**Exercício 4-1:** descreva a posição do diagramas de casos de uso no processo de desenvolvimento incremental e iterativo. Quando eles são utilizados? Para que são utilizados?

Do livro: Princípios de Análise e Projeto de Sistemas com UML. Eduardo Bezerra. Editora Campus.

Considerando que o processo de desenvolvimento incremental é utilizado, a identificação da maioria dos atores e casos de uso é feita pelos analistas na fase de concepção. A descrição dos casos de uso considerados mais críticos começa já nessa fase, que termina com 10% a 20% do modelo de casos de uso completo. Na fase de elaboração, a construção do modelo continua de tal forma que, ao seu término, 80% do modelo de casos de uso esteja construído.

Na fase de construção, 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. Então, o desenvolvimento do sistema segue a alocação realizada: em cada iteração, um grupo de casos de uso é detalhado (utilizando um nível de abstração real) e desenvolvido. O processo continua até que todos os casos de uso tenham sido desenvolvidos e o sistema esteja completamente construído. Esse tipo de desenvolvimento é também chamado de desenvolvimento dirigido a casos de uso.

Conforme mencionado há pouco nesta seção, um fator importante para o sucesso do desenvolvimento do sistema é considerar os casos de uso mais importantes primeiramente. Murray Cantor (Cantor, 1998) propõe uma classificação dos casos de uso identificados para um sistema em função de dois parâmetros: risco de desenvolvimento e prioridades estabelecidas pelo usuário. Dessa forma, cada caso de uso se encaixa em uma das categorias a seguir:

1. Risco alto e prioridade alta: casos de uso nesta categoria são os mais críticos. Devem ser considerados o quanto antes.

2. Risco alto e prioridade baixa: embora os casos de uso nesta categoria tenham risco alto, é necessário, antes de começar a considerá-los, negociar com o cliente em relação a sua verdadeira necessidade.

3. Risco baixo e prioridade alta: embora os casos de uso tenham prioridade alta, é necessário ter em mente que os casos de uso de mais alto risco devem ser considerados primeiro.

4. Risco baixo e prioridade baixa: em situações em que o desenvolvimento do sistema está atrasado, estes casos de uso são os primeiros a serem "cortados".

Através da atribuição de importância segundo a categorização de Cantor, um caso de uso não tão importante não será contemplado nas iterações iniciais. Se o requisito correspondente a esse caso de uso for modificado ou não mais precisar ser considerado, os analistas não terão desperdiçado tempo com ele.

Note também que a descrição expandida de um determinado caso de uso é deixada para a iteração na qual este deve ser implementado. Isso evita que se perca tempo inicialmente no seu detalhamento. Além disso, essa estratégia é mais adaptável aos requisitos voláteis. Se todos os casos de uso forem detalhados inicialmente, e se um ou mais requisitos são modificados durante o desenvolvimento, toda a modelagem correspondente a esses casos de uso sofrerá modificações. Por outro lado, se a descrição detalhada é deixada para a iteração à qual o caso de uso foi alocado, uma eventual mudança dos requisitos associados a esse caso de uso não afetará tão profundamente o desenvolvimento.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A construção do modelo de casos de uso deve se adequar ao processo de desenvolvimento sendo utilizado. Os casos de uso mais arriscados devem ser considerados primeiramente.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Casos de uso nas atividades de análise e projeto

Alguns desenvolvedores escrevem as descrições iniciais de casos de uso mencionando detalhes de interface gráfica com o usuário (considerando atores humanos). A justificativa é que a narrativa dos casos de uso segundo essa abordagem fornece uma idéia mais concreta de como se apresentará uma determinada funcionalidade do sistema. Contudo, as desvantagens dessa abordagem se sobrepõem às vantagens.

Considere a situação de uma parte da interface estar sendo continuamente modificada, por alguma razão. Nessa situação, a desvantagem de utilização de casos de uso reais (vide Seção 4.1.1.3) se torna nítida, pois o fato de a interface ser modificada possivelmente resultará na modificação da narrativa do caso de uso. Além disso, lembre-se do objetivo principal do modelo de casos de uso, a saber, modelar os requisitos do sistema. Portanto, casos de uso devem ser independentes do desenho da interface pelo fato de que os requisitos do sistema não devem estar associados a detalhes de interface.

Uma melhor abordagem é utilizar inicialmente casos de uso essenciais (vide Seção 4.1.1.3) para não acoplar os detalhes da interface da aplicação na especificação narrativa das interações de um caso de uso. Nessa especificação, a atenção do modelador deve recair sobre a essência das interações entre atores e o sistema, em vez de como cada interação é realizada fisicamente. Especificações de casos de uso feitas dessa forma ficam mais imunes a futuras mudanças na interface com o usuário, além de permitir que o analista de sistemas se concentre no que é realmente importante em uma narrativa de caso de uso: as interações entre ator(es) e sistema. Por exemplo, considere o termo "envia uma requisição" em contraposição com o termo "duplo clique sobre o botão de envio de requisições".

Em resumo, casos de uso que mencionam detalhes de interface gráfica são indesejáveis durante a análise. O mais adequado é utilizar casos de uso essenciais e, posteriormente, na etapa de projeto, transformá-los em casos de uso reais adicionando mais detalhes.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Na fase de análise, descrições de casos de uso devem capturar os requisitos funcionais do sistema e ignorar aspectos de projeto, como a interface gráfica com o usuário.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Pode-se, agora, descrever um procedimento a ser utilizado na construção do modelo de casos de uso quando da utilização de um processo de desenvolvimento incremental:

1. Identifique os atores e casos de uso na fase de concepção. Alguns atores e casos de uso só serão identificados posteriormente, mas a grande maioria deve ser descoberta nesta fase.

2. Na fase de elaboração:

a. Desenhe o(s) diagrama(s) de casos de uso;

b. Escreva os casos de uso em um formato de alto nível e essencial.

c. Ordene a lista de casos de uso de acordo com prioridade e risco. Cada partição corresponde a um grupo de casos de uso que será implementado em um dos ciclos de desenvolvimento do sistema.

3. Associe cada grupo de casos de uso a urna iteração da fase de construção. Os grupos mais prioritários e arriscados devem ser alocados às iterações iniciais.

4. Na i-ésima iteração da fase de construção:

a. Detalhe os casos de uso do grupo associado a esta iteração (se necessário, utilize o nível de abstração real).

1. Implemente estes casos de uso.

# Casos de uso e outras atividades do desenvolvimento

O modelo de casos de uso direciona a realização de várias outras atividades do desenvolvimento.

# Planejamento e gerenciamento do projeto

O modelo de casos de uso é uma ferramenta fundamental para o gerente de um projeto no planejamento e controle de um processo de desenvolvimento incremental e iterativo. Ao final de cada iteração, o gerente pode avaliar a produtividade na realização das tarefas. Essa avaliação serve como massa de dados para que esse profissional realize a alocação das tarefas e recursos para as próximas iterações.

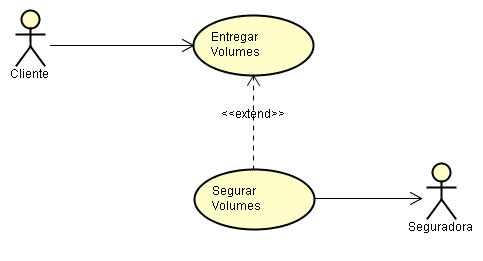
# Testes do sistema

Em um processo de desenvolvimento incremental e iterativo, não há uma fase de testes propriamente dita. Ao contrário, os testes do software são realizados continuamente durante todo o desenvolvimento. Os profissionais responsáveis pelos testes utilizam o modelo de casos de uso para planejar as atividades de teste. Os casos de uso e seus cenários oferecem casos de teste. Quando o sistema está sendo testado, os cenários sobre o sistema podem ser verificados para identificar a existência de erros.

# Documentação do usuário

Os manuais e guias do usuário também podem ser construídos com base no modelo de casos de uso. Na verdade, se o modelo de casos de uso foi bem construído, deve haver uma correspondência clara entre cada caso de uso do sistema e uma seção do manual do usuário. Isso porque esse modelo está baseado na noção de que o sistema é construído para se adequar à perspectiva de seus usuários.

**Exercício 4-2:** construa um modelo de casos de uso para a seguinte situação fictícia: *"Estamos criando um serviço de entregas. Nossos clientes podem nos requisitar a entrega de volumes. Alguns volumes são considerados de maior valor por nossos clientes, e, portanto, eles querem ter tais volumes segurados durante o transporte. Contratamos uma companhia de seguro para segurar volumes de valor".*



**Exercício 4-3:** Considerea seguinte narrativa do caso de uso Realizar Saque. Identifique os erros existentes nesta narrativa. Construa uma nova versão deste caso de uso que não contenha os erros encontrados.

*A operação de um caixa eletrônico tem início a partir de uma sessão em que o cliente seleciona a opção de realizar saque. O cliente então escolhe uma quantia a ser retirada, a partir de um conjunto de opções de quantia disponíveis.*

*O sistema verifica se a conta correspondente tem saldo suficiente para satisfazer a requisição. Senão, uma mensagem adequada é reportada, o que acarreta na execução da extensão. Se há dinheiro suficiente, os números da conta e da agência do cliente são enviados ao banco, que aprova ou desaprova a transação. Se a transação é aprovada, a máquina libera a quantia correspondente e emite um recibo. Se a transação é desaprovada, a extensão Informar Falha é executada.*

*O banco é notificado, independentemente de uma transação aprovada ter sido completada ou não pela máquina. Se a transação é completada, o banco realiza o débito na conta do cliente (Bjork, 1998).*

# Caso de Uso - Realizar Saque

**Sumário**: Este caso de uso possibilita a um cliente realize um saque de um caixa eletrônico

**Ator Primário**: Cliente

**Ator Secundário**: Banco

**Pré-Condições**: Cliente autenticado

# Fluxo Principal

1. O caso de uso tem início quando o ator **Cliente** seleciona a opção realizar **saque**
2. O sistema pergunta ao Cliente a quantia a ser retirada.

**{Especifica Valor}**

1. O Cliente digita a quantia desejada.

**{Verifica Disponibilidade de Valor no Caixa}**

1. Executa o sub-fluxo “**Avalia Quantia Disponível”**.

**{Verifica Saldo Suficiente}**

1. O sistema contata o ator banco para determinar se existe **saldo** suficiente na **conta** do Cliente.

**{Aprova Transação}**

1. O sistema inicia uma **transação** com o ator banco e solicita a **retirada** da quantia desejada.
2. O sistema libera a quantia desejada
3. O sistema emite um **recibo** para o Cliente
4. O sistema fecha a **transação** com o ator banco.
5. O sistema armazena um **log** da transação.
6. O caso de uso se encerra.

# S1: *Avalia Quantia Disponível*

1. O sistema determina se tem fundos suficientes à mão para fornecer a quantia solicitada
2. O sistema verifica se a importância requisitada é maior do que a quantia disponível.
3. O sistema verifica se a importância desejada pode ser fornecida com as notas existentes no caixa eletrônico. (R$ 50,00 não podem ser fornecidos se só houver três notas de R$ 20,00).

**Fluxos Alternativos**

A1 *O cliente não digita a quantia desejada*

Em {**Especifica Valor**} se o ator cliente não especifica a quantia desejada

1. ...

A2 *O caixa automático não pode fornecer a quantia solicitada*

Em **{Verifica Disponibilidade de Valor no Caixa}** se o caixa não tem disponibilidade de dinheiro para atender a solicitação do ator cliente.

1. O sistema reporta uma mensagem adequada
2. O caso de uso se encerra.

A3 *O link com o banco caiu*

Em qualquer ponto do fluxo principal.

1. ...

A4 *O Cliente não tem saldo suficiente*

Em **{Verifica Saldo Suficiente}** se o ator Cliente não tem recursos suficientes em sua conta para cobrir a retirada

1. O sistema reporta uma mensagem adequada
2. O caso de uso se encerra.

A5 *O banco não aprova a transação*

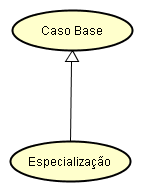
Em **{Aprova Transação}** se o ator Banco não aprova a transação devido à violação de alguma regra de negócio (por exemplo: limite diário excedido)

1. O sistema reporta uma mensagem adequada
2. Retorna ao fluxo principal.

**Exercício 4-4:** qual é a notação da UML para um caso *de uso?* Qual é a notação da UML para um ator? Qual a notação utilizada na UML para o relacionamento de generalização?







**Exercício 4-5:** defina o que significa um ator. O que significa um ator estar associado a um caso de uso por um relacionamento de comunicação?

Ator:

* Um ator define um papel que pode ser desempenhado por um usuário na sua interação com o sistema. Um usuário aqui pode ser um indivíduo ou um outro sistema.
* Um mesmo usuário pode assumir vários papéis ao longo de sua interação com o sistema
* Uma lista de atores é uma lista de papéis e não uma lista de usuários.
* Atores estão fora do sistema, e normalmente fora do controle do sistema

Conectando atores e casos de uso:

* Os atores e os casos do uso com os quais eles interagem são ligados pela associação de comunicação.
* A seta é opcional, mas, quando usada, ela indica qual elemento começa a interação.
* Para entender plenamente o papel definido para um ator, você deve saber em que casos de uso o ator está envolvido.
* Para entender plenamente o alcance de um caso de uso, você deve saber os atores com os quais ele se comunica.
* Atores se comunicam com o sistema por muitas razões, incluindo:
  + Iniciar um caso de uso. Os casos de uso sempre são iniciados por atores.
  + Pedir alguns dados armazenados no sistema, os quais então o caso do uso apresenta ao ator.
  + Mudar os dados armazenados no sistema por meio de um dialogo com o sistema.
  + Informar que ocorreu algo que o sistema deve estar ciente,.
* Um ator inicia um caso de uso. Entretanto, depois que o caso de uso começou, ele pode se comunicar com vários outros atores.
* Considera-se às vezes, erradamente, que a associação de comunicação representa o fluxo de dados. Não é isso. A associação de comunicação representa um diálogo entre o ator e o sistema, um tipo de canal de comunicação sobre o qual podem fluir dados em ambas as direções durante o diálogo.
* Casos de uso se comunicam com atores por muitos motivos:
  + Se algo especial aconteceu no sistema, um ator pode ter de ser informado.
  + Um caso de uso pode necessitar da ajuda de um ator para tomar uma decisão.
  + Um caso de uso pode delegar responsabilidade a um ator.

**Exercício 4-6:** qual o objetivo dos diagramas de *casos de uso?*

O diagrama de casos de uso tem o objetivo de ilustrar em um nível alto de abstração quais elementos externos interagem com que funcionalidades do sistema. Nesse sentido, a finalidade de um DCU é apresentar um tipo de “diagrama de contexto” que apresenta os elementos externos de um sistema e as maneiras segundo as quais eles as utilizam.

**Exercício 4-7:** defina o conceito de requisito. Que tipos de requisitos existem? Explique o que é realizado na fase de levantamento de requisitos de um sistema de informações.

Um requisito descreve uma condição ou capacidade a que um sistema deve se adaptar; eles podem ser derivados diretamente da necessidade de um *stakeholder* ou usuário ou extraídos de um contrato, padrão, especificação, ou outro documento formalmente imposto. Às vezes é útil expressar tipos de requisitos diferentes:

Requisitos Funcionais:

* Ações que o produto deve realizar de modo a fornecer funcionalidades úteis para seus usuários. Estes requisitos definem as razões fundamentais para a existência do produto.
* Exemplo: Software para uma central telefônica servindo a um prédio de apartamentos: O sistema deve emitir uma conta telefônica por apartamento considerando todas as ligações realizadas pelo apartamento no mês, contendo as seguintes informações:...

Requisitos não funcionais:

* São propriedades ou qualidades que o produto deve possuir
* Estes requisitos normalmente são relacionados à funcionalidade do produto, ou seja, uma vez que saibamos o que o sistema deve fazer, podemos determinar como ele irá se comportar e que características de qualidade ele deve apresentar (por exemplo, performance e nível de segurança desejado)
* Exemplo: Automóvel

acelerar, mudar marcha (requisitos funcionais)

conforto, cor do painel de instrumentos (requisitos não funcionais)

**Exercício 4-8:** que tipo de relacionamento é possível entre um ator e um caso de uso? Que tipo de relacionamento pode haver entre casos de uso? Que tipo de relacionamento pode haverentre atores?

A tabela a seguir exibe as alternativas possíveis entre relacionamentos entre atores e casos de uso em um diagrama de casos de uso. As células da tabela com um X indicam possibilidade. As células não preenchidas indicam impossibilidade.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Entre atores | Entre casos de uso | Entre ator e caso de uso |
| Comunicação |  |  | X |
| Inclusão |  | X |  |
| Extensão |  | X |  |
| Generalização | X | X |  |

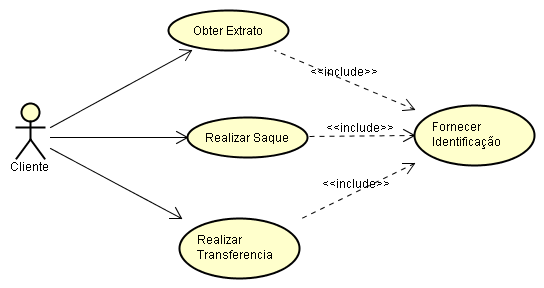
**Exercício 4-9**: descreva a(s) diferença(s) entre os relacionamentos de inclusão, de extensão e de herança?

*“If most systems can be fully described using only use cases with relationships only to their actors, what would force us to add relationships between use cases, especially when we have already noted that most teams get into trouble when they try to use them? The simple answer is that two forces draw us toward using relationships. One force is commonality in behavior between two or more use cases, and the other is reducing complexity by isolating portions of use cases that may apply only in specific circumstances. When common behavior occurs, gathering it into a use case of its own can lead to improved readability and consistency of description. Similarly, if some behavior applies only in very specific contexts, separating that behavior into a use case of its own can make the rest of the use case easier to understand. The trick is knowing when to separate behavior into its own use case and when to leave well enough alone.*

*No matter what—and we cannot emphasize this enough—do not introduce relationships between use cases until you have at least a draft of the flow of events of the use cases. Outlining may seem sufficient, but it often lacks the detail necessary to see commonality (in the case of inclusion) or variation (in the case of extension). The only reason for introducing relationships is to deal with commonality and variations in the flows of events of the use cases; if you introduce them earlier, you are doing so without any real knowledge. Once you have at least drafted the use-case descriptions, commonality and variations will become obvious and you may safely proceed, provided that introducing the relationships will increase the understandability of the use-case descriptions.”*

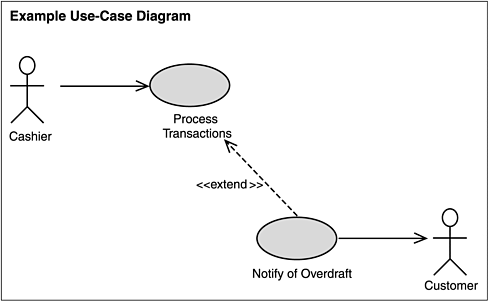
*Use Case Modeling – Ian Spence & Kurt Bittner,*

### **O relacionamento Include:**



1. Um relacionamento *include* permite extrair seções comuns da descrição de dois ou mais casos e colocá-las em um caso de uso separado a partir do qual elas podem ser referenciadas. Cada caso de uso original passa a ter então um relacionamento *includes* com o novo caso de uso;

### **O relacionamento Extend:**



O relacionamento *extend* é usado em casos onde comportamento *opcional* ou *excepcional* é inserido em um caso de uso existente. O propósito original da extensão era fornecer um mecanismo para especificar opções que pudessem ser adicionadas a um produto existente tais como: adicionar o envio de e-mails por voz a um serviço telefônico convencional existente. É útil pensar no relacionamento de extensão como um relacionamento de *acréscimo,* uma vez que ele sempre acrescenta funcionalidade a um caso de uso existente. A característica marcante do caso de uso que estende um caso de uso original é que ele não demanda nenhuma alteração no caso de uso original. Isto significa que o caso de uso estendido deve ser capaz de “se virar sozinho”. Ele deve ser completo, sem qualquer necessidade de extensões a fim de gerar valor.

As seguintes situações podem dar margem à utilização do *extend:*

* Descrições de características que são opcionais ao comportamento básico do sistema, por exemplo, características que podem ser adquiridas ou não.
* Descrições complexas de erros ou tratamentos de exceções que, de outra forma, iriam obscurecer o comportamento primário do sistema. Exemplos disso são fluxos alternativos de tamanho significativo, especialmente aqueles cujo tamanho é maior do que o do fluxo principal.
* Customização do modelo de requisitos para atender a necessidades específicas do usuário. Exemplos disso são fluxos alternativos que especificam como usuários específicos tratam diferentes condições que ocorrem dentro de um mesmo caso de uso.
* Gerência de escopo e versão. Um exemplo disso são características que não serão introduzidas até as últimas versões.

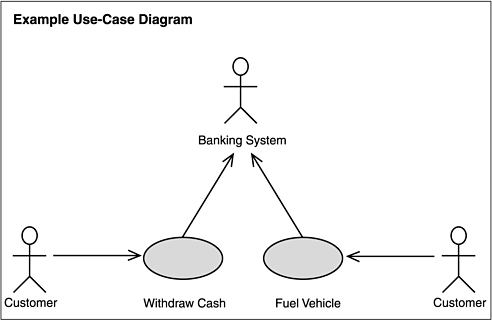
Em contraste ao relacionamento *include*, um relacionamento *extend* conhece necessariamente o caso de uso que ele estende.

Conceitualmente, o mecanismo de extensão é idêntico aquele dos fluxos alternativos. Um caso de uso de extensão, assim como um fluxo alternativo, insere a si próprio no fluxo do caso de uso que ele estende. Somente o caso de uso de extensão conhece o ponto no caso de uso base onde o comportamento será inserido. Em conseqüência, freqüentemente um caso de uso de extensão começa sua vida como um fluxo alternativo.

Nem todo fluxo alternativo deve virar um caso de uso de extensão. As regras para os fluxos alternativos são mais frouxas do que aquelas para os casos de uso de extensão. Devido ao fato de que os fluxos alternativos são parte do caso de uso, eles podem explorar seu conhecimento do estado do caso de uso, suas pré-condições, e outros fluxos de eventos para terminar o caso de uso ou para continuar o fluxo do caso de uso em pontos de extensão diferentes daquele onde eles assumiram o controle. *Tudo o que os casos de uso de extensão conhecem a respeito do caso de uso original é o ponto de extensão onde eles introduziram a si próprios no fluxo de eventos do caso de uso estendido.*

## Generalização entre Casos de Uso

A generalização entre casos de uso nos permite criar descrições genéricas de comportamento que podemos especializar para satisfazer necessidades particulares.



***Generalização entre atores***

A generalização entre atores é usada para mostrar semelhanças entre atores. O principal valor é mostrar que alguns grupos de atores compartilham responsabilidades ou características comuns.

Algumas vezes, o uso de generalização entre atores pode simplificar o modelo de casos de uso reduzindo o número de linhas de comunicação entre atores e casos de uso. Na maioria das vezes, no entanto, ela é de nenhuma utilidade. O uso de generalização entre atores é tipicamente um sintoma de que os modeladores confundiram papéis com cargos da organização.

Não revi as respostas daqui em diante... Jonas

**Exercício 4-10**: considere um sistema de controle de uma biblioteca. Forneça a descrição narrativa para os seguintes casos de uso: Reservar Livro (situação em que um usuário faz a reserva de um livro), Obter Empréstimo de Livro (situação em que um usuário pega um exemplar de livro emprestado), Cancelar Reserva (situação em que um usuário cancela uma reserva) e Devolver Cópia (situação em que um usuário devolve uma cópia anteriormente adquirida).

Observação: Pessoalmente, tenho dúvidas se Reservar Livro e Cancelar Reserva são dois casos de uso distintos. Acho que são fluxos diferentes do mesmo caso de uso: “Manter Reserva”

# Caso de Uso - Reservar Livro

**Sumário**: Este caso de uso possibilita a um usuário da biblioteca fazer a reserva de um livro. Esta reserva é efetuada diretamente pelo Usuário usando a Internet.

**Ator** **Primário**: Usuário

**Pré-Condições**: O Usuário é cadastrado na Biblioteca. O Usuário já foi autenticado pelo Sistema.

# Fluxo Principal

1. O caso de uso tem início quando o ator **usuário** decide reservar um **livro** e escolhe a opção correspondente.

**{Localizar livro}**

1. Usuário fornece a **identificação** do livro desejado
2. O sistema localiza o livro desejado

**{Verificar disponibilidade de exemplar}**

1. O sistema verifica a disponibilidade de **exemplar** do **livro** desejado

**{Verifica reservas em aberto}**

1. O sistema verifica o número de reservas em aberto para o usuário (**RN01**)
2. O sistema efetua a reserva e informa ao usuário o prazo máximo para a retirada do livro (**RN02**)
3. O caso de uso se encerra

**Fluxos Alternativos**

A1 *Não existe nenhum exemplar disponível*

Em **{Verificar disponibilidade de exemplar}** se não existe exemplar disponível para reserva

1. O sistema reporta uma mensagem adequada para o usuário e informa a data prevista para que haja um exemplar disponível
2. O sistema pergunta ao usuário se ele deseja efetuar a reserva de exemplar emprestado
3. Se o usuário responder afirmativamente, agenda a reserva e reporta uma mensagem informando que o usuário será contatado quando o exemplar estiver disponível, caso contrário o caso de uso se encerra.

A2 *O usuário excedeu o número máximo de reservas em aberto*

Em **{Verifica reservas em aberto}** se o usuário excedeu o número máximo permitido de reservas (RN01)

1. O sistema reporta uma mensagem adequada para o Usuário
2. O Caso de Uso se encerra

A3 *A biblioteca não possui o livro desejado*

Em **{Localizar livro}** se a biblioteca não tem o livro desejado

1. O sistema reporta uma mensagem adequada para o Usuário
2. O Caso de Uso se encerra

# Obter Empréstimo de Livro

Sumário: Este caso de uso possibilita a um usuário pegar emprestado um exemplar de livro.

Ator Primário: Bibliotecária

Pré-Condições: Bibliotecária identificada. A Bibliotecária tem em mãos o exemplar a ser emprestado.

Fluxo Principal

* 1. A Bibliotecária entra com a identificação do exemplar e do usuário.
  2. O sistema verifica a existência de reservas para o exemplar desejado
  3. O sistema verifica a existência de **empréstimos em aberto** para o usuário.
  4. O sistema verifica o número de exemplares em poder do usuário (**RN03**)
  5. O sistema registra o empréstimo e imprime um recibo contendo os **dados do empréstimo**
  6. O caso de uso se encerra

Fluxo Alternativo (2): Já existe uma reserva do exemplar desejado para outro usuário

* + 1. O sistema reporta uma mensagem adequada para a bibliotecária
    2. O caso de uso se encerra

Fluxo Alternativo (2): O exemplar estava reservado para o próprio usuário

1. O sistema dá baixa na reserva
2. Retorna ao passo (3)

Fluxo Alternativo (3): O usuário têm **empréstimos em aberto**

1. O sistema reporta uma mensagem adequada
2. O caso de uso se encerra

Fluxo Alternativo (4): O usuário excedeu o número máximo de empréstimos

1. O sistema reporta uma mensagem adequada
2. O caso de uso se encerra

# Cancelar Reserva

Sumário: Este caso de uso possibilita a um usuário cancelar uma reserva. O próprio usuário, usando a Internet, registra o cancelamento.

Ator Primário: Usuário

Pré-Condições: O Usuário é cadastrado na Biblioteca. O Usuário já foi autenticado pelo Sistema.

Fluxo Principal:

1. O usuário solicita sua lista de reservas
2. O sistema apresenta a lista de reservas
3. O usuário seleciona a reserva a ser cancelada
4. O sistema pede a confirmação do usuário para cancelar a reserva
5. O usuário confirma o cancelamento
6. O sistema cancela a reserva
7. O sistema mostra para o usuário a lista de reservas atualizada e oferece ao usuário a opção de efetuar novo cancelamento ou encerrar o caso de uso.

Fluxo Alternativo (2): O usuário não tem reservas em seu nome

1. O sistema reporta uma mensagem apropriada
2. O caso de uso se encerra

Fluxo Alternativo (5): o usuário não confirma o cancelamento da reserva

1. Volta para o passo 2

# Devolver Cópia

Sumário: O usuário devolve um exemplar em seu poder

Ator Primário: Bibliotecária

Pré-Condições: Bibliotecária identificada. A Bibliotecária tem em mãos o exemplar a ser devolvido.

Fluxo Principal:

1. A bibliotecária entra com o código do exemplar
2. O sistema apresenta o registro do empréstimo
3. A bibliotecária confirma a devolução
4. O sistema registra a devolução
5. O sistema verifica se houve atraso na devolução
6. O sistema verifica a existência de reservas agendadas para o exemplar devolvido
7. O caso de uso se encerra

Fluxo Alternativo (5): devolução em atraso

1. O sistema calcula o número de dias em atraso e a multa a ser paga (**RN04**)
2. O sistema reporta uma mensagem apropriada
3. A bibliotecária registra o pagamento da multa
4. Volta ao passo 6

Fluxo Alternativo (6): existem reservas agendadas para o exemplar devolvido

1. Executa o Caso de Uso “Efetivar Reserva Agendada”
2. Volta para o passo 7

#### Glossário

**Empréstimo em Aberto**: empréstimo vencido e não devolvido

**Dados do Empréstimo**: título do livro, código do exemplar, data do empréstimo, data da devolução

#### Regras de Negócio

**RN01**: Um Usuário não pode ter mais de duas reservas em aberto em seu nome.

**RN02**: O prazo máximo para a retirada de um livro reservado é de dois dias.

**RN03**: Um Usuário pode ter no máximo três exemplares em seu poder num determinado instante de tempo.

**RN04**: A multa a ser aplicada é de R$ 1,00 por dia útil de atraso.

**Exercício 4-11**: durante a execução de um caso de uso, podem ocorrer exceções. Considere o caso de uso Realizar Pedido, no qual pode ser que o cliente solicite um produto que está fora de estoque. Como você modelaria tal situação? Desenhe um diagrama de casos de uso.

# Realizar Pedido

Fluxo Principal:

...

O usuário entra com a identificação do produto

O sistema verifica no estoque a disponibilidade do produto pedido

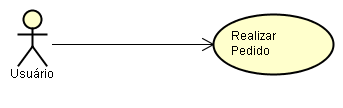
O sistema acrescenta o produto ao carrinho de compras do usuário

...

Fluxo de Exceção (...): O produto solicitado não existe em estoque

O sistema reporta uma mensagem apropriada

...



**Exercício 4-12**: construa o modelo de casos de uso para a seguinte situação. Tente identificar também regras de negócio que se apliquem à situação, de acordo com o texto fornecido.

*Uma rede de televisão está requisitando um sistema para gerenciar informações sobre uma de suas produções televisivas (por exemplo, uma minissérie ou uma novela).*

*Uma produção televisiva tem uma verba e é composta de cenas. Cenas são escolhidas em uma determinada seqüência. Cada cena tem uma duração em minutos e é gravada em uma ou mais fitas. Cada fita possui um número de série e uma capacidade (medida em minutos que podem ser gravados na mesma). Deseja-se saber em que fita(s) se encontra uma determinada cena. Cada cena pode ter sido gravada muitas vezes (futuramente, na edição da obra, o produtor selecionará uma dessas tomadas de cena para compor a versão final da produção televisiva). Deve-se manter o registro de todas as cenas filmadas, de quais atores e dublês participaram de cada cena. Deseja-se saber também, que dublê substituiu que ator em cada cena.*

*Para uma produção televisiva como um todo, deseja-se manter a informação de quais outros funcionários, os chamados funcionários de apoio, participaram das filmagens. Esses funcionários podem ser de diversos tipos (câmeras, iluminadores, contra-regras etc.). Além disso, pode haver funcionários de apoio que exerçam mais de uma função na mesma produção televisiva.*

*Atores e dublês negociam seus salários individualmente, em cada produção televisiva em que participam. Os demais funcionários têm um salário fixo por função. É necessário também armazenar essas informações para ter uma idéia do consumo de recursos em relação à verba.*

*Após o término de uma obra, o sistema deve produzir um relatório com o valor a ser pago para cada funcionário. O sistema também deve produzir um relatório de informações sobre as cenas de uma obra televisiva, e sobre que atores, dublês e demais funcionários participaram dessa obra televisiva.*

# Atores

Produtor

# Modelo de Casos de Uso

Casos de Uso Principais:

CSU01: Obter informações sobre cenas

Sumário: Através deste caso de uso o ator obtém informações sobre uma cena (gravações efetuadas, atores e dublês que participaram de cada cena, fitas onde foram gravadas as cenas, etc.)

CSU02: Obter informações sobre funcionários que participam da produção

Sumário: Relação de funcionários que participaram das filmagens, funções exercidas, salários pagos, etc.

CSU03: Gerar relação de cenas

Sumário: Relação de cenas segundo o número de seqüência, contendo informações tais como: sinopse, duração, personagens envolvidos, número de gravações efetuadas, datas das gravações, gravação escolhida, etc.

Casos de Uso Secundários:

Casos de Uso CRUD (Create, Read, Update, Delete). Exemplos:

Cadastrar Produção (Registrar detalhes da produção: verba, título, etc.)

Cadastrar Cena (Registrar detalhes de uma cena: duração, locação, personagens, etc.)

Registrar Gravação de Cena (data, locação, atores envolvidos, dublês utilizados, fitas gravadas, etc.)

Cadastrar Fita (numero de série, capacidade, etc.)

Cadastrar Função (funções necessárias à produção: descrição, salário, etc.)

Cadastrar Funcionário (registrar participação do funcionário na produção: funções exercidas)

Cadastrar Ator (registrar participação de ator: dados pessoais, salário negociado, etc.)

Cadastrar Dublê (registrar participação de dublê: dados pessoais, salário negociado, etc.)

# Regras de Negócio

1. Um funcionário pode exercer mais de uma função na produção
2. Uma fita não pode ser usada para a gravação de mais de uma cena nem para gravações diferentes de uma mesma cena
3. ...

**Exercício 4-13**: o seguinte documento de requisitos foi adaptado do livro (Wirfs-Brock et aI, 1991). Leia o texto com atenção. A seguir, elabore um modelo de casos de uso inicial para o sistema.

*O GNU Editor é um editor gráfico interativo. Com ele, usuários podem criar e editar desenhos compostos de linhas, retângulos, elipses e texto.*

*Há dois modos de operação do editor. Apenas um modo de operação está ativo em um dado momento.*

*Os dois modos de operação são: modo de seleção e modo de criação. Quando o modo de seleção está ativado, os elementos gráficos podem ser selecionados com o cursor do mouse. Um ou mais elementos gráficos podem ser selecionados e manipulados; se vários elementos gráficos forem selecionados, eles podem ser manipulados como se fossem um único elemento gráfico. Elementos que tenham sido selecionados desse modo são definidos como a "seleção atual". A seleção atual é indicada visualmente através da exibição dos pontos de controle para o elemento. Um clique seguido de um arrasto de mouse sobre um ponto de controle modifica o elemento ao qual o ponto de controle está associado.*

*Quando o modo de criação está ativado, a seleção atual está vazia. O usuário pode selecionar um objeto gráfico a partir de um conjunto de objetos gráficos predefinidos.*

*A criação de um elemento de texto: a posição do primeiro caractere do texto é determinada pela posição na qual o usuário clica o botão do mouse. O modo de criação é desativado quando o usuário clica o mouse fora do elemento de texto. Os pontos de controle para um elemento de texto são posicionados nos quatro cantos da região em que o texto é inserido. O arrasto desses pontos de controle muda a região.*

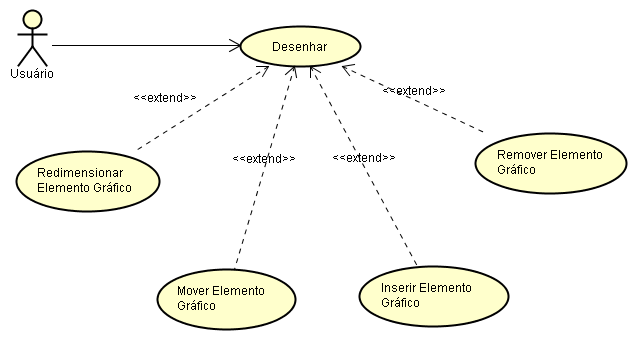
*Os outros elementos que podem ser criados pelo usuário são linhas, retângulos e elipses. O elemento apropriado começa quando o botão do mouse é pressionado e se completa quando o botão do mouse é liberado. Esses dois eventos criam o "ponto de partida" e o "ponto de parada"*

*A "criação de linha" define uma linha do ponto de partida até o ponto de parada. Esses são os pontos de controle. O arrasto de um ponto de controle modifica o ponto extremo correspondente.*

*A "criação de retângulo" define um retângulo tal que dois dos cantos do retângulo diametralmente opostos do retângulo correspondem ao ponto de partida e ao ponto de parada. Os cantos do retângulo formam os pontos de controle. O arrasto de um ponto de controle modifica o canto correspondente.*

*A "criação de elipse” define uma elipse que está contida dentro de um retângulo definido pelos dois pontos definidos acima. O raio maior da elipse é metade do comprimento do retângulo, e o seu raio menor é metade da altura do retângulo. Os pontos de controle são os cantos do retângulo que contém a elipse. O arrasto de um ponto de controle modifica o canto correspondente.*

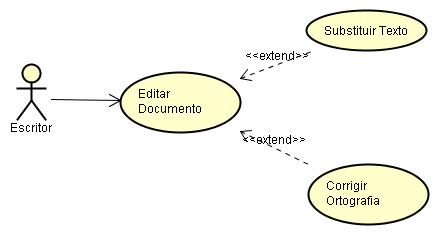
*Será assumido que o programa deve fornecer uma tela gráfica do diagrama sendo criado, e que um mouse e um teclado serão utilizados como dispositivos de entrada.*

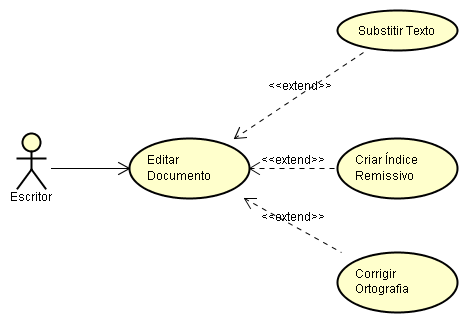


**Exercício 4-14**: considere a seguinte declaração obtida de um gerente de uma empresa que comercializa livros por correio durante o levantamento de requisitos para construção de um sistema de software: *"Após a ordem de compra do cliente ter sido registrada, o vendedor envia uma requisição ao depósito com detalhes da ordem de compra."* Quais atores em potencial podem ser identificados a partir desse texto?

Considerando-se somente o trecho fornecido no exercício, podem ser identificados 3 atores em potencial, a saber: Cliente, Vendedor e Depósito. O primeiro é o ator primário e os dois últimos são atores secundários. O nome do caso de uso correspondente poderia ser Comprar Produtos.

**Exercício 4-15**: considere o exemplo de relacionamento de extensão entre casos de uso apresentado na Seção 4.1.3.3 , que descreve relacionamentos de extensão entre os casos de uso Editar Documento e os extensores Corrigir Ortografia e Substituir Texto. Desenhe um diagrama de casos de uso para essa situação. Como você faria para estender seu diagrama de casos de uso com um novo requisito, a saber, permitir que o editor de textos possibilite a criação de um índice remissivo sobre um documento sendo editado?



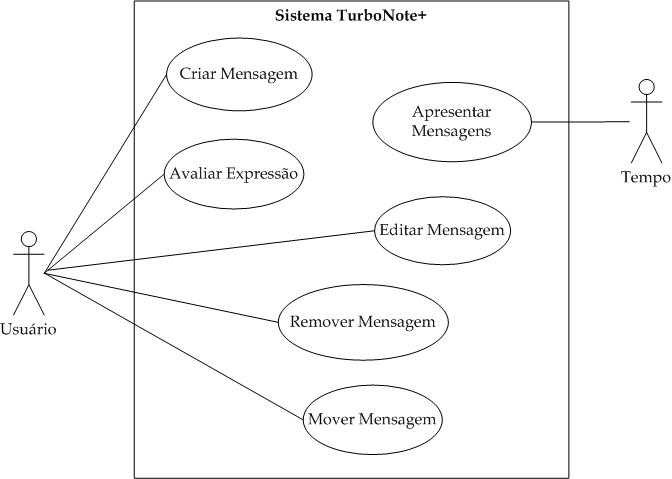


**Exercício 4-16**: em uma empresa, vários projetos são realizados. Os cinqüenta empregados da empresa trabalham em pelos menos um projeto. Há um sistema implantado na empresa que permite aos participantes de um determinado projeto marcarem suas horas de trabalho. Esse sistema também permite que outra pessoa, ao fim do mês, gere os relatórios com os totais de horas trabalhadas de cada participante. Quantos atores você definiria para esse sistema? E quantos papéis?

Na situação descrita neste exercício, pode-se definir um ator denominado Empregado . Este seria o ator primário no caso de uso Registrar Horas Trabalhadas. Podemos também criar um ator denominado Gerência. que seria o ator primário no caso de uso Obter Horas Trabalhadas. O diagrama de casos de uso a seguir ilustra a solução aqui descrita.



**Exercício 4-17**: O TurboNote+ é um programa *shareware* que permite aos seus usuários criar mensagens de lembrete que permanecem na área de trabalho de seus computadores. (Esse programa funciona como uma versão eletrônica daqueles bloquinhos de papel cujas folhas podem ser afixadas na parede.) Ao criar uma nova folhinha no Turbo-Note+, o usuário pode preenchê-la com texto. As folhinhas podem ser movidas pela área de trabalho, conforme a vontade do usuário. As folhinhas permanecem na área de trabalho. Toda vez que o usuário inicia o seu computador, as folhinhas estão lá, na área de trabalho. Quando não são mais necessárias, as folhinhas podem ser removidas. Se o usuário escrever uma expressão aritmética em uma folhinha, o resultado da expressão é exibido. Desenhe o diagrama de casos de uso para o TurboNote+.



Não Confunda Casos de Uso com "Funções" A infeliz semelhança visual entre diagramas de caso de uso e diagramas de fluxos de dados às vezes leva as pessoas a definir casos de uso que são em verdade apenas funções ou itens de menu. Qualquer que seja a razão, este é provavelmente o erro mais freqüente cometido pelos iniciantes em modelagem usando casos de uso. Veja Figura 4-5.

A figura 4-5. Uso incorreto de casos de uso como opções de menu ou funcões

O que há de errado com Figura 4-5? Pense novamente na nossa definição de uma descrição de uso-caso (“um relato sobre algum meio de usar o sistema para fazer algo útil”). Os casos de uso acima são todos, independentemente, úteis?

A resposta, naturalmente, é não. A figura retrata coisas que o sistema deve fazer, mas elas são todas relacionadas a uma única coisa que o cliente quer fazer no sistema: fazer pedidos. Todas as coisas restantes são fluxos alternativos deste caso de uso—são coisas que podem são feitas no curso de fazer um pedido. Onde há só uma coisa útil a ser feita, há só um caso de uso. A “solução” mostrada na Figura 4-5 é um exemplo de decomposição funcional, ou (como diz um colega meu) um exemplo dos "vagões em formação circular” - um ator no centro de um círculo de casos de uso. Este problema é muito comum. Por que as pessoas são levadas a este tipo de solução? Temos uma necessidade intrínseca para ordenar as coisas e, onde não existe nenhuma ordem, tentamos impô-la. No caso da decomposição funcional, temos uma tendência natural a tentar quebrar o problema em pedaços cada vez menores, numa crença ingênua de que assim procedendo podemos simplificar o problema. Esta percepção está errada; quando decompomos casos de uso, nós realmente complicamos o problema.

Eis por que.

O propósito de um caso de uso é descrever como alguém ou alguma coisa usará o sistema para fazer algo que é útil para ele. Ele descreve o que o sistema faz num nível conceitual de modo que podemos entender bastante sobre o sistema para decidir se o sistema se comporta corretamente ou não. Ele nos ajuda a formar um modelo conceitual do sistema. Agora pergunte a si mesmo: você vai querer usar este sistema somente para investigar o estado de um pedido se eu nunca fiz um pedido? Não é muito provável. Ou, eu vou querer mudar um pedido se eu nunca fiz um pedido? Não, provavelmente não. Todas estas coisas são úteis para mim somente se eu fiz um pedido; todos eles são necessários para prover a capacidade do sistema de me permitir fazer um pedido.

Decompor o sistema em casos de uso menores realmente obscurece o propósito real do sistema; no extremo, nós acabamos com uma quantidade enorme de pedaços de comportamento, desconexos e isolados. Nós não podemos dizer o que o sistema faz. É como olhar um carro que foi desmontado—talvez você possa dizer que é um carro, e você sabe que as partes devem ser úteis de alguma forma, mas você realmente não pode dizer como. Quando trabalhar com casos de uso, lembre-se que casos de uso são um meio de pensar no sistema total e organizá-lo em pedaços administráveis de funcionalidade— pedaços que fazem algo útil. Para obter o jogo correto de casos de uso, pergunte a si mesmo: "o que os atores realmente estão tentando fazer com este sistema?"

Caso você esteja se perguntando como a versão melhorada da Figura 4-5 se pareceria, observe a Figura 4-6. Estes dois casos de uso incluiriam todas as "funções" que o diagrama anterior separou em casos de uso. Você pode perguntar por que isto é melhor. A resposta é simples—ela foca no valor que o cliente quer do sistema, e não em como subdividimos e estruturamos a funcionalidade dentro do sistema.

**Exercício 4-18**: suponha que um sistema de vendas deve gerar de forma automática um conjunto de estatísticas para a diretoria da empresa no último dia útil de cada mês. Desenhe o diagrama de casos de uso para essa situação. Há mais de uma maneira de representá-la?

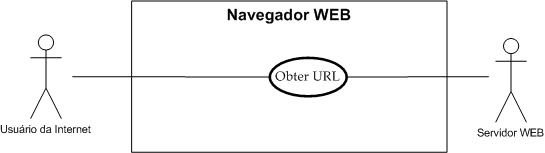


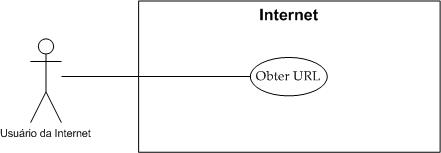
**Exercício 4-19**: Na utilização da Internet, normalmente um usuário utiliza um programa navegador (browser) que, por sua vez, se comunica com um ou mais servidores Web para fornecer as páginas nas quais o usuário está interessado. O que está errado no diagrama a seguir? Desenhe novos diagramas para representar corretamente a situação, considerando duas alternativas de escopo. Na primeira, o programa navegador é o sistema. Na segunda, a Internet é o sistema.



O erro do diagrama está no fato de que atores não se comunicam em um diagrama de casos de uso.

Na primeira alternativa de escopo (o sistema é o browser), o sistema sendo modelado tem que se comunicar (interagir) com dois atores no caso de uso Obter URL: Usuário da Internet e Servidor WEB. Na segunda alternativa de escopo (o sistema é a Internet), o próprio servidor WEB faz parte do sistema sendo modelado e por conta disse não deve ser representado como um ator. Os dois diagramas de casos de uso a seguir ilustram as duas alternativas diferentes.





**Exercício 4-20**: assinale V ou F para as seguintes assertivas:

(   ) pessoas com o mesmo cargo em uma empresa podem representar papéis de diversos atores.

(   ) um ator pode representar pessoas de diferentes cargos.

A primeira e a segunda assertiva são verdadeiras. Na verdade essas assertivas são formas diferentes de declarar a mesma informação: um ator representa um papel em relação ao sistema.

Considere o exemplo do exercício 4-16. Pode haver uma pessoa que seja um funcionário comum em um certo projeto, além de ser o gerente em outro projeto. Neste caso, a mesma pessoa assumirá papéis diferentes em instantes distintos em relação ao sistema.

**Exercício 4-21**: altere os seguintes "nomes de casos de uso" de acordo com as nomenclaturas apresentadas neste capítulo:

1. Cliente realiza transferência de fundos em um caixa eletrônico.
2. Clientes compram livros na livraria.
3. É produzido um relatório de vendas para o gerente.
4. Hóspede se registra em um hotel.

A seguir, são apresentados os nomes de casos de uso de acordo com a nomenclatura adotada no livro. Possíveis nomes para atores primários em cada situação são também fornecidos. Deve-se enfatizar, no entanto, que isso é somente uma convenção de nomenclatura. Outras convenções podem ser usadas.

|  |  |  |
| --- | --- | --- |
| a. | Transferir Fundos | Cliente |
| b. | Comprar Livros | Usuário |
| c. | Obter Relatório de Vendas | Gerência |
| d. | Abrir Estadia | Hóspede |

**Exercício 4-22**: desenhe diagramas de casos de uso para os seguintes sistemas:

1. A biblioteca de sua universidade.
2. O seu aparelho celular.
3. Um sistema de validação de cartões de crédito.

**Exercício 4-23**: suponha que exista um caso de uso Pagar Pedido em um sistema, que é realizado pelo ator Cliente. Neste caso de uso, o cliente realiza o pagamento de um pedido realizado em algum momento do passado. Considerando este caso de uso, você pode pensar em algum outro caso de uso do sistema?

**Exercício 4-24**: considere o modelo de casos de uso inicial para o Sistema de Controle Acadêmico (Seção 4.7.3). Modifique esse modelo para contemplar as seguintes novidades:

a. O coordenador informa à equipe de desenvolvimento que há datas inicial e final pré-estabelecidas dentro de um semestre para que um professor possa lançar notas ou fornecer sua disponibilidade de carga horária para semestre letivo seguinte. E o próprio coordenador que deve estabelecer essas datas.

b. Da mesma forma, há um período para realização de inscrições e outro para cancelamentos das mesmas. Fora desses períodos, o sistema não deve aceitar tais operações. O coordenador também deve ter a possibilidade de definir esses períodos.

c. O coordenador declara que precisa ser informado pelo sistema (por e-mail, por exemplo) quando este último cria uma nova lista de espera para uma determinada disciplina.

**Exercício 4-25:** considere novamente o Sistema de Controle Acadêmico. Suponha que o analista de sistemas identificou uma nova regra de negócio, descrita a seguir:

Inscrição em Projeto Final (RN07)

Descrição: Para se inscrever na disciplina de "Projeto Final de Curso", o aluno precisa ter, no mínimo, 100 créditos concluídos.

Para contemplar essa regra do negócio, o analista de sistemas resolveu criar um caso de uso Realizar Inscrição em Projeto Final e fazê-lo herdeiro do caso de uso Realizar Inscrição, através de um relacionamento de generalização. Veja a figura a seguir. Discuta essa solução. Ela é correta? Forneça a descrição do caso de uso herdeiro.

