normalmente um agente externo inicia a seqüência de interações como o sistema.



# Atores



- Categorias de atores:
  - *cargos* (Empregado, Cliente, Gerente, Almoxarife, Vendedor, etc);
  - *organizações* (Empresa Fornecedora, Agência de Impostos, Administradora de Cartões, etc);
  - *outros sistemas* (Sistema de Cobrança, Sistema de Estoque de Produtos, etc).
  - *equipamentos* (Leitora de Código de Barras, Sensor, etc.)
- Essa categorização indica para nós que o conceito de ator depende do **escopo** do sistema.



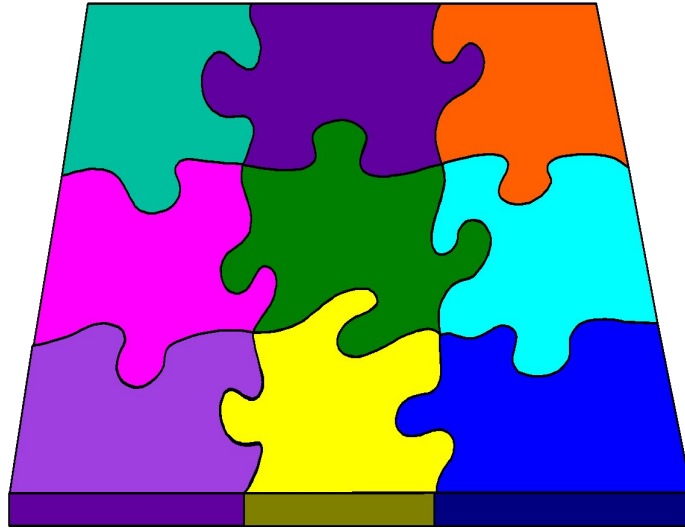
# Atores

- Um ator corresponde a um *papel* representado em relação ao sistema.
  - O mesmo indivíduo pode ser o **Cliente** que compra mercadorias e o **Vendedor** que processa vendas.
  - Uma pessoa pode representar o papel de **Funcionário** de uma instituição bancária que realiza a manutenção de um caixa eletrônico, mas também pode ser o **Cliente** do banco que realiza o saque de uma quantia.
- O nome dado a um ator deve lembrar o seu papel, em vez de lembrar quem o representa.
  - e.g.: João Fernandes *versus* Fornecedor

# Atores versus Casos de Uso

- Um **ator** representa um conjunto coerente de papéis que os usuários de casos desempenham quando interagem com o sistema
- Um **caso de uso** representa o que um ator quer que o sistema faça.
- Atores servem para definir o **ambiente do sistema**
- Atores representam um **papel** exercido por uma pessoa ou por um sistema externo que interage com o sistema.
- Se comunicam enviando mensagens e/ou recebendo mensagens do sistema, conforme o caso de uso é executado
- Quando definimos o que os atores fazem e o que os casos de uso fazem, delimitamos, de forma clara, o **escopo do sistema**.



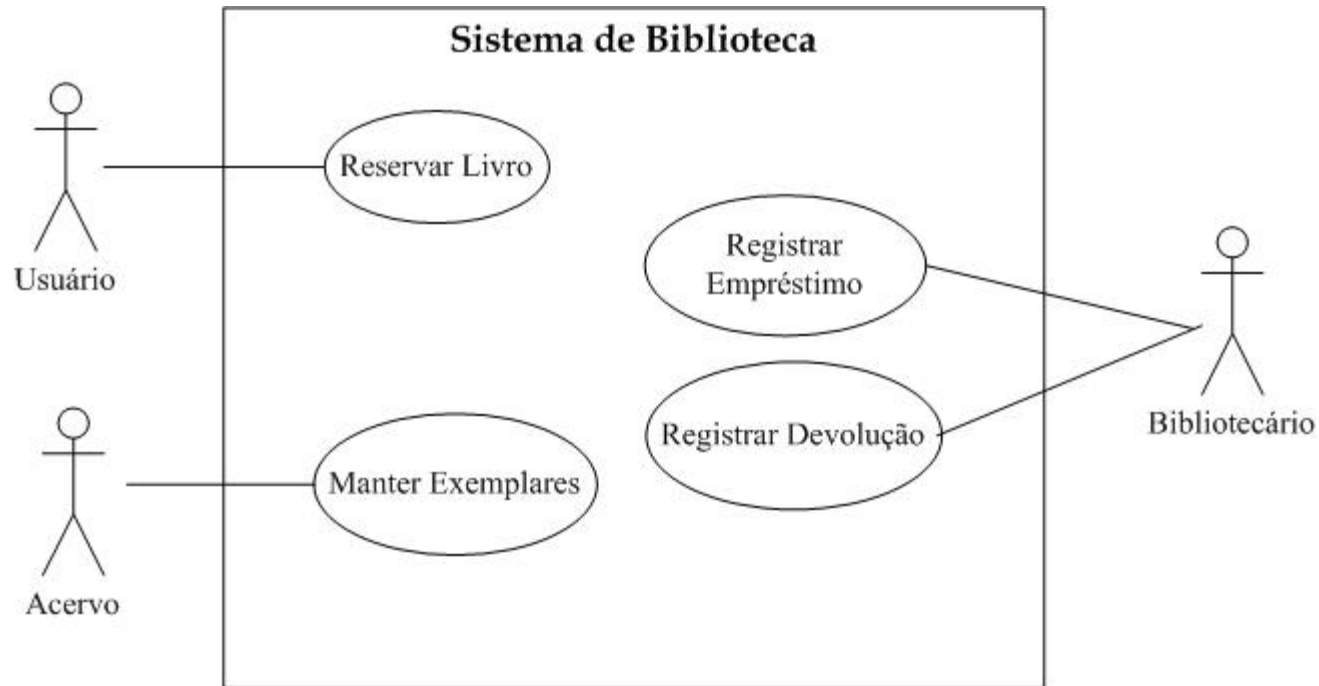


## 4.2 Diagrama de casos de uso

# Diagrama de casos de uso (DCU)

- Representa *graficamente* os atores, casos de uso e relacionamentos entre os elementos.
- Tem o objetivo de ilustrar em um nível alto de abstração quais elementos externos interagem com que funcionalidades do sistema.
- Uma espécie de “diagrama de contexto”.
  - Apresenta os elementos externos de um sistema e as maneiras segundo as quais eles as utilizam.

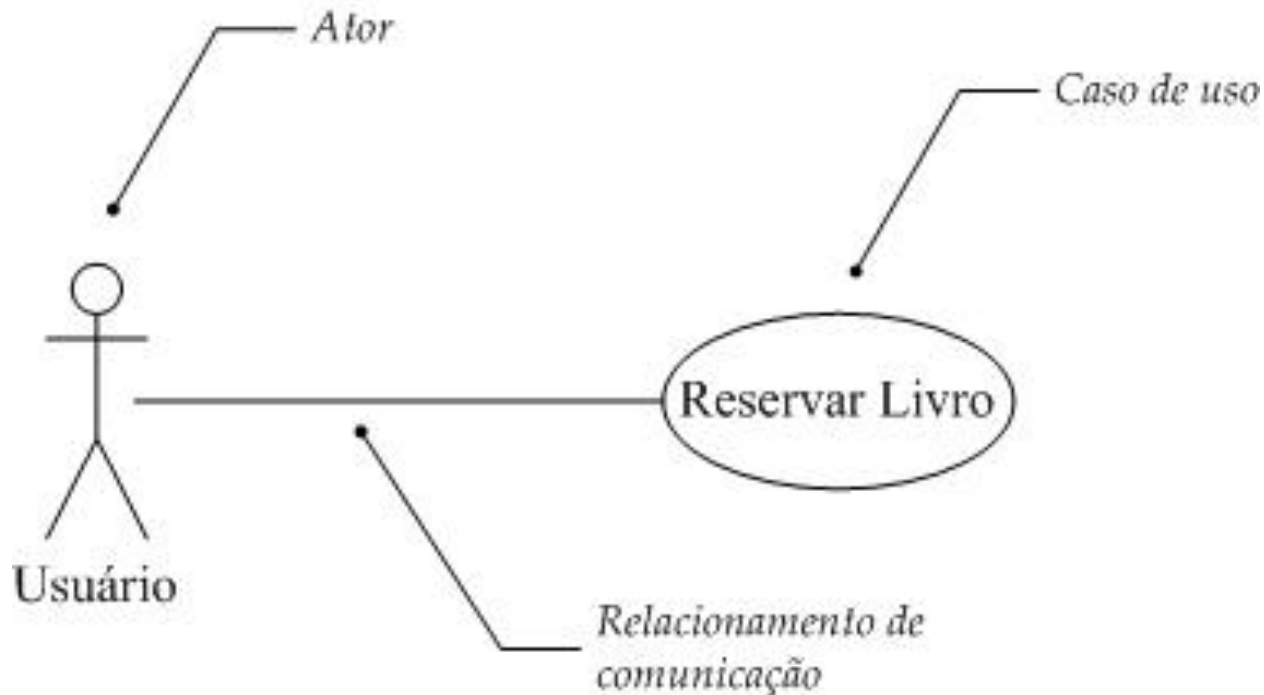
# Exemplo de DCU



# Elementos de um MCU

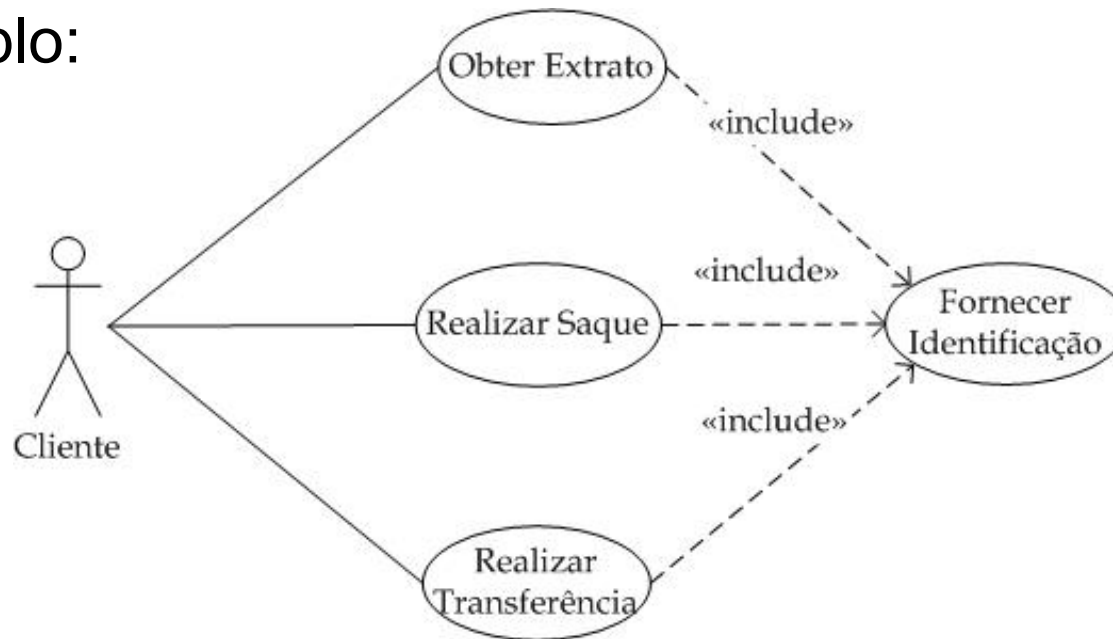
- Um MCU possui diversos elementos, e cada um deles pode ser representado graficamente. Os elementos mais comuns em um MCU são:
  - *Ator*
  - *Caso de uso*
- Além disso, a UML define diversos de relacionamentos entre esses elementos para serem usados no modelo de casos de uso:
  - *Comunicação*
  - *Inclusão*
  - *Extensão*
  - *Generalização*
- Para cada um desses elementos, a UML define uma notação gráfica e uma semântica específicas.

# Ator, caso de uso, comunicação



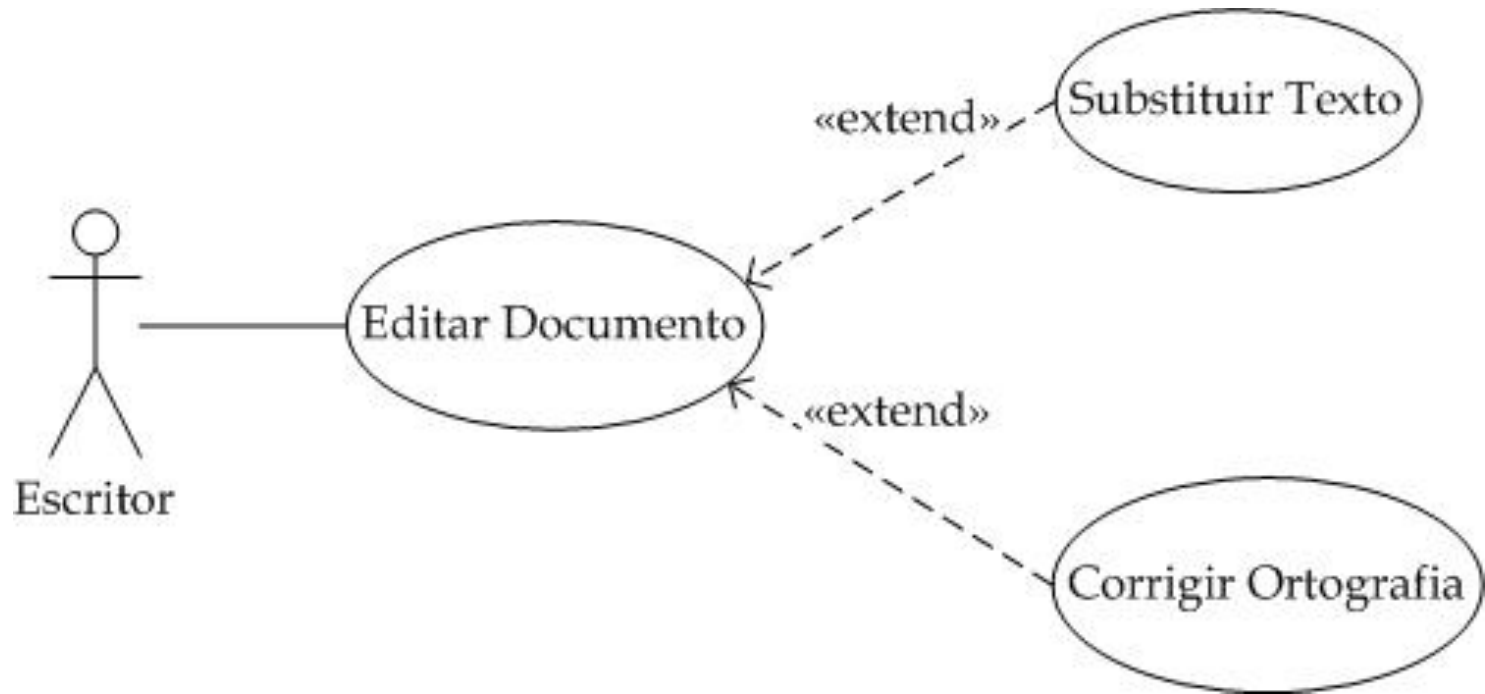
# Inclusão (include)

- Exemplo:

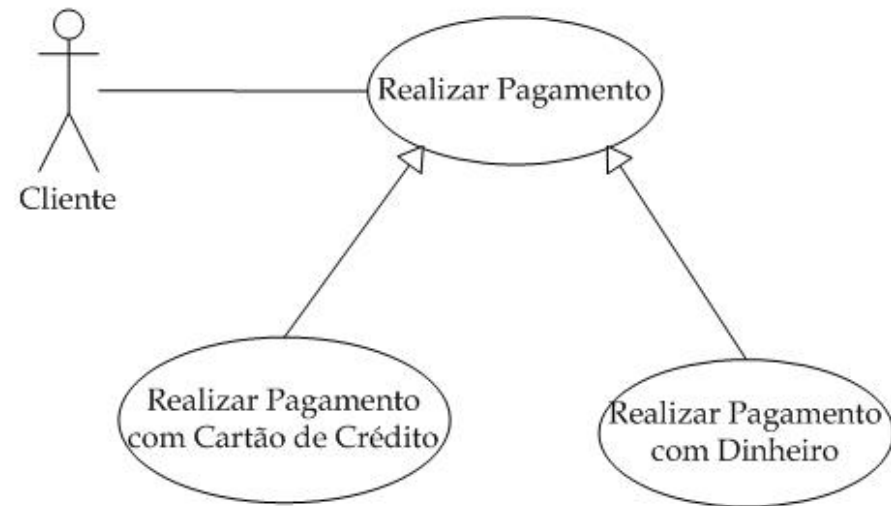
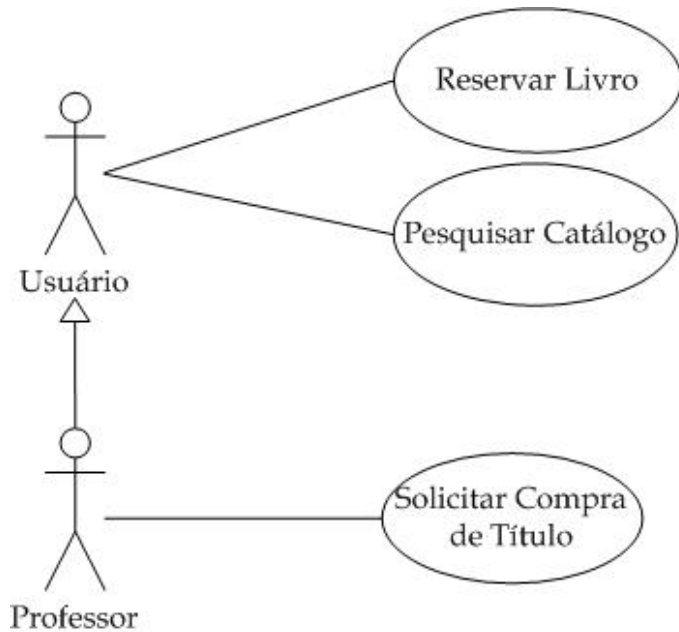


- Referência no texto do caso de uso inclusor:  
***Include(Fornecer Identificação)***

# Extensão (extend)



# Generalização

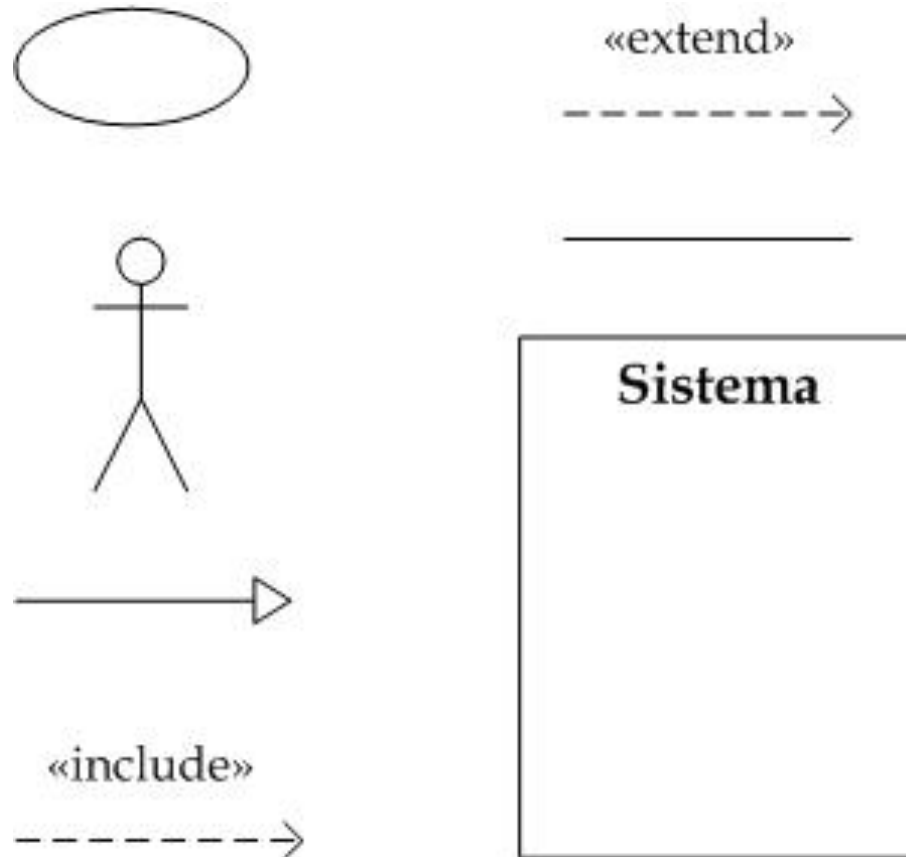


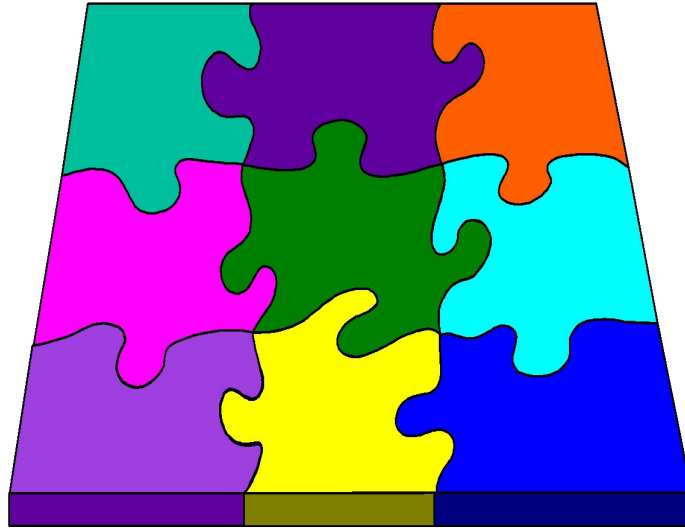


**Tabela 4-1:** Possibilidades de relacionamentos entre os elementos do modelo de casos de uso

	Comunicação	Extensão	Inclusão	Herança
Caso de uso e caso de uso		X	X	X
Ator e ator				X
Caso de uso e ator	X			

# Resumo da Notação





## 4.3 Identificação dos elementos do MCU

# Identificação dos elementos do MCU

- Atores e os casos de uso são identificados a partir de informações coletadas no levantamento de requisitos.
  - Durante esta fase, analistas devem identificar as atividades do negócio relevantes ao sistema a ser construído.
- Não há uma regra geral que indique quantos casos de uso e atores são necessários para descrever um sistema.
  - A quantidade de casos de uso e atores depende da complexidade do sistema.
- Note também que as identificações de atores e de casos de uso são atividades que se intercalam.

# Identificação de atores



- Fontes e os destinos das informações a serem processadas são atores em potencial.
  - uma vez que, por definição, um ator é todo elemento externo que *interage* com o sistema.
- O analista deve identificar:
  - as áreas da empresa que serão afetadas ou utilizarão o sistema.
  - fontes de informações a serem processadas e os destinos das informações geradas pelo sistema.

# Identificação de atores

- Há algumas perguntas úteis cujas respostas potencialmente identificam atores.
  - Que órgãos, empresas ou pessoas (cargos) irão utilizar o sistema?
  - Que outros sistemas irão se comunicar com o sistema?
  - Alguém deve ser informado de alguma ocorrência no sistema?
  - Quem está interessado em um certo requisito funcional do sistema?

# Identificação de Casos de Uso

- A partir da lista (inicial) de atores, deve-se passar à identificação dos casos de uso.
- Nessa identificação, pode-se distinguir entre dois tipos de casos de uso
  - Primário: representa os *objetivos* dos atores.
  - Secundário: aquele que não traz benefício direto para os atores, mas que é necessário para que sistema funcione adequadamente.

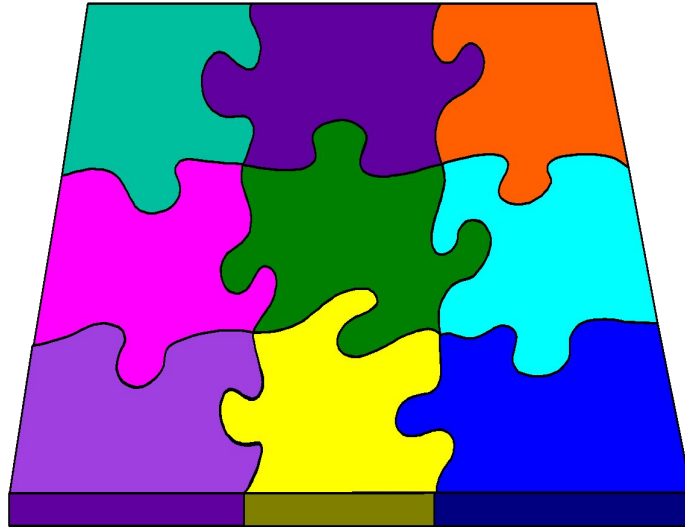
# Casos de Uso Primários

- Perguntas úteis:
  - Quais são as necessidades e objetivos de cada ator em relação ao sistema?
  - Que informações o sistema deve produzir?
  - O sistema deve realizar alguma ação que ocorre regularmente no tempo?
  - Para cada requisito funcional, existe um (ou mais) caso(s) de uso para atendê-lo?
- Outras técnicas de identificação:
  - *Caso de uso “oposto”*
  - *Caso de uso que precede/sucedee a outro caso de uso*
  - *Caso de uso temporal*
  - *Caso de uso relacionado a uma condição interna*



# Casos de Uso Secundários

- Estes se encaixam nas seguintes categorias:
  - Manutenção de cadastros;
  - Manutenção de usuários;
  - Gerenciamento de acesso;
  - Manutenção de informações provenientes de outros sistemas.
- Obs: casos de uso secundários, são menos importantes que os casos de uso primários.
  - O sistema de software não existe para cadastrar informações, nem tampouco para gerenciar os usuários.
  - O objetivo principal de um sistema é agregar **valor** ao ambiente no qual ele está implantado.



## 4.4 Construção do MCU

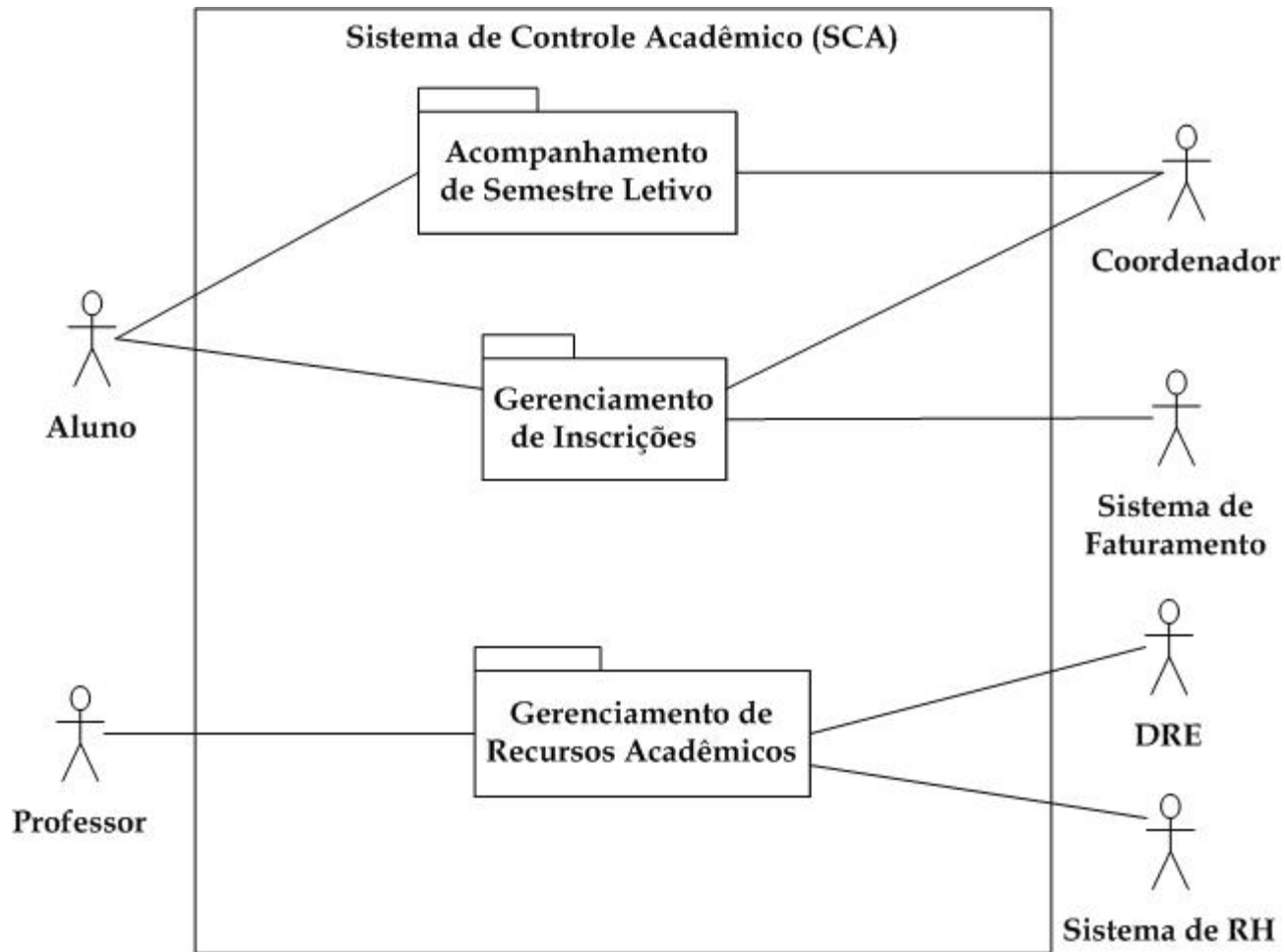
# Construção do DCU

- Os diagramas de casos de uso devem servir para dar suporte à parte textual do modelo, fornecendo uma visão de alto nível.
- Quanto mais fácil for a leitura do diagrama representando casos de uso, melhor.
- Se o sistema sendo modelado não for tão complexo, pode ser criado um único DCU.
- É útil e recomendada a utilização do retângulo de fronteira para delimitar e separar visualmente casos de uso e atores.

# Construção do DCU (cont.)

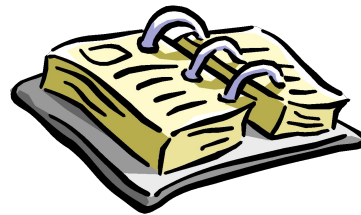
- Em sistemas complexos, representar todos os casos de uso do sistema em um único DCU talvez o torne um tanto ilegível.
- Alternativa: criar vários diagramas (de acordo com as necessidades de visualização) e agrupá-los em **pacotes**.
  - Todos os casos de uso para um ator;
  - Todos os casos de uso a serem implementados em um ciclo de desenvolvimento.
  - Todos os casos de uso de uma área (departamento, seção) específica da empresa.

# Construção do DCU (cont.)



# Documentação dos atores

- Uma breve descrição para cada ator deve ser adicionada ao MCU.
- O nome de um ator deve lembrar o papel desempenhado pelo mesmo.
- Exemplo  
**“Aluno: representa pessoas que fazem um curso dentro da universidade.”**



# Documentação dos casos de uso



- Infelizmente, a UML não define um padrão para descrição textual dos casos de uso de um sistema.
- Por conta disso, há diversos estilos de descrição possíveis (numerada, livre, tabular, etc).
- É necessário, no entanto que a equipe de desenvolvimento padronize o seu estilo de descrição.
- Algumas seções normalmente encontradas:
  - Sumário
  - Atores
  - Fluxo principal
  - Fluxos alternativos
  - Referências cruzadas (para requisitos não funcionais)

# Documentação dos casos de uso

- Nome
- Descrição
- Identificador
- Importância
- Sumário
- Ator Primário
- Atores Secundários
- Pré-condições
- Fluxo Principal
- Fluxos Alternativos
- Fluxos de Exceção
- Pós-condições
- Regras do Negócio
- Histórico
- Notas de Implementação

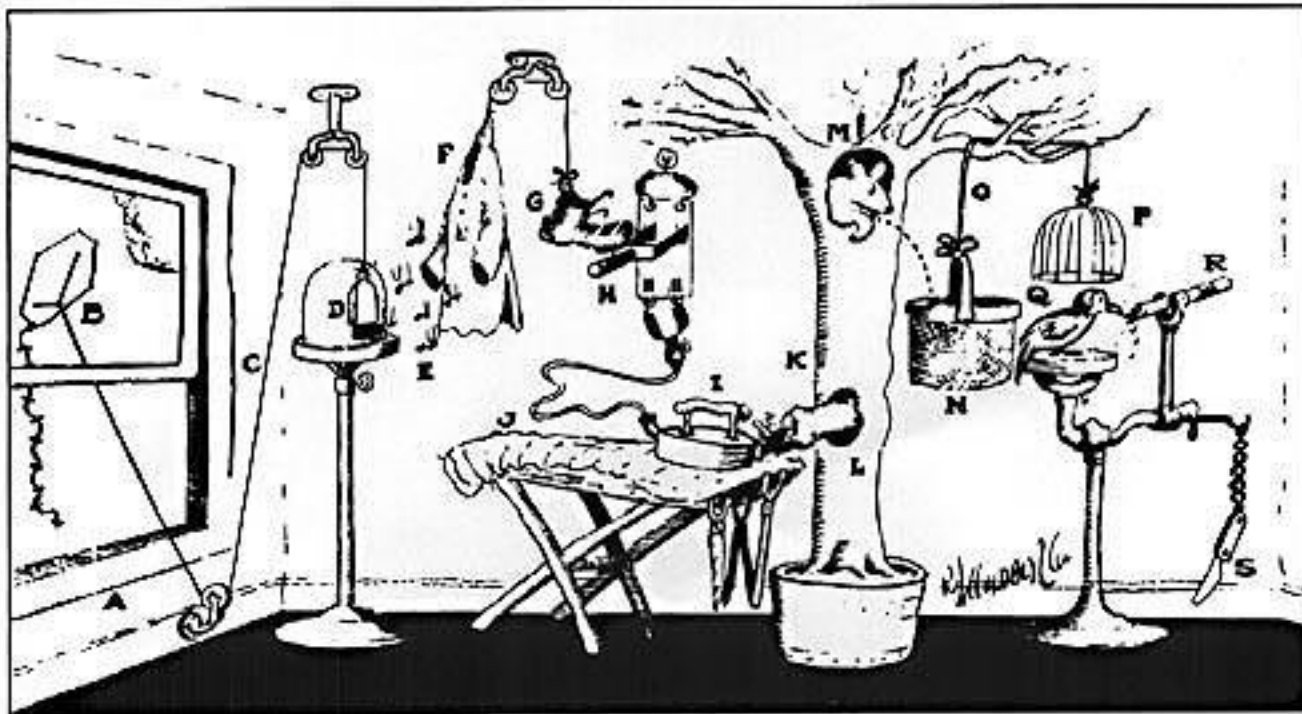


# Documentação dos casos de uso

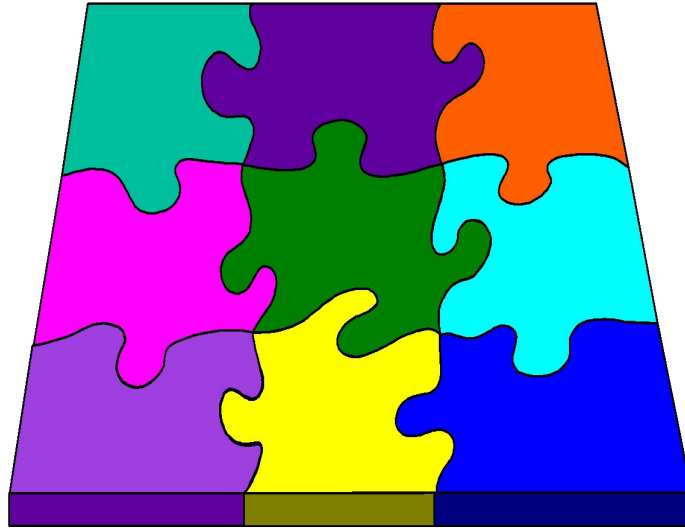
- Algumas boas práticas na documentação de casos de uso.
  - Comece o nome do caso de uso com um verbo no infinitivo (para indicar um processo ou ação).
  - Tente descrever os passos de caso de sempre na forma sujeito + predicado. Ou seja, deixe explícito quem é o agente da ação.
  - Não descreva **como** o sistema realiza internamente um passo de um caso de uso.
    - "You apply use cases to capture the intended behavior of the system [...], without having to specify how that behavior is implemented. (Booch)
  - Tente dar nomes a casos de uso seguindo perspectiva do ator primário. Foque no **objetivo** desse ator. Exemplos: Registrar Pedido, Abrir Ordem de Produção, Manter Referência, Alugar Filme, etc.
  - Tente manter a descrição de cada caso de uso no nível mais simples possível...

# Documentação dos casos de uso

- ...repetindo: tente manter a descrição de cada caso de uso no nível mais simples possível!



Pencil Sharpener RUBE GOLDBERG (tm) RQI 038



## 4.5 Documentação suplementar ao MCU

# Documentação Associada

- O modelo de casos de uso força o desenvolvedor a pensar em como os agentes externos interagem com o sistema.
- No entanto, este modelo corresponde somente aos requisitos funcionais.
- Outros tipos de requisitos (desempenho, interface, segurança, regras do negócio, etc.) também devem ser identificados e modelados.
- Esses outros requisitos fazem parte da documentação associada ao MCU.
- Dois itens importantes dessa documentação associada são o *modelo de regras do negócio* e os *requisitos de desempenho*.

# Regras do Negócio

- São políticas, condições ou restrições que devem ser consideradas na execução dos processos de uma organização.
  - Descrevem a maneira pela qual a organização funciona.
- Estas regras são identificadas e documentadas no chamado *modelo de regras do negócio* (MRN).
  - A descrição do modelo de regras do negócio pode ser feita utilizando-se texto informal, ou através de alguma forma de estruturação.
- Regras do negócio normalmente influenciam o comportamento de determinados casos de uso.
  - Quando isso ocorre, os identificadores das regras do negócio devem ser adicionados à descrição dos casos de uso em questão.
  - Uso da seção “regras do negócio” da descrição do caso de uso.

# Exemplos de Regras do Negócio

- O valor total de um pedido é igual à soma dos totais dos itens do pedido acrescido de 10% de taxa de entrega.
- Um professor só pode estar lecionando disciplinas para as quais esteja habilitado.
- Um cliente de uma das agências do banco não pode retirar mais do que R\$ 1.000 por dia de sua conta. Após as 18:00h, esse limite cai para R\$ 100,00.
- Os pedidos para um cliente não especial devem ser pagos antecipadamente.

# Regras do Negócio

- Possível formato para documentação de uma regra de negócio no MRN.

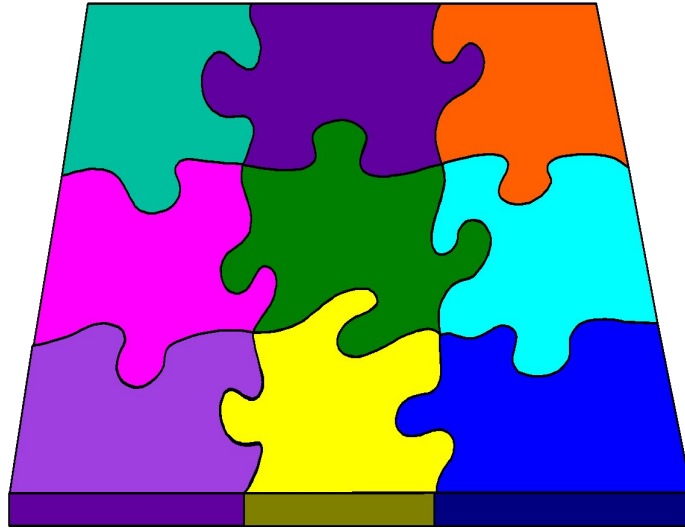
Nome	<b>Quantidade de inscrições possíveis (RN01)</b>
Descrição	Um aluno não pode ser inscrever em mais de seis disciplinas por semestre letivo.
Fonte	Coordenador da escola de informática
Histórico	Data de identificação: 12/07/2002

# Requisitos de desempenho

- Conexão de casos de uso a requisitos de desempenho.

Identificador do caso de uso	Frequência da utilização	Tempo máximo esperado	...
CSU01	5/mês	Interativo	...
CSU02	15/dia	1 segundo	...
CSU03	60/dia	Interativo	...
CSU04	180/dia	3 segundos	...
CSU05	600/mês	10 segundos	...
CSU07	500/dia durante 10 dias seguidos.	10 segundos	...





## 4.6 O MCU em um processo de desenvolvimento iterativo e incremental

# Casos de uso e outras atividades

- Validação
  - Clientes e usuários devem entender o modelo (**validação**) e usá-lo para comunicar suas necessidades de forma consistente e não redundante.
- Planejamento e gerenciamento do projeto
  - Uma ferramenta fundamental para o gerente de um projeto no planejamento e controle de um processo de desenvolvimento incremental e iterativo
- Testes do sistema
  - Os casos de uso e seus cenários oferecem *casos de teste*.

# Casos de uso e outras atividades (cont)

- Documentação do sistema para os usuários
  - manuais e guias do usuário podem ser construídos com base nos casos de uso.
- Realização de uma iteração
  - Os casos de uso podem se **alocados** entre os membros de equipe de desenvolvimento
- Essa estratégia de utilizar o MCU como ponto de partida para outras atividades é denominada **Desenvolvimento Dirigido por Casos de Uso**
  - **Use Case Driven Development**

# MCU no processo de desenvolvimento

- Casos de uso formam uma base natural através da qual podem-se realizar as iterações do desenvolvimento.
- Um grupo de casos é alocado a cada iteração.
- Em cada iteração, o grupo de casos de uso é detalhado e desenvolvido.
- O processo continua até que todos os casos de uso tenham sido desenvolvidos e o sistema esteja completamente construído.
- A descrição expandida de um caso de uso pode ser deixada para a iteração na qual este deve ser implementado.
  - evita perda de tempo inicial no detalhamento.
  - estratégia mais adaptável aos requisitos voláteis.

# MCU no processo de desenvolvimento

- Cantor propõe uma classificação em função do risco de desenvolvimento e das prioridades estabelecidas pelo usuário.
  - 1) Risco alto e prioridade alta
  - 2) Risco alto e prioridade baixa
  - 3) Risco baixo e prioridade alta
  - 4) Risco baixo e prioridade baixa
- Considerando-se essa categorização, devemos considerar os casos de uso mais importantes e mais arriscados primeiramente.
  - Atacar o risco maior mais cedo...