

Capítulo 5 – Modelagem de Casos de Uso

A análise de requisitos é um processo que envolve a construção de diversos modelos. Diagramas de classes e diagramas de interação incluem detalhes relacionados à estrutura interna dos objetos, suas associações, como eles interagem dinamicamente e como invocam o comportamento dos demais. Essas informações são necessárias para projetar e construir um sistema, mas não são suficientes para comunicar requisitos com clientes e usuários. Elas não capturam o conhecimento sobre as tarefas a serem realizadas e, portanto, é difícil avaliar se o sistema a ser construído a partir de um modelo desse tipo, isoladamente, vai realmente atender às necessidades dos usuários.

Assim, o primeiro modelo a ser construído deve ser passível de compreensão tanto por desenvolvedores – analistas, projetistas, programadores e testadores – como pela comunidade usuária – clientes e usuários. Esse modelo inicial deve descrever o sistema, seu ambiente e como sistema e ambiente estão relacionados.

Modelos de caso de uso (*use cases*) são uma forma de estruturar essa visão. Como o próprio nome sugere, um caso de uso é uma maneira de usar o sistema. Usuários interagem com o sistema, interagindo com seus casos de uso. Tomados em conjunto, os casos de uso de um sistema definem a sua funcionalidade. Casos de uso são, portanto, os “itens” que o desenvolvedor negocia com seus clientes.

O propósito do modelo de casos de uso é capturar e descrever a funcionalidade que um sistema deve prover. Um sistema geralmente serve a vários atores, para os quais ele provê diferentes serviços. Tipicamente, a funcionalidade a ser provida por um sistema é muito grande para ser analisada como uma única unidade e, portanto, é importante ter um mecanismo de dividir essa funcionalidade em partes menores e mais gerenciáveis. O conceito de caso de uso é muito útil para esse propósito (OLIVÉ, 2007).

É importante ter em mente que modelos de casos de uso são fundamentalmente uma ferramenta textual. Ainda que casos de uso sejam também descritos graficamente (p.ex., fluxogramas ou algum diagrama da UML, dentre eles diagramas de casos de uso, diagramas de sequência e diagramas de atividades), não se deve perder de vista a natureza textual dos modelos de casos de uso. Olhando casos de uso apenas a partir da UML, que não trata do conteúdo ou da escrita de casos de uso, pode-se pensar, equivocadamente, que casos de uso são uma construção gráfica ao invés de textual. Em essência, casos de uso servem como um meio de comunicação entre pessoas, algumas delas sem nenhum treinamento especial e, portanto, o uso de texto para especificar casos de uso é geralmente a melhor escolha. Casos de uso são amplamente usados no desenvolvimento de sistemas, porque, por meio sobretudo de suas descrições textuais, usuários e clientes conseguem visualizar qual a funcionalidade a ser provida pelo sistema, conseguindo reagir mais rapidamente no sentido de refinar, alterar ou rejeitar as funções previstas para o sistema (COCKBURN, 2005). Assim, um modelo de casos de uso inclui duas partes principais: (i) os diagramas de casos de uso e (ii) as descrições de

atores e de casos de uso, sendo que essas últimas podem ser complementadas com outros diagramas associados, tais como os diagramas de atividade e de sequência da UML⁶.

Outro aspecto a ser realçado é que os modelos de caso de uso são independentes do método de análise a ser usado e até mesmo do paradigma de desenvolvimento. Assim, pode-se utilizar a modelagem de casos de uso tanto no contexto do desenvolvimento orientado a objetos (foco deste texto), como em projetos desenvolvidos segundo o paradigma estruturado. De fato, o uso de modelos de caso de uso pode ser ainda mais amplo. Casos de uso podem ser usados, por exemplo, para documentar processos de negócio de uma organização. Contudo, neste texto, explora-se a utilização de casos de uso para modelar e documentar requisitos funcionais de sistemas. Assim, geralmente são interessados⁷ (*stakeholders*) nos casos de uso: as pessoas que usarão o sistema (usuários), o cliente que requer o sistema, outros sistemas com os quais o sistema em questão terá de interagir e outros membros da organização (ou até mesmo de fora dela) que têm restrições que o sistema precisa garantir.

Este capítulo aborda a técnica de modelagem de casos de uso, discutindo os principais elementos de modelos de casos de uso. A Seção 5.1 discute os dois principais conceitos empregados na modelagem de casos de uso: atores e casos de uso. A Seção 5.2 aborda os diagramas de casos de uso e sua notação segundo a UML. A Seção 5.3 trata da especificação de casos de uso. A Seção 5.4 discute os tipos de relacionamentos que podem ser estabelecidos entre casos de uso, a saber: inclusão, extensão e generalização / especialização. Finalmente, a Seção 5.5 discute como trabalhar com casos de uso e como usá-los em outras atividades do processo de software.

5.1 – Atores e Casos de Uso

Nenhum sistema computacional existe isoladamente. Todo sistema interage com atores humanos ou outros sistemas, que utilizam esse sistema para algum propósito e esperam que o sistema se comporte de certa maneira. Um caso de uso especifica um comportamento de um sistema segundo uma perspectiva externa e é uma descrição de uma sequência de ações realizada pelo sistema para produzir um resultado de valor para um ator (BOOCH; RUMBAUGH; JACOBSON, 2006).

Segundo Cockburn (2005), um caso de uso captura um contrato entre os interessados (*stakeholders*) em um sistema sobre o seu comportamento. Um caso de uso descreve o comportamento do sistema sob certas condições, em resposta a uma requisição feita por um interessado, dito o ator primário do caso de uso. Assim, os dois principais conceitos da modelagem de casos de uso são atores e casos de uso.

5.1.1 - Atores

Dá-se nome de ator a um papel desempenhado por entidades físicas (pessoas ou outros sistemas) que interagem com o sistema em questão da mesma maneira, procurando atingir os mesmos objetivos. Uma mesma entidade física pode desempenhar diferentes papéis no mesmo sistema, bem como um dado papel pode ser desempenhado por diferentes entidades (OLIVÉ, 2007).

⁶ O uso de diagramas de atividade e de sequência para complementar a especificação de um caso de uso é discutido no Capítulo 7.

⁷ Alguém ou algo com interesse no comportamento do sistema sob discussão (COCKBURN, 2005).

Atores são externos ao sistema. Um ator se comunica diretamente com o sistema, mas não é parte dele. A modelagem dos atores ajuda a definir as fronteiras do sistema, isto é, o conjunto de atores de um sistema delimita o ambiente externo desse sistema, representando o conjunto completo de entidades para as quais o sistema pode servir (BLAHA; RUMBAUGH, 2006; OLIVÉ, 2007).

Uma dúvida que sempre passa pela cabeça de um iniciante em modelagem de casos de uso é saber se o ator é a pessoa que efetivamente opera o sistema (p.ex., o atendente de uma locadora de automóveis) ou se é a pessoa interessada no resultado do processo (p.ex., o cliente que efetivamente loca o automóvel e é atendido pelo atendente). Essa definição depende, em essência, da fronteira estabelecida para o sistema. Sistemas de informação podem ter diferentes níveis de automatização. Por exemplo, se um sistema roda na Internet, seu nível de automatização é maior do que se ele requer um operador. Assim, é importante capturar qual o nível de automatização requerido e levar em conta o real limite do sistema (WAZLAWICK, 2004). Se o caso de uso roda na Internet (p.ex., um caso de uso de reserva de automóvel), então o cliente é o ator efetivamente. Se o caso de uso requer um operador (p.ex., um caso de uso de locação de automóvel, disponível apenas na locadora e para ser usado por atendentes), então o operador é o ator.

Quando se for considerar um sistema como sendo um ator, deve-se tomar o cuidado para não confundir a ideia de sistema externo (ator) com produtos usados na implementação do sistema em desenvolvimento. Para que um sistema possa ser considerado um ator, ele deve ser um sistema de informação completo (e não apenas uma biblioteca de classes, por exemplo). Além disso, ele deve estar fora do escopo do desenvolvimento do sistema atual. O analista não terá a oportunidade de alterar as funções do sistema externo, devendo adequar a comunicação às características do mesmo (WAZLAWICK, 2004).

Um ator primário é um ator que possui metas a serem cumpridas através do uso de serviços do sistema e que, tipicamente, inicia a interação com o sistema (OLIVÉ, 2007). Um ator secundário é um ator que interage com o sistema para prover um serviço para este último. A identificação de atores secundários é importante, uma vez que ela permite identificar interfaces externas que o sistema usará e os protocolos que regem as interações ocorrendo através delas (COCKBURN, 2005).

De maneira geral, o ator primário é o usuário direto do sistema ou outro sistema computacional que requisita um serviço do sistema em desenvolvimento. O sistema responde à requisição procurando atendê-la, ao mesmo tempo em que protege os interesses de todos os demais interessados no caso de uso. Entretanto, há situações em que o iniciador do caso de uso não é o ator primário. O tempo, por exemplo, pode ser o acionador de um caso de uso. Um caso de uso que roda todo dia à meia-noite ou ao final do mês tem o tempo como acionador. Mas o caso de uso ainda visa atingir um objetivo de um ator e esse ator é considerado o ator primário do caso de uso, ainda que ele não interaja efetivamente com o sistema (COCKBURN, 2005).

Para nomear atores, recomenda-se o uso de substantivos no singular, iniciados com letra maiúscula, possivelmente combinados com adjetivos. Exemplos: Cliente, Bibliotecário, Correntista, Correntista Titular etc.

5.1.2 – Casos de Uso

Um caso de uso é uma porção coerente da funcionalidade que um sistema pode fornecer para atores interagindo com ele (BLAHA; RUMBAUGH, 2006). Um caso de uso corresponde a um conjunto de ações realizadas pelo sistema (ou por meio da interação com o sistema), que produz um resultado observável, com valor para um ou mais atores do sistema. Geralmente, esse valor é a realização de uma meta de negócio ou tarefa (OLIVÉ, 2007). Assim, um caso de uso captura alguma função visível ao ator e, em especial, busca atingir uma meta desse ator.

Deve-se considerar que um caso de uso corresponde a uma transação completa, ou seja, um usuário poderia ativar o sistema, executar o caso de uso e desativar o sistema logo em seguida, e a operação estaria completa e consistente e atenderia a uma meta desse usuário (WAZLAWICK, 2004).

Ser uma transação completa é uma característica essencial de um caso de uso⁸, pois somente transações completas são capazes de atingir um objetivo do usuário. Casos de uso que necessitam de múltiplas sessões não passam nesse critério e devem ser divididos em casos de uso menores. Seja o exemplo de um caso de uso de concessão de empréstimo. Inicialmente, um atendente interagindo com um cliente informa os dados necessários para a avaliação do pedido de empréstimo. O pedido de empréstimo é, então, enviado para análise por um analista de crédito. Uma vez analisado e aprovado, o empréstimo é concedido, quando o dinheiro é entregue ao cliente e um contrato é assinado, dentre outros. Esse processo pode levar vários dias e não é realizado em uma sessão única. Assim, o caso de uso de concessão de empréstimo deveria ser subdividido em casos de uso menores, tais como casos de uso para efetuar pedido de empréstimo, analisar pedido de empréstimo e formalizar concessão de empréstimo.

Por outro lado, casos de uso muito pequenos, que não caracterizam uma transação completa, devem ser considerados passos de um caso de uso maior⁹. Seja o exemplo de uma biblioteca a qual cobra multa na devolução de livros em atraso. Um caso de uso específico para apenas calcular o valor da multa não é relevante, pois não caracteriza uma transação completa capaz de atingir um objetivo do usuário. O objetivo do usuário é efetuar a devolução e, neste contexto, uma regra de negócio (a que estabelece a multa) tem de ser levada em conta. Assim, calcular a multa é apenas um passo do caso de uso que efetua a devolução, o qual captura uma ação do sistema para garantir a regra de negócio e, portanto, satisfazer um interesse da biblioteca como organização.

Um caso de uso reúne todo o comportamento relevante de uma parte da funcionalidade do sistema. Isso inclui o comportamento principal normal, as variações de comportamento normais, as condições de exceção e o cancelamento de uma requisição. O conjunto de casos de uso captura a funcionalidade completa do sistema (BLAHA; RUMBAUGH, 2006).

Casos de uso fornecem uma abordagem para os desenvolvedores chegarem a uma compreensão comum com os usuários finais e especialistas do domínio, acerca da funcionalidade a ser provida pelo sistema (BOOCH; RUMBAUGH; JACOBSON, 2006).

⁸ Esta regra tem como exceção os casos de uso de inclusão e extensão, conforme discutido mais adiante na seção que trata de relacionamentos entre casos de uso.

⁹ As mesmas exceções da nota anterior aplicam-se aqui, conforme discutido mais adiante.

Os objetivos dos atores são um bom ponto de partida para a identificação de casos de uso. Pode-se propor um caso de uso para satisfazer cada um dos objetivos de cada um dos atores. A partir desses objetivos, podem-se estudar as possíveis interações do ator com o sistema e refinar o modelo de casos de uso.

Cada caso de uso tem um nome. Esse nome deve capturar a essência do caso de uso. Para nomear casos de uso sugere-se usar frases iniciadas com verbos no infinitivo, seguidos de complementos, que representem a meta ou tarefa a ser realizada com o caso de uso. As primeiras letras (exceto preposições) de cada palavra devem ser grafadas em letra maiúscula. Exemplos: Cadastrar Cliente, Devolver Livro, Efetuar Pagamento de Fatura etc.

Um caso de uso pode ser visto como um tipo cujas instâncias são cenários. Um cenário é uma execução de um caso de uso com entidades físicas particulares desempenhando os papéis dos atores e em um particular estado do domínio de informação. Um cenário, portanto, exercita um certo caminho dentro do conjunto de ações de um caso de uso (OLIVÉ, 2007).

Alguns cenários mostram o objetivo do caso de uso sendo alcançado; outros terminam com o caso de uso sendo abandonado (COCKBURN, 2005). Mesmo quando o objetivo de um caso de uso é alcançado, ele pode ser atingido seguindo diferentes caminhos. Assim, um caso de uso deve comportar todas essas situações. Para tal, um caso de uso é normalmente descrito por um conjunto de fluxos de eventos, capturando o fluxo de eventos principal, i.e., o fluxo de eventos típico que conduz ao objetivo do caso de uso, e fluxos de eventos alternativos, descrevendo exceções ou variantes do fluxo principal.

5.2 - Diagramas de Casos de Uso

Basicamente, um diagrama de casos de uso mostra um conjunto de casos de uso e atores e seus relacionamentos, sendo utilizado para ilustrar uma visão estática das maneiras possíveis de se usar o sistema (BOOCH; RUMBAUGH; JACOBSON, 2006).

Os diagramas de casos de uso da UML podem conter os seguintes elementos de modelo, ilustrados na Figura 5.1 (BOOCH; RUMBAUGH; JACOBSON, 2006):

- Assunto: o assunto delimita a fronteira de um diagrama de casos de uso, sendo normalmente o sistema ou um subsistema. Os casos de uso de um assunto descrevem o comportamento completo do assunto. O assunto é exibido em um diagrama de casos de uso como um retângulo envolvendo os casos de uso que o compõem. O nome do assunto (sistema ou subsistema) pode ser mostrado dentro do retângulo.
- Ator: representa um conjunto coerente de papéis que os usuários ou outros sistemas desempenham quando interagem com os casos de uso. Tipicamente, um ator representa um papel que um ser humano, um dispositivo de hardware ou outro sistema desempenha com o sistema em questão. Atores não são parte do sistema. Eles residem fora do sistema. Atores são representados por um ícone de homem, com o nome colocado abaixo do ícone.
- Caso de Uso: representa uma funcionalidade que o sistema deve prover. Casos de uso são parte do sistema e, portanto, residem dentro dele. Um caso de uso é representado por uma elipse com o nome do caso de uso dentro ou abaixo dela.

- Relacionamentos de Dependência, Generalização e Associação: são usados para estabelecer relacionamentos entre atores, entre atores e casos de uso, e entre casos de uso.

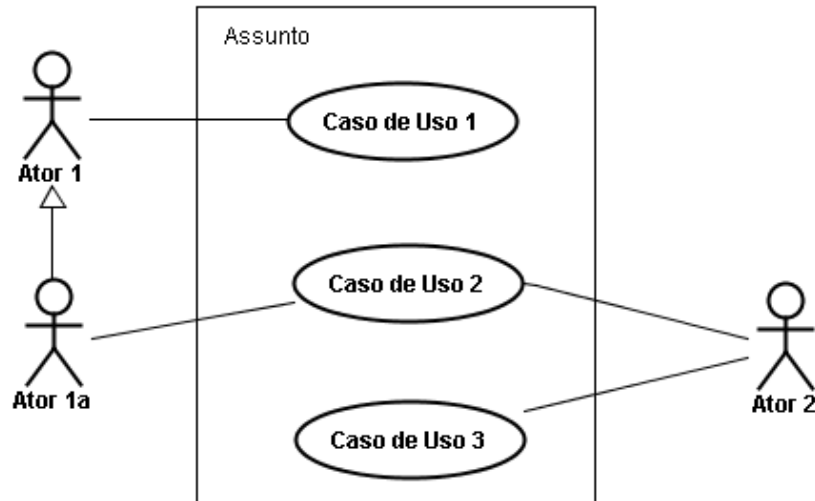


Figura 5.1 - Diagrama de Casos de Uso – Conceitos e Notação.

Atores só podem estar conectados a casos de uso por meio de associações. Uma associação entre um ator e um caso de uso significa que estímulos podem ser enviados entre atores e casos de uso. A associação entre um ator e um caso de uso indica que o ator e o caso de uso se comunicam entre si, cada um com a possibilidade de enviar e receber mensagens (BOOCH; RUMBAUGH; JACOBSON, 2006).

Atores podem ser organizados em hierarquias de generalização / especialização, de modo a capturar que um ator filho herda o significado e as associações com casos de uso de seu pai, especializando esse significado e potencialmente adicionando outras associações como outros casos de uso.

A Figura 5.2 mostra um diagrama de casos de uso para um sistema de caixa automático. Nesse diagrama, o assunto é o sistema como um todo. Os atores são: os clientes do banco, o sistema bancário e os responsáveis pela manutenção do numerário no caixa eletrônico. Cliente e mantenedor são atores primários, uma vez que têm objetivos a serem atingidos pelo uso do sistema. O sistema bancário é um ator secundário, pois o sistema do caixa automático precisa interagir com o sistema bancário para realizar os casos de uso *Efetuar Saque*, *Emitir Extrato* e *Efetuar Pagamento*.

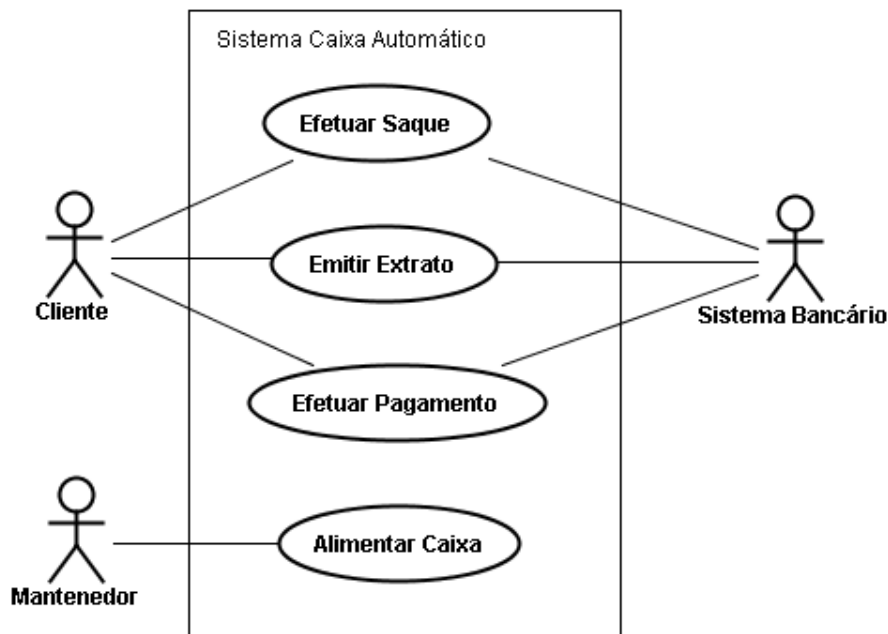


Figura 5.2 - Diagrama de Casos de Uso – Caixa Automático.

Um caso de uso descreve o que um sistema deve fazer. O diagrama de casos de uso provê uma visão apenas parcial disso, uma vez que mostra as funcionalidades por perspectiva externa. É necessário, ainda, capturar uma visão interna de cada caso de uso, especificando o comportamento do caso de uso pela descrição do fluxo de eventos que ocorre internamente (passos do caso de uso). Assim, uma parte fundamental do modelo de casos de uso é a descrição dos casos de uso.

5.3 - Descrevendo Casos de Uso

Um caso de uso deve descrever *o que* um sistema faz. Exceto para situações muito simples, um diagrama de casos de uso é insuficiente para este propósito. Assim, deve-se especificar o comportamento de um caso de uso pela descrição textual de seu fluxo de eventos, de modo que outros interessados possam compreendê-lo.

A especificação ou descrição de um caso de uso deve conter, dentre outras informações, um conjunto de sentenças, cada uma delas designando um passo simples, de modo que aprender a ler um caso de uso não requeira mais do que uns poucos minutos. Dependendo da situação, diferentes estilos de escrita podem ser adotados (COCKBURN, 2005).

Cada passo do fluxo de eventos de um caso de uso tipicamente descreve uma das seguintes situações: (i) uma interação entre um ator e o sistema, (ii) uma ação que o sistema realiza para atingir o objetivo do ator primário ou (iii) uma ação que o sistema realiza para proteger os interesses de um interessado. Essas ações podem incluir validações e mudanças do estado interno do sistema (COCKBURN, 2005).

Não há um padrão definido para especificar casos de uso. Diferentes autores propõem diferentes estruturas, formatos e conteúdos para descrições de casos de uso, alguns mais indicados para casos de uso essenciais e mais complexos, outros para casos de uso cadastrais e mais simples. Mais além, pode ser útil utilizar mais de um formato dentro do mesmo

projeto, em função das peculiaridades de cada caso de uso. De todo modo, é recomendável que a organização defina modelos de descrição de casos de uso a serem adotados em seus projetos, devendo definir tantos modelos quantos julgar necessários.

Cockburn (2005) recomenda que pelo menos dois modelos de descrição de casos de uso sejam definidos: um casual, escrito como um texto corrido livre, a ser usado em projetos com pouca formalidade; outro completo, com uma estrutura bem definida, para projetos de maior formalidade.

As seguintes informações são um bom ponto de partida para a definição de um modelo de descrição de casos de uso:

- Nome: nome do caso de uso, capturando a sua essência
- Escopo: diz respeito ao que está sendo documentado pelo caso de uso. Tipicamente pode ser um processo de negócio, um sistema ou um subsistema. Vale lembrar que este texto não aborda a utilização de casos de uso para a modelagem de processos de negócio. Assim, o escopo vai apontar o sistema / subsistema do qual o caso de uso faz parte.
- Descrição do Propósito: uma descrição sucinta do caso de uso, na forma de um único parágrafo, procurando descrever o objetivo do caso de uso.
- Ator Primário: nome do ator primário, ou seja, o interessado que tem um objetivo em relação ao sistema, o qual pode ser atingido pela execução do caso de uso.
- Interessados e Interesses: um interessado é alguém ou algo (um outro sistema) que tem um interesse no comportamento do caso de uso sendo descrito. Nesta seção são descritos cada um dos interessados no sistema e qual o seu interesse no caso de uso, incluindo o ator primário.
- Pré-condições: o que deve ser verdadeiro antes da execução do caso de uso. Se as pré-condições não forem satisfeitas, o caso de uso não pode ser realizado.
- Pós-condições: o que deve ser verdadeiro após a execução do caso de uso, considerando que o fluxo de eventos normal é realizado com sucesso.
- Fluxo de Eventos Normal: descreve os passos do caso de uso realizados em situações normais, considerando que nada acontece de errado e levando em conta a maneira mais comum do caso de uso ser realizado.
- Fluxo de Eventos Alternativos: descreve formas alternativas de realizar certos passos do caso de uso. Há duas formas alternativas principais: fluxos variantes, que são considerados dentro da normalidade do caso de uso; e fluxos de exceção, que se referem ao tratamento de erros durante a execução de um passo do fluxo normal (ou de um fluxo variante ou até mesmo de um outro fluxo de exceção).
- Requisitos Relacionados: listagem dos identificadores dos requisitos (funcionais, não funcionais e regras de negócio) tratados pelo caso de uso sendo descrito, de modo a permitir rastrear os requisitos. Casos de uso podem ser usados para conectar vários requisitos, de tipos diferentes. Assim, essa listagem ajuda a manter um rastro entre requisitos funcionais, não funcionais e regras de negócio, além de permitir verificar se algum requisito deixou de ser tratado.

- Classes / Entidades: classes (no paradigma orientado a objetos) ou entidades (no paradigma estruturado) necessárias para tratar o caso de uso sendo descrito. Esta seção é normalmente preenchida durante a modelagem conceitual estrutural e é igualmente importante para permitir rastrear requisitos para as etapas subsequentes do desenvolvimento (projeto e implementação, sobretudo).

5.3.1 – Descrevendo os Fluxos de Eventos

Uma vez que o conjunto inicial de casos de uso estiver estabilizado, cada um deles deve ser descrito em mais detalhes. Primeiro, deve-se descrever o fluxo de eventos principal (ou curso básico), isto é, o curso de eventos mais importante, que normalmente ocorre. O fluxo de eventos normal (ou principal) é uma informação essencial na descrição de um caso de uso e não pode ser omitido em nenhuma circunstância. O fluxo de eventos normal é, portanto, a principal seção de uma descrição de caso de uso, a qual descreve o processo quando tudo dá certo, ou seja, sem a ocorrência de nenhuma exceção (WAZLAWICK, 2004).

Variantes do curso básico de eventos e tratamento de exceções que possam vir a ocorrer devem ser descritos em cursos alternativos. Normalmente, um caso de uso possui apenas um único curso básico, mas diversos cursos alternativos. Seja o exemplo de um sistema de caixa automático de banco, cujo diagrama de casos de uso é mostrado na Figura 5.2. O caso de uso *Efetuar Saque* poderia ser descrito como mostrado na Figura 5.3.

Como visto nesse exemplo, um caso de uso pode ter um número de cursos alternativos que podem levar o caso de uso por diferentes caminhos. Tanto quanto possível, esses cursos alternativos, muitos deles cursos de exceção, devem ser identificados durante a especificação do fluxo de eventos normal de um caso do uso.

Vale realçar que uma exceção não é necessariamente um evento que ocorre muito raramente, mas sim um evento capaz de impedir o prosseguimento do caso de uso, se não for devidamente tratado. Uma exceção também não é algo que impede o caso de uso de ser iniciado, mas algo que impede a sua conclusão. Condições que impedem um caso de uso de ser iniciado devem ser tratadas como pré-condições. As pré-condições nunca devem ser testadas durante o processo do caso de uso, pois, por definição, elas impedem que o caso de uso seja iniciado. Logo, seria inconsistente imaginar que elas pudessem ocorrer durante a execução do caso de uso. Se uma pré-condição é falsa, então o caso de uso não pode ser iniciado (WAZLAWICK, 2004).

Observa-se que a maioria das exceções ocorre nos passos em que alguma informação é passada dos atores para o sistema. Isso porque, quando uma informação é passada para o sistema, muitas vezes ele realiza validações. Quando uma dessas validações falha, tipicamente ocorre uma exceção (WAZLAWICK, 2004).

Nome: Efetuar Saque

Escopo: Sistema de Caixa Automático

Descrição do Propósito: Este caso de uso permite que um cliente do banco efetue um saque, retirando dinheiro de sua conta bancária.

Ator Primário: Cliente

Interessados e Interesses:

- Cliente: deseja efetuar um saque.
- Banco: garantir que apenas o próprio cliente efetuará saques e que os valores dos saques sejam compatíveis com o limite de crédito do cliente.

Pré-condições: O caixa automático deve estar conectado ao sistema bancário.

Pós-condições: O saque é efetuado, debitando o valor da conta do cliente e entregando o mesmo valor para o cliente em espécie.

Fluxo de Eventos Normal

O cliente insere seu cartão no caixa automático, que analisa o cartão e verifica se ele é aceitável. Se o cartão é aceitável, o caixa automático solicita que o cliente informe a senha. O cliente informa a senha. O caixa automático envia os dados do cartão e da senha para o sistema bancário para validação. Se a senha estiver correta, o caixa solicita que o cliente informe o tipo de transação a ser efetuada. O cliente seleciona a opção saque e o caixa solicita que seja informada a quantia. O cliente informa a quantia a ser sacada. O caixa envia uma requisição para o sistema bancário para que seja efetuado um saque na quantia especificada. Se o saque é autorizado, as notas são preparadas e liberadas.

Fluxos de Eventos de Exceção

- O cartão não é aceitável: Se o cartão não é aceitável, seja porque sua tarja magnética não é passível de leitura seja porque é de um tipo incompatível, uma mensagem de erro de leitura é mostrada.
- Senha incorreta: Se a senha informada está incorreta, uma mensagem é mostrada para o cliente que poderá entrar com a senha novamente. Caso o cliente informe três vezes senha incorreta, o cartão deverá ser bloqueado.
- Saque não autorizado: Se o saque não for aceito pelo sistema bancário, uma mensagem de erro é exibida e a operação é abortada.
- Não há dinheiro suficiente disponível no caixa eletrônico: Uma mensagem de erro é exibida e a operação é abortada.
- Cancelamento: O cliente pode cancelar a transação a qualquer momento, enquanto o saque não for autorizado pelo sistema bancário.

Requisitos Relacionados: RF01, RN01, RNF01, RNF02¹⁰

Classes: Cliente, Conta, Cartão, Transação, Saque.

Figura 5.3 – Descrição do Caso de Uso *Efetuar Saque*.

¹⁰ São as seguintes as descrições dos requisitos listados: RF01 – O sistema de caixa automático deve permitir que clientes efetuem saques em dinheiro; RN01 – Não devem ser permitidas transações que deixem a conta do cliente com saldo inferior ao de seu limite de crédito; RNF01 – O sistema de caixa automático deve estar integrado ao sistema bancário; RNF02 – As operações realizadas no caixa automático devem dar respostas em até 10s a partir da entrada de dados.

Em sistemas de médio a grande porte, pode ser útil considerar a fusão de casos de uso fortemente relacionados em um único caso de uso, contendo mais de um fluxo de eventos normal. Em muitos sistemas é necessário dar ao usuário a possibilidade de cancelar ou alterar dados de uma transação efetuada anteriormente com sucesso. Se cada uma dessas possibilidades for considerada como um caso de uso isolado, o número de casos de uso pode crescer demasiadamente, aumentando desnecessariamente a complexidade do modelo de casos de uso. Além disso, o fluxo de eventos normal de um caso de uso desse tipo tende a ser muito simples, não justificando documentar todo um conjunto de informações para adicionar apenas duas ou três linhas descrevendo os passos do caso de uso. Assim, em situações dessa natureza, é interessante considerar apenas um caso de uso, contendo diversos fluxos de eventos principais. Essa abordagem é bastante recomendada para casos de uso cadastrais, em que um único caso de uso inclui fluxos de eventos normais para criar, alterar, consultar e excluir entidades.

Fluxos de eventos normais podem ser descritos de diferentes maneiras, dependendo do nível de formalidade que se deseja para as descrições. Dentre os formatos possíveis, há dois principais:

- Livre: o fluxo de eventos normal é escrito na forma de um texto corrido, como no exemplo da Figura 5.3;
- Enumerado: cada passo do fluxo de eventos normal é numerado, de modo que possa ser referenciado nos fluxos de eventos alternativos ou em outros pontos do fluxo de eventos normal. A Figura 5.4 reapresenta o exemplo da Figura 5.3 neste formato. As seções iniciais foram omitidas por serem iguais às da Figura 5.3. Neste texto, advogamos em favor do uso do formato enumerado.

Cada exceção deve ser tratada por um fluxo alternativo de exceção. Fluxos alternativos de exceção devem ser descritos contendo as seguintes informações (WAZLAWICK, 2004): um identificador, uma descrição sucinta da exceção que ocorreu, os passos para tratar a exceção (ações corretivas) e uma indicação de como o caso de uso retorna ao fluxo principal (se for o caso) após a execução das ações corretivas.

Quando um formato de descrição enumerado é utilizado, não é necessário colocar uma verificação como uma condicional no fluxo principal. Por exemplo, no caso da Figura 5.4, o passo 3 não deve ser escrito como “3. Se o cartão é válido, o caixa automático solicita que o cliente informe a senha.”. Basta o fluxo alternativo, no exemplo, o fluxo 2a.

Ainda quando o formato de descrição enumerado é utilizado, o identificador da exceção deve conter a linha do fluxo de eventos principal (ou eventualmente de algum outro fluxo de eventos alternativo) no qual a exceção ocorreu e uma letra para identificar a própria exceção (WAZLAWICK, 2004), como ilustra o exemplo da Figura 5.4.

Uma informação que precisa estar presente na descrição de um fluxo de eventos de exceção diz respeito a como finalizar o tratamento de uma exceção. Wazlawick (2004) aponta quatro formas básicas para finalizar o tratamento de uma exceção:

- Voltar ao início do caso de uso, o que não é muito comum nem prático.
- Voltar ao início do passo em que ocorreu a exceção e executá-lo novamente. Esta é a situação mais comum.
- Voltar para algum um passo posterior. Esta situação ocorre quando as ações corretivas realizam o trabalho que o passo (ou a sequência de passos) posterior

deveria executar. Neste caso, é importante verificar se novas exceções não poderiam ocorrer.

- Abortar o caso de uso. Neste caso, não se retorna ao fluxo principal e o caso de uso não atinge seus objetivos.

Nome: Efetuar Saque

Fluxo de Eventos Normal

1. O cliente insere seu cartão no caixa automático.
2. O caixa automático analisa o cartão e verifica se ele é aceitável.
3. O caixa automático solicita que o cliente informe a senha.
4. O cliente informa a senha.
5. O caixa automático envia os dados do cartão e da senha para o sistema bancário para validação.
6. O caixa automático solicita que o cliente informe o tipo de transação a ser efetuada.
7. O cliente seleciona a opção saque.
8. O caixa automático solicita que seja informada a quantia.
9. O cliente informa a quantia a ser sacada.
10. O caixa automático envia uma requisição para o sistema bancário para que seja efetuado um saque na quantia especificada.
11. As notas são preparadas e liberadas.

Fluxos de Eventos de Exceção

- 2a – O cartão não é aceitável: Se o cartão não é aceitável, seja porque sua tarja magnética não é passível de leitura seja porque é de um tipo incompatível, uma mensagem de erro de leitura é mostrada e se retorna ao passo 1.
- 5a – Senha incorreta:
- 5a.1 – 1ª e 2ª tentativas: Uma mensagem de erro é mostrada para o cliente. Retornar ao passo 3.
 - 5a.2 – 3ª tentativa: bloquear o cartão e abortar a transação.
- 10a - Saque não autorizado: Uma mensagem de erro é exibida e a operação é abortada.
- 11a - Não há dinheiro suficiente disponível no caixa eletrônico: Uma mensagem de erro é exibida e a operação é abortada.
- 1 a 9: Cancelamento: O cliente pode cancelar a transação, enquanto o saque não for autorizado pelo sistema bancário. A transação é abortada.

Figura 5.4 – Descrição do Caso de Uso *Efetuar Saque* – Formato Enumerado

Além dos fluxos de exceção, há outro tipo de fluxo de eventos alternativo: os fluxos variantes. Fluxos variantes são considerados dentro da normalidade do caso de uso e indicam formas diferentes, mas igualmente normais, de se realizar uma certa porção de um caso de uso. Seja o caso de um sistema de um supermercado, mais especificamente um caso de uso para efetuar uma compra. Um passo importante desse caso de uso é a realização do pagamento, o qual pode se dar de três maneiras distintas: pagamento em dinheiro, pagamento em cheque, pagamento em cartão. Nenhuma dessas formas de pagamento constitui uma exceção. São todas maneiras diferentes, mas normais, de realizar um certo passo do caso de

uso e, portanto, pode-se dizer que o fluxo principal possui três variações. A descrição de um fluxo variante deve conter: um identificador, uma descrição sucinta do passo especializado e os passos enumerados, como ilustra a Figura 5.5.

Nome: Efetuar Compra

Fluxo de Eventos Normal

...

1. De posse do valor a ser pago, o atendente informa a forma de pagamento.
2. Efetuar o pagamento:
 - 2a. Em dinheiro
 - 2b. Em cheque
 - 2c. Em cartão
3. O pagamento é registrado.

Fluxos de Eventos Variantes

2a – Pagamento em Dinheiro:

- 2a.1 – O atendente informa a quantia em dinheiro entregue pelo cliente.
- 2a.2 – O sistema informa o valor do troco a ser dado ao cliente.

2b – Pagamento em Cheque:

- 2b.1 – O atendente informa os dados do cheque, a saber: banco, agência, conta e valor.

2c – Pagamento em Cartão:

- 2c.1 – O atendente informa os dados do cartão e o valor da compra.
- 2c.2 – O sistema envia os dados informados no passo anterior, junto com a identificação da loja para o serviço de autorização do Sistema de Operadoras de Cartão de Crédito.
- 2c.3 – O Sistema de Operadoras de Cartão de Crédito autoriza a compra e envia o código da autorização.

Figura 5.5 – Descrição Parcial do Caso de Uso *Efetuar Compra* – com Variantes

Por fim, em diversas situações, pode ser desnecessariamente trabalhoso especificar casos de uso segundo um formato completo, seja usando uma descrição dos fluxos de eventos no formato livre seja no formato enumerado. Para esses casos, um formato simplificado, na forma de uma tabela, pode ser usado. O formato tabular é normalmente empregado para casos de uso que possuem uma estrutura de interação simples, seguindo uma mesma estrutura geral, tais como casos de uso cadastrais (ou CRUD¹¹) e consultas. Casos de uso cadastrais de baixa complexidade tipicamente envolvem inclusão, alteração, consulta e exclusão de entidades e seguem o padrão de descrição mostrado na Figura 5.6.

¹¹ CRUD – do inglês: Create, Read, Update and Delete; em português: Criar, Consultar, Atualizar e Excluir, ou seja, casos de uso que proveem as funções básicas de manipulação de dados de uma entidade de interesse do sistema.

Fluxos de Eventos Normais

Criar [Novo Objeto]

O [ator] informa os dados do [novo objeto], a saber: [atributos e associações do objeto]. Caso os dados sejam válidos, as informações são registradas.

Alterar Dados

O [ator] informa o [objeto] do qual deseja alterar dados e os novos dados. Os novos dados são validados e a alteração registrada.

Consultar Dados

O [ator] informa o [objeto] que deseja consultar. Os dados do [objeto] são apresentados.

Excluir [Objeto]

O [ator] informa o [objeto] que deseja excluir. Os dados do [objeto] são apresentados e é solicitada uma confirmação. Se a exclusão for confirmada, o [objeto] é excluído.

Fluxos de Eventos de Exceção

Incluir [Novo Objeto] / Alterar Dados

- Dados do [objeto] inválidos: uma mensagem de erro é exibida, solicitando correção da informação inválida.

Figura 5.6 – Padrão Típico de Descrição de Casos de Uso Cadastrais.

Assim, para simplificar a descrição de casos de uso cadastrais, recomenda-se utilizar o modelo tabular mostrado na Tabela 5.1. Quando essa tabela for empregada, estar-se-á assumindo que o caso de uso envolve os fluxos de eventos indicados (I para inclusão, A para alteração, C para consulta e E para exclusão), com a descrição base mostrada na Figura 5.5.

Tabela 5.1 – Modelo de Descrição de Casos de Uso Cadastrais

Caso de Uso	Ações Possíveis	Observações	Requisitos	Classes
<nome do caso de uso>	< I, A, C, E >			

A coluna **Observações** é usada para listar informações importantes relacionadas às ações, tais como os itens informados na inclusão, uma restrição a ser considerada para que a exclusão possa ser feita, uma informação que não pode ser alterada ou uma informação do objeto que não é apresentada na consulta. Deve-se indicar antes da observação a qual ação ela se refere ([I] para inclusão, [A] para alteração, [C] para consulta e [E] para exclusão).

As colunas **Requisitos** e **Classes** indicam, respectivamente, os requisitos que estão sendo (ou que devem ser) tratados pelo caso de uso e as classes do domínio do problema necessárias para a realização do caso de uso. O objetivo dessas colunas é manter a rastreabilidade dos casos de uso para requisitos e classes, respectivamente, de maneira análoga ao recomendado no formato completo.

A Tabela 5.2 ilustra a descrição de casos de usos cadastrais do subsistema Controle de Acervo de uma videolocadora.

Tabela 5.2 – Descrição de Casos de Uso Cadastrais – Controle de Acervo de Videolocadora

Caso de Uso	Ações Possíveis	Observações	Requisitos	Classes
Cadastrar Filme	I, A, C, E	[I] Informar: título original, título em português, país, ano, diretores, atores, sinopse, duração, gênero, distribuidora, tipo de áudio (p.ex., Dolby Digital 2.0), idioma do áudio e idioma da legenda. [E] Não é permitida a exclusão de filmes que tenham itens associados. [E] Ao excluir um filme, devem-se excluir as reservas associadas.	RF9, RNF1	Filme, Distribuidora
Cadastrar Item	I, A, C, E	[I] Informar: filme, tipo de mídia, data de aquisição e número de série. [E] Não é permitido excluir um item que tenha locações associadas.	RF9, RNF1, RNF3	Item, Filme, TipoMidia
Cadastrar Distribuidora	I, A, C, E	[I] Informar: razão social, CNPJ, endereço, telefone e pessoa de contato. [E] Não é permitido excluir uma distribuidora que tenha filmes associados.	RF10, RNF1	Distribuidora
Cadastrar Tipo de Mídia	I, A, C, E	[I] Informar: nome e valor de locação. [E] Não é permitido excluir um tipo de mídia que tenha itens associados. [E] Ao excluir um tipo de mídia, devem-se excluir as reservas que especificam apenas esse tipo de mídia.	RF9, RNF1	TipoMidia

Para casos de uso de consulta mais abrangente do que a consulta de um único objeto (já tratada como parte dos casos de uso cadastrais), mas ainda de baixa complexidade (tais como consultas que combinam informações de vários objetos envolvendo filtros), sugere-se utilizar o formato tabular mostrado na Tabela 5.3.

Tabela 5.3 – Modelo de Descrição de Casos de Uso de Consulta

Caso de Uso	Observações	Requisitos	Classes
<nome do caso de uso>			

A coluna **Observações** deve ser usada para listar informações importantes relacionadas à consulta, tais como dados que podem ser informados para a pesquisa, totalizações feitas em relatórios etc.

As colunas **Requisitos** e **Classes** têm a mesma função de suas homônimas no modelo da Tabela 5.1, ou seja, indicam, respectivamente, os requisitos que estão sendo tratados (ou que devem ser) pelo caso de uso e as classes do domínio do problema necessárias para a realização do mesmo.

A Tabela 5.4 ilustra a descrição de um caso de usos de consulta do subsistema Controle de Acervo de uma videolocadora.

Tabela 5.4 – Descrição de Casos de Uso de Consulta – Controle de Acervo de Videolocadora

Caso de Uso	Observações	Requisitos	Classes
Consultar Acervo	As consultas ao acervo poderão ser feitas informando uma (ou uma combinação) das seguintes informações: título (ou parte dele), original ou em português, gênero, tipo de mídia disponível, ator, diretor, nacionalidade e lançamentos.	RF11, RNF1, RNF2	Filme, Item, TipoMidia, Distribuidora

5.3.2 – Descrevendo Informações Complementares

As descrições dos fluxos de eventos principal, variantes e de exceção são cruciais em uma descrição de casos de uso. Contudo, há outras informações complementares que são bastante úteis e, portanto, que devem ser levantadas e documentadas. Conforme listado no início desta seção, são informações complementares importantes: atores, interessados e interesses, pré-condições, pós-condições, requisitos relacionados e classes relacionadas.

Conforme discutido na Seção 5.1, um ator representa um papel que entidades físicas ou sociais podem desempenhar na interação com o sistema. Essas entidades físicas são tipicamente pessoas, dispositivos ou outros sistemas que são externos ao sistema em desenvolvimento. Muitas vezes, apenas o nome de um ator, como mostrado em um diagrama de casos de uso, pode ser pouco para um real entendimento do que representa esse ator. Assim, é importante que uma descrição sucinta dos atores seja feita. Uma vez que um mesmo ator pode atuar em vários casos de uso, a descrição dos atores não deve ser feita dentro da descrição dos casos de uso, mas separada, como uma seção específica dentro do Documento de Especificação de Requisitos. Inicialmente, atores podem ser documentados em uma tabela de duas colunas, contendo o nome e a descrição do ator.

Como atores interagindo com o sistema definem as interfaces do sistema com o mundo externo, pode ser útil adicionar informações sobre o perfil do ator nessa interação. Quando o ator é um ator humano, esse perfil indicaria as habilidades e a experiência do ator, informações valiosas para o projeto da interface com o usuário. Adicionalmente, pode-se incluir uma classificação segundo aspectos como nível de habilidade, nível na organização e membros em diferentes grupos. Pressman (2006) propõe uma classificação que considera três grupos principais:

- *Usuário novato*: conhece pouco a interface para utilizá-la eficientemente (conhecimento sintático; p.ex., não sabe como atingir uma funcionalidade desejada) e entende pouco as funções e objetivos do sistema (conhece pouco a semântica da aplicação) ou não sabe bem como usar computadores em geral;
- *Usuário conhecedor e esporádico*: possui um conhecimento razoável da semântica da aplicação, mas tem relativamente pouca lembrança dos mecanismos de interação providos pela interface (informações sintáticas necessárias para utilizar a interface);
- *Usuário conhecedor e frequente*: possui bom conhecimento tanto sintático quanto semântico e buscam atalhos e modos abreviados de interação.

Não são apenas os atores os interessados em um caso de uso. Outras pessoas ou unidades de uma organização podem ter interesse nos resultados do caso de uso. Seja o caso de uma locadora de automóveis. Em um caso de uso de locação, o único papel a interagir com o sistema é o de funcionário do atendimento. Contudo, o cliente, o setor de preparação de automóveis, a contabilidade, dentre outros, são também interessados neste caso de uso. Assim, mesmo que essas pessoas não interajam diretamente com o sistema para a realização do caso de uso, elas devem ser listadas como interessados. Deve-se lembrar que o sistema deve satisfazer os interesses de todos os envolvidos, direta ou indiretamente. Assim, na seção “Interessados e Interesses”, deve-se listar os diversos interessados e uma descrição sucinta de seus interesses em relação à execução do caso de uso. Ao analisar esses interesses é possível, dentre outros, capturar regras de negócio e informações e descobrir ações que o sistema tem de realizar para atender a essas expectativas, tais como validações, atualizações e registros. (WAZLAWICK, 2004; COCKBURN, 2005).

Pré-condições estabelecem o que precisa ser verdadeiro antes de se iniciar um caso de uso. Pós-condições, por sua vez, estabelecem o que será verdadeiro após a execução do caso de uso. Pré-condições precisam ser verdadeiras para que o caso de uso possa ser iniciado. Não se deve confundi-las com exceções. Pré-condições não são testadas durante a execução do caso de uso (como ocorre com as condições que geram exceções). Ao contrário, elas são testadas antes de iniciar o caso de uso. Se a pré-condição é falsa, então não é possível executar o caso de uso. Para documentar as pré-condições, recomenda-se listar as condições que têm de ser satisfeitas na seção “Pré-condições”. Pré-condições devem ser escritas como uma simples asserção sobre o estado do mundo no momento em que o caso de uso inicia (COCKBURN, 2005).

Muitas vezes, uma pré-condição para ser atendida requer que um outro caso de uso já executado tenha estabelecido essa pré-condição. Contudo, um erro bastante comum é escrever como uma pré-condição algo que frequentemente, mas não necessariamente, é verdadeiro (COCKBURN, 2005). Seja o caso de uma locadora de vídeos em que clientes em atraso não podem locar novos itens até que regularize suas pendências. Neste caso, uma pré-condição do tipo “cliente não está em atraso” como pré-condição de um caso de uso “efetuar locação” é inadequada. Observe que a identificação do cliente é parte do caso de uso efetuar locação e, portanto, não é possível garantir que o cliente não está em atraso antes de iniciar o caso de uso. Esta situação tem de ser tratada como uma exceção e não como uma pré-condição.

As seções de requisitos e classes relacionados são importantes para a gerência de requisitos. A primeira estabelece um rastro entre casos de uso e os requisitos de usuário documentados no Documento de Requisitos, permitindo, em um primeiro momento, analisar se algum requisito não foi tratado. Em um segundo momento, quando uma alteração em um requisito é solicitada, é possível usar essa informação para analisar o impacto da alteração. Para documentar os requisitos relacionados, recomenda-se listar os identificadores de cada um dos requisitos na seção de “Requisitos Relacionados”.

A seção de classes relacionadas indica quais são as classes do modelo conceitual estrutural necessárias para a realização do caso de uso. Essa seção permite rastrear casos de uso para classes em vários níveis, uma vez que há uma grande tendência de as mesmas classes do modelo conceitual estrutural estarem presentes nos modelos de projeto e no código-fonte. Para documentar as classes relacionadas, recomenda-se listar o nome de cada uma das classes envolvidas na seção de “Classes Relacionadas”. Vale ressaltar que essa informação é tipicamente preenchida durante a modelagem conceitual estrutural ou até mesmo depois,

durante a elaboração de modelos de interação. A partir das informações de requisitos e classes relacionados, pode-se, por exemplo, construir matrizes de rastreabilidade.

5.4 - Relacionamentos entre Casos de Uso

Para permitir uma modelagem mais apurada dos casos de uso em um diagrama, três tipos de relacionamentos entre casos de uso podem ser empregados. Casos de uso podem ser descritos como versões especializadas de outros casos de uso (relacionamento de **generalização/ especialização**); casos de uso podem ser incluídos como parte de outro caso de uso (relacionamento de **inclusão**); ou casos de uso podem estender o comportamento de um outro caso de uso (relacionamento de **extensão**). O objetivo desses relacionamentos é tornar um modelo mais compreensível, evitar redundâncias entre casos de uso e permitir descrever casos de uso em camadas. A seguir esses tipos de relacionamentos são abordados.

5.4.1 - Inclusão

Uma associação de inclusão de um *caso de uso base* para um *caso de uso de inclusão* significa que o comportamento definido no caso de uso de inclusão é incorporado ao comportamento do caso de uso base. Ou seja, a relação de inclusão incorpora um caso de uso (o caso de uso incluído) dentro da sequência de comportamento de outro caso de uso (o caso de uso base) (BLAHA; RUMBAUGH, 2006; OLIVÉ, 2007).

Esse tipo de associação é útil para extrair comportamento comum a vários casos de uso em uma única descrição, de modo que esse comportamento não tenha de ser descrito repetidamente. O caso de uso de inclusão pode ou não ser passível de utilização isoladamente. Assim, ele pode ser apenas um fragmento de uma funcionalidade, não precisando ser uma transação completa. A parte comum é incluída por todos os casos de uso base que têm esse caso de uso de inclusão em comum e a execução do caso de uso de inclusão é análoga a uma chamada de subrotina (OLIVÉ, 2007).

Na UML, o relacionamento de inclusão entre casos de uso é mostrado como uma dependência (seta pontilhada) estereotipada com a palavra-chave *include*, partindo do caso de uso base para o caso de uso de inclusão, como ilustra a Figura 5.6.



Figura 5.6 – Associação de Inclusão na UML

Uma associação de inclusão deve ser referenciada também na descrição do caso de uso base. O local em que esse comportamento é incluído deve ser indicado na descrição do caso de uso base, através de uma referência explícita à chamada ao caso de uso incluído. Assim, a descrição do fluxo de eventos (principal ou alternativo) do caso de uso base deve conter um passo que envolva a chamada ao caso de uso incluído, referenciada por “Incluir nome do caso de uso incluído”. Para destacar referências de um caso de uso para outro, sugere-se que o nome do caso de uso referenciado seja sublinhado e escrito em itálico.

No exemplo do caixa automático, todos os três casos de uso têm em comum uma porção que diz respeito à validação inicial do cartão. Neste caso, um relacionamento de inclusão deve ser empregado, conforme mostra a Figura 5.7.

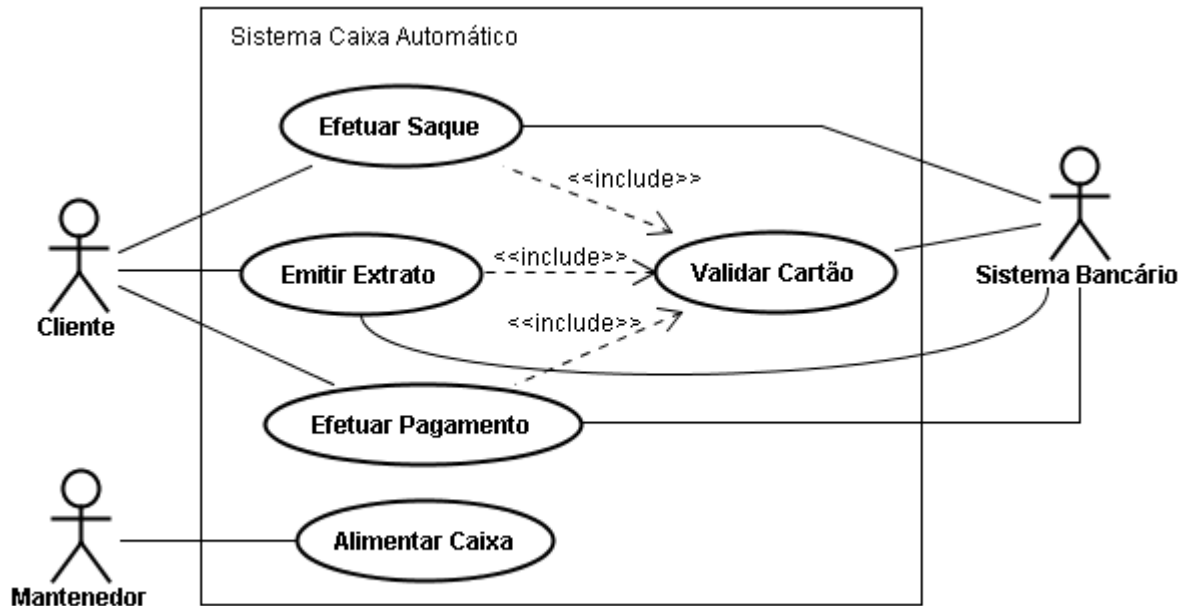


Figura 5.7 - Diagrama de Casos de Uso – Caixa Automático com Inclusão.

O caso de uso *Validar Cartão* extrai o comportamento descrito na Figura 5.8. Ao isolar este comportamento no caso de uso de *Validar Cliente*, o caso de uso *Efetuar Saque* passaria a apresentar a descrição mostrada na Figura 5.9.

Deve-se observar que não necessariamente o comportamento do caso de uso incluído precisa ser executado todas as vezes que o caso de uso base é realizado. Assim, é possível que a inclusão esteja associada a alguma condição. O caso de uso incluído é inserido em um local específico dentro da sequência do caso de uso base, da mesma forma que uma subrotina é chamada de um local específico dentro de outra subrotina (BLAHA; RUMBAUGH, 2006).

Nome: Validar Cartão

Fluxo de Eventos Normal

1. O cliente insere o cartão no caixa automático.
2. O caixa automático analisa o cartão e verifica se ele é aceitável.
3. O caixa automático solicita que o cliente informe a senha.
4. O cliente informa a senha.
5. O caixa automático envia os dados do cartão e da senha para o sistema bancário para validação.
6. O caixa automático solicita que o cliente informe o tipo de transação a ser efetuada.

Fluxos de Eventos de Exceção

- 2a – O cartão não é aceitável: Se o cartão não é aceitável, seja porque sua tarja magnética não é passível de leitura seja porque é de um tipo incompatível, uma mensagem de erro de leitura é mostrada e se retorna ao passo 1.
- 5a – Senha incorreta:
- 5a.1 – 1ª e 2ª tentativas: Uma mensagem de erro é mostrada para o cliente. Retornar ao passo 3.
- 5a.2 – 3ª tentativa: bloquear o cartão e abortar a transação.
- 1 a 5: Cancelamento: O cliente solicita o cancelamento da transação e a transação é abortada.

Figura 5.8 – Descrição do Caso de Uso *Validar Cartão*

Nome: Efetuar Saque

Fluxo de Eventos Normal

1. Incluir *Validar Cartão*.
2. O cliente seleciona a opção saque.
3. O caixa automático solicita que seja informada a quantia.
4. O cliente informa a quantia a ser sacada.
5. O caixa automático envia uma requisição para o sistema bancário para que seja efetuado um saque na quantia especificada.
6. As notas são preparadas e liberadas.

Fluxos de Eventos de Exceção

- 5a - Saque não autorizado: Uma mensagem de erro é exibida e a operação é abortada.
- 6a - Não há dinheiro suficiente disponível no caixa eletrônico: Uma mensagem de erro é exibida e a operação é abortada.
- 1 a 3: Cancelamento: O cliente pode cancelar a transação, enquanto o saque não for autorizado pelo sistema bancário. A transação é abortada.

Figura 5.9 – Descrição do Caso de Uso *Efetuar Saque com inclusão*.

Por fim, é importante frisar que não há um consenso sobre a possibilidade (ou não) de um caso de uso incluído poder ser utilizado isoladamente. Diversos autores, dentre eles Olivé (2007) e Blaha e Rumbaugh (2006), admitem essa possibilidade; outros não. Em (BOOCH; RUMBAUGH; JACOBSON, 2006), diz-se explicitamente que um “caso de uso incluído

nunca permanece isolado, mas é apenas instanciado como parte de alguma base maior que o inclui”. Neste texto, admitimos a possibilidade de um caso de uso incluído poder ser utilizado isoladamente, pois isso permite representar situações em que um caso de uso chama outro caso de uso (como uma chamada de subrotina), mas este último pode também ser realizado isoladamente. A Figura 5.10 ilustra uma situação bastante comum, em que, ao se realizar um processo de negócio (no caso a reserva de um carro em um sistema de locação de automóveis), caso uma informação necessária para esse processo (no caso o cliente) não esteja disponível, ela pode ser inserida no sistema. Contudo, o cadastro da informação também pode ser feito dissociado do processo de negócio que o inclui (no caso, o cliente pode se cadastrar fora do contexto da reserva de um carro). Ao não se admitir a possibilidade de um caso de uso incluído poder ser utilizado isoladamente, não é possível modelar situações desta natureza, as quais são bastante frequentes.

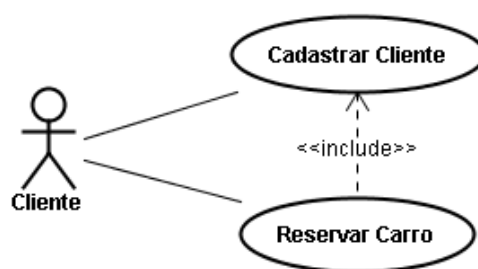


Figura 5.10 – Exemplo de Associação de Inclusão.

5.4.2 - Extensão

Uma associação de extensão entre um *caso de uso de extensão* e um *caso de uso base* significa que o comportamento definido no caso de uso de extensão pode ser inserido dentro do comportamento definido no caso de uso base, em um local especificado indiretamente pelo caso de uso de extensão. A extensão ocorre em um ou mais pontos de extensão específicos definidos no caso de uso base. A extensão pode ser condicional. Neste caso, a extensão ocorre apenas se a condição é verdadeira quando o ponto de extensão especificado é atingido. O caso de uso base é definido de forma independente do caso de uso de extensão e é significativo independentemente do caso de uso de extensão (OLIVÉ, 2007; BOOCH; RUMBAUGH; JACOBSON, 2006).

Um caso de uso pode ter vários pontos de extensão e esses pontos são referenciados por seus nomes (BOOCH; RUMBAUGH; JACOBSON, 2006). O caso de uso base apenas indica seus pontos de extensão. O caso de uso de extensão especifica em qual ponto de extensão ele será inserido. Por isso, diz-se que o caso de uso de extensão especifica indiretamente o local onde seu comportamento será inserido.

A associação de extensão é como uma relação de inclusão olhada da direção oposta, em que a extensão se incorpora ao caso de uso base, em vez de o caso de uso base incorporar explicitamente a extensão. Ela conecta um caso de uso de extensão a um caso de uso base. O caso de uso de extensão é geralmente um fragmento, ou seja, ele não aparece sozinho como uma sequência de comportamentos. Além disso, na maioria das vezes, a relação de extensão possui uma condição associada e, neste caso, o comportamento de extensão ocorre apenas se a condição for verdadeira. O caso de uso base, por sua vez, precisa ser, obrigatoriamente, um caso de uso válido na ausência de quaisquer extensões (BLAHA; RUMBAUGH, 2006).

Na UML, a associação de extensão entre casos de uso é mostrada como uma dependência (seta pontilhada) estereotipada com a palavra chave *extend*, partindo do caso de uso de extensão para o caso de uso base, como ilustra a Figura 5.11. Pontos de extensão podem ser indicados no compartimento da elipse do caso de uso, denominado “*extension points*” (pontos de extensão). Opcionalmente, a condição a ser satisfeita e a referência ao ponto de extensão podem ser mostradas por meio de uma nota¹² anexada à associação de extensão (OLIVÉ, 2007). Assim, no exemplo da Figura 5.11, o *Caso de Uso de Extensão 1* é executado quando o *ponto de extensão 1* do *Caso de Uso Base* for atingido, se a *condição* for verdadeira.

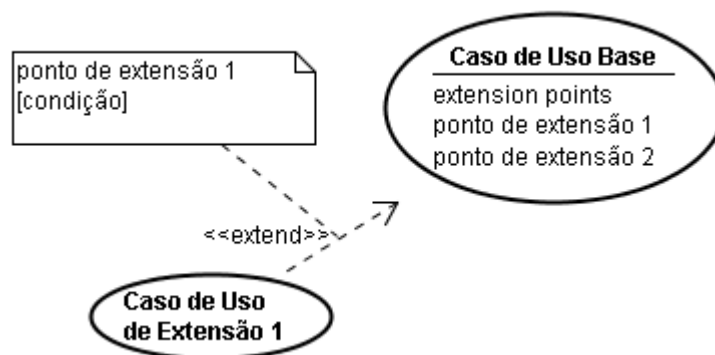


Figura 5.11 – Associação de Extensão na UML

Uma importante diferença entre as associações de inclusão e extensão é que, na primeira o caso de uso base está ciente do caso de uso de inclusão, enquanto na segunda o caso de uso base não está ciente dos possíveis casos de uso de extensão (OLIVÉ, 2007).

Assim como no caso da inclusão, uma associação de extensão deve ser referenciada na descrição do caso de uso base. Neste caso, contudo, o caso de uso base apenas aponta o ponto de extensão, sem fazer uma referência explícita ao caso de uso de extensão. O local de cada um dos pontos de extensão deve ser indicado na descrição do caso de uso base, através de uma referência ao nome do ponto de extensão seguido de “: ponto de extensão”. Assim, a descrição do fluxo de eventos (principal ou alternativo) do caso de uso base deve conter indicações explícitas para cada ponto de extensão.

No exemplo do caixa automático, suponha que se deseja coletar dados estatísticos sobre os valores das notas entregues nos saques, de modo a permitir alimentar o caixa eletrônico com as notas mais adequadas para saque. Poder-se-ia, então, estender o caso de uso *Efetuar Saque*, de modo que, quando necessário, outro caso de uso, denominado *Coletar Estatísticas de Notas*, contasse e acumulasse o tipo das notas entregues em um saque, conforme mostra a Figura 5.12. A Figura 5.13 mostra a descrição do caso de uso *Efetuar Saque* indicando o ponto de extensão *entrega do dinheiro*.

¹² Nota é o único item de anotação da UML. Notas são usadas para explicar partes de um modelo da UML. São comentários incluídos para descrever, esclarecer ou fazer alguma observação sobre qualquer elemento do modelo. Assim, uma nota é apenas um símbolo para representar restrições e comentários anexados a um elemento ou a uma coleção de elementos. Graficamente, uma nota é representada por um retângulo com um dos cantos com uma dobra de página, acompanhado por texto e anexada ao(s) elemento(s) anotados por meio de linha(s) pontilhada(s) (BOOCH; RUMBAUGH; JACOBSON, 2006). No exemplo da Figura 5.11, a nota está anexada ao relacionamento de extensão, adicionando-lhe informações sobre o ponto de extensão e a condição associados à extensão.

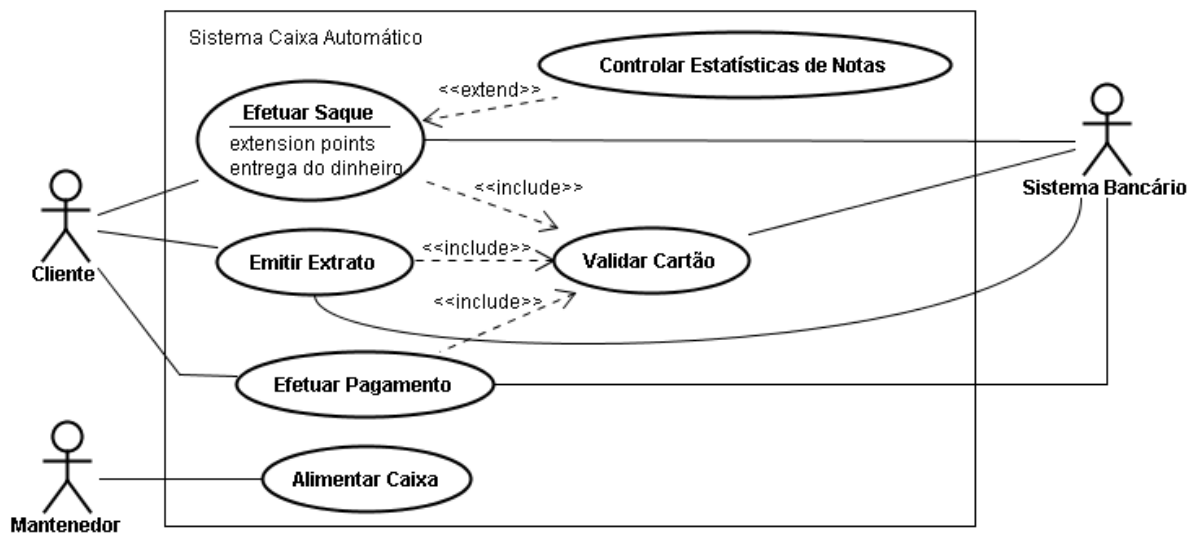


Figura 5.12 - Diagrama de Casos de Uso – Caixa Automático com Extensão.

Nome: Efetuar Saque

Fluxo de Eventos Normal

1. Incluir *Validar Cartão*.
2. O cliente seleciona a opção saque.
3. O caixa automático solicita que seja informada a quantia.
4. O cliente informa a quantia a ser sacada.
5. O caixa automático envia uma requisição para o sistema bancário para que seja efetuado um saque na quantia especificada.
6. As notas são preparadas.
- entrega do dinheiro: ponto de extensão.**
7. As notas são liberadas

Fluxos de Eventos de Exceção

- 5a - Saque não autorizado: Uma mensagem de erro é exibida e a operação é abortada.
- 6a - Não há dinheiro suficiente disponível no caixa eletrônico: Uma mensagem de erro é exibida e a operação é abortada.
- 1 a 3: Cancelamento: O cliente pode cancelar a transação, enquanto o saque não for autorizado pelo sistema bancário. A transação é abortada.

Figura 5.13 – Descrição do Caso de Uso *Efetuar Saque* com extensão.

5.4.3 – Generalização / Especialização

Um relacionamento de generalização / especialização entre um *caso de uso pai* e um *caso de uso filho* significa que o caso de uso filho herda o comportamento e o significado do caso de uso pai, acrescentando ou sobrescrevendo seu comportamento (OLIVÉ, 2007; BOOCH; RUMBAUGH; JACOBSON, 2006). Na UML, relacionamentos de generalização / especialização são representados como uma linha cheia direcionada com uma seta aberta (símbolo de herança), como ilustra a Figura 5.14.



Figura 5.14 – Associação de Generalização / Especialização entre Casos de Uso na UML

Voltando ao exemplo do sistema de caixa automático, suponha que haja duas formas adotadas para se validar o cartão: a primeira através de senha, como descrito anteriormente, e a segunda por meio de análise da retina do cliente. Neste caso, poderiam ser criadas duas especializações do caso de uso *Validar Cliente*, como mostra a Figura 5.15.

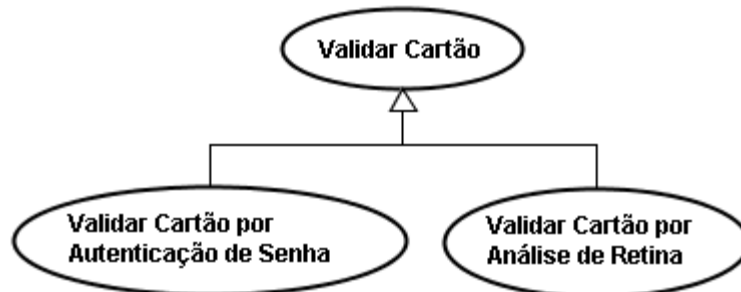


Figura 5.15 – Exemplo de Generalização / Especialização entre Casos de Uso

A descrição do caso de uso pai teria de ser generalizada para acomodar diferentes tipos de validação. Esses tipos de validação seriam especializados nas descrições dos casos de uso filhos. A Figura 5.16 mostra as descrições desses três casos de uso.

A generalização / especialização é aplicável quando um caso de uso possui diversas variações. O comportamento comum pode ser modelado como um caso de uso abstrato e especializado para as diferentes variações (BLAHA; RUMBAUGH, 2006). Contudo, avalie se não fica mais simples e direto descrever essas variações como fluxos alternativos variantes na descrição de casos de uso. Quando forem poucas e pequenas as variações, muito provavelmente será mais fácil capturá-las na descrição, ao invés de criar hierarquias de casos de uso. A Figura 5.17 mostra uma solução análoga à da Figura 5.16, sem usar, no entanto, especializações do caso de uso.

Nome: Validar Cartão

Fluxo de Eventos Normal

1. O cliente insere o cartão no caixa automático.
2. O caixa automático analisa o cartão e verifica se ele é aceitável.
3. O caixa automático solicita informação para identificação do cliente.
4. O cliente informa sua identificação.
5. O caixa automático envia os dados do cartão e da identificação para o sistema bancário para validação.
6. O caixa automático solicita que o cliente informe o tipo de transação a ser efetuada.

Fluxos de Eventos de Exceção

- 2a – O cartão não é aceitável: Se o cartão não é aceitável, seja porque sua tarja magnética não é passível de leitura seja porque é de um tipo incompatível, uma mensagem de erro de leitura é mostrada e se retorna ao passo 1.
- 5a – Dados de Identificação Incorretos:
- 5a.1 – 1ª e 2ª tentativas: Uma mensagem de erro é mostrada para o cliente. Retornar ao passo 3.
- 5a.2 – 3ª tentativa: bloquear o cartão e abortar a transação.
- 1 a 5: Cancelamento: O cliente solicita o cancelamento da transação e a transação é abortada.

Nome: Validar Cartão por Análise de Retina

Fluxo de Eventos Normal

3. O caixa automático solicita que o cliente se posicione corretamente para a captura da imagem da retina.
4. O caixa automático retira uma foto da retina do cliente.
5. O caixa automático envia os dados do cartão e a foto da retina para o sistema bancário para validação.

Nome: Validar Cartão por Autenticação de Senha

Fluxo de Eventos Normal

3. O caixa automático solicita a senha.
4. O cliente informa a senha.
5. O caixa automático envia os dados do cartão e a senha para o sistema bancário para validação.

Figura 5.16 – Descrição do Caso de Uso *Validar Cartão* e suas Especializações.

Nome: Validar Cartão

Fluxo de Eventos Normal

1. O cliente insere o cartão no caixa automático.
2. O caixa automático analisa o cartão e verifica se ele é aceitável.
3. Validar cartão.
4. O caixa automático solicita que o cliente informe o tipo de transação a ser efetuada.

Fluxos de Eventos Variantes

3a – Validar cartão por autenticação de senha:

3a.1 – O caixa automático solicita a senha.

3a.2 – O cliente informa a senha.

3a.3 – O caixa automático envia os dados do cartão e a senha para o sistema bancário para validação.

3b – Validar cartão por análise de retina:

3b.1 – O caixa automático solicita que o cliente se posicione corretamente para a captura da imagem da retina.

3b.2 – O caixa automático retira uma foto da retina do cliente.

3b.3 – O caixa automático envia os dados do cartão e a foto da retina para o sistema bancário para validação.

Fluxos de Eventos de Exceção

2a – O cartão não é aceitável: Se o cartão não é aceitável, seja porque sua tarja magnética não é passível de leitura seja porque é de um tipo incompatível, uma mensagem de erro de leitura é mostrada e se retorna ao passo 1.

5a – Dados de Identificação Incorretos:

5a.1 – 1ª e 2ª tentativas: Uma mensagem de erro é mostrada para o cliente. Retornar ao passo 3.

5a.2 – 3ª tentativa: bloquear o cartão e abortar a transação.

1 a 5: Cancelamento: O cliente solicita o cancelamento da transação e a transação é abortada.

Figura 5.17 – Descrição do Caso de Uso *Validar Cartão* com Variantes.

5.4.4 – Diretrizes para o Uso dos Tipos de Relacionamentos entre Casos de Uso

Os relacionamentos entre casos de uso devem ser utilizados com cuidado para evitar a introdução de complexidade desnecessária. As seguintes orientações são úteis para ajudar a decidir quando usar relacionamentos entre casos de uso em um diagrama de casos de uso:

- A inclusão é tipicamente aplicável quando se deseja capturar um fragmento de comportamento comum a vários casos de uso. Na maioria das vezes, o caso de uso de inclusão é uma atividade significativa, mas não como um fim em si mesma (BLAHA; RUMBAUGH, 2006). Ou seja, o caso de uso de inclusão não precisa ser uma transação completa.
- Um relacionamento de inclusão é empregado quando há uma porção de comportamento que é similar ao longo de um ou mais casos de uso e não se deseja repetir a sua descrição. Para evitar redundância e assegurar reúso, extrai-se essa descrição e se compartilha a mesma entre diferentes casos de uso. Desta maneira, utiliza-se a inclusão para evitar ter de descrever o mesmo fragmento de comportamento várias vezes, capturando o comportamento comum em um caso de uso próprio (BOOCH; RUMBAUGH; JACOBSON, 2006).
- Não se deve utilizar o relacionamento de generalização / especialização para compartilhar fragmentos de comportamento. Para este propósito, deve-se usar a relação de inclusão (BLAHA; RUMBAUGH, 2006).
- A relação de extensão é bastante útil em situações em que se pode definir um caso de uso significativo com recursos adicionais. O comportamento básico é capturado no caso de uso base e os recursos adicionais nos casos de uso de extensão. Use a relação de extensão quando o sistema puder ser usado em diferentes configurações, algumas com os recursos adicionais e outras sem eles (BLAHA; RUMBAUGH, 2006).
- Tanto a inclusão quanto a extensão podem ser usadas para dividir o comportamento em partes menores. A inclusão, entretanto, implica que o comportamento incluído é uma parte necessária de um sistema configurado, mesmo que seu comportamento não seja executado todas as vezes, ou seja, mesmo que o comportamento incluído esteja associado a uma condição. A extensão, por sua vez, implica que o sistema sem o comportamento adicionado pela extensão é significativo (BLAHA; RUMBAUGH, 2006).

5.5 – Trabalhando com Casos de Uso

Para se utilizar a modelagem de casos de uso para o refinamento de requisitos de usuário em requisitos de sistema é necessário proceder um exame detalhado do processo de negócio a ser apoiado pelo sistema. Assim, atividades de levantamento de requisitos, como entrevistas, observação, workshop de requisitos e cenários, dentre outras, certamente acontecerão em paralelo com a modelagem de casos de uso.

Uma boa maneira de trabalhar com casos de uso consiste em, a partir dos requisitos funcionais de usuário descritos no Documento de Requisitos, procurar derivar casos de uso. Este é apenas um ponto de partida, uma vez que vários casos de uso podem ser derivados a partir de um mesmo requisito funcional de usuário.

Uma maneira complementar de identificar casos de uso é começar pela identificação de atores. Cada ator deve ter um propósito único e coerente, o qual deve ser descrito e documentado. Para cada ator identificado, pode-se, então, levantar quais são as funcionalidades por ele requeridas, listando-as na forma de casos de uso. Cada caso de uso deve representar uma transação completa que seja algo de valor para os atores envolvidos. Contudo, antes de identificar atores e casos de uso, é necessário determinar claramente os limites do sistema. Sem deixar claro quais são os limites do sistema, é muito difícil identificar atores ou casos de uso (BLAHA; RUMBAUGH, 2006).

Uma vez identificados atores e casos de uso, pode-se elaborar uma versão preliminar do diagrama de casos de uso. Vale lembrar que, até mesmo para sistemas de pequeno porte, é útil trabalhar com subsistemas, procurando agrupar casos de uso em pacotes. Assim, é importante procurar agrupar casos de uso relacionados em pacotes, construindo também diagramas de pacotes à medida que os casos de uso vão sendo agrupados.

Uma vez identificados e agrupados os casos de uso, é interessante fazer uma descrição sucinta de seu propósito. Não se deve partir diretamente para os detalhes, descrevendo fluxos de eventos e outras informações. Fazendo apenas uma descrição sucinta, é possível levar mais rapidamente os casos de uso à discussão com os clientes e usuários, permitindo identificar melhor quais são efetivamente os casos de uso a serem contemplados pelo sistema. Além disso, pode-se dividir o trabalho, designando diferentes analistas para trabalhar com casos de uso (ou pacotes) específicos.

Somente então se deve passar para a descrição detalhada dos casos de uso. Inicialmente, o foco deve ser no fluxo de eventos principal, ou seja, aquele em que tudo dá certo na interação. Depois de descrever o fluxo de eventos normal, deve-se analisar de forma crítica cada passo desses fluxos de eventos, procurando verificar o que pode dar errado (WAZLAWICK, 2004), bem como se devem investigar maneiras alternativas, ainda normais, de realizar o caso de uso, permitindo a identificação de fluxos variantes. A partir da identificação de possíveis exceções e variações, deve-se trabalhar na descrição de fluxos alternativos de exceção (descrevendo procedimentos para contornar os problemas) e variantes (descrevendo maneiras alternativas de realizar com sucesso uma certa porção do caso de uso).

Assim, uma maneira adequada para trabalhar com casos de uso consiste em identificá-los, modelá-los e descrevê-los com diferentes níveis de precisão. O seguinte processo resume a abordagem descrita anteriormente:

1. Listar atores e casos de uso relacionados: neste momento, é montada apenas uma lista dos atores associados aos casos de uso de seu interesse. Apenas o nome do caso de uso é indicado.
2. Para cada caso de uso identificado, fazer uma descrição sucinta do mesmo. Essa descrição deve conter, em essência, o objetivo do caso de uso.
3. Elaborar um ou mais diagramas de casos de uso.
4. Revisar a exatidão e a completude do conjunto de casos de uso com os interessados e priorizar os casos de uso.
5. Definir o formato de descrição de caso de uso a ser usado (e o correspondente modelo de descrição de caso de uso a ser adotado) para cada caso de uso.

6. Definir os fluxos de eventos principais a serem comportados pelo caso de uso: de maneira análoga ao passo 1, apenas uma lista dos fluxos de eventos principais é elaborada, sem descrevê-los ainda.
7. Descrever cada um dos fluxos principais de eventos do caso de uso, segundo o modelo de descrição de caso de uso estabelecido no passo anterior. De acordo com o modelo predefinido, levantar informações adicionais como pré-condições e requisitos relacionados.
8. Identificar fluxos alternativos: neste momento, é levantada apenas uma lista de exceções e variações que podem ocorrer no fluxo principal de eventos do caso de uso, sem no entanto definir como o sistema deve tratá-las.
9. Descrever os passos dos fluxos alternativos: descrever como o sistema deve responder a cada exceção ou como ele deve funcionar em cada variação.

Vale ressaltar que a descrição de casos de uso na fase de análise de requisitos deve ser feita sem considerar a tecnologia de interface. Neste momento não interessa saber a forma das interfaces do sistema, mas quais informações são trocadas entre o sistema e o ambiente externo (atores). O analista deve procurar abstrair a tecnologia e se concentrar na essência das informações trocadas. Assim, diz-se que a descrição de caso de uso na fase de análise é uma descrição essencial. A tecnologia de interface será objeto da fase de projeto do sistema. Agindo dessa maneira, abre-se caminho para se pensar em diferentes alternativas de interfaces durante o projeto do sistema (WAZLAWICK, 2004).

Uma técnica de levantamento de requisitos bastante útil para apoiar a escrita de casos de uso são os cenários. Pode-se pedir para que o usuário descreva alguns cenários na forma de exemplos situados de um caso de uso em ação, mostrando o ator usando o sistema para realizar o caso de uso em questão (COCKBURN, 2005).

Um cenário é uma sequência específica de ações que ilustra o comportamento de um caso de uso. Assim, os cenários são, na verdade, instâncias de um caso de uso.

Os modelos de casos de uso são uma maneira eficaz para analistas, clientes, especialistas de domínio e usuários chegarem a uma compreensão comum acerca das funcionalidades que o sistema deve prover. Além disso, servem para ajudar a verificar e validar o sistema à medida que ele vai sendo desenvolvido. Neste contexto, os casos de uso podem ser utilizados como base para o projeto de casos de teste para o sistema, em uma abordagem de testes baseada em casos de uso, na qual casos de teste são projetados a partir dos fluxos de eventos principal e alternativos dos casos de uso, procurando explorar diferentes cenários de uso do sistema.

No contexto da Engenharia de Requisitos, casos de uso têm dois importantes papéis:

- *Casos de uso especificam os requisitos funcionais de um sistema.* Um modelo de caso de uso descreve detalhadamente o comportamento de um sistema através de um conjunto de casos de uso. O ambiente do sistema é definido pela descrição dos diferentes atores que utilizam o sistema realizando os casos de uso.
- *Casos de uso oferecem uma abordagem para a modelagem de sistemas.* Para gerenciar a complexidade de sistemas reais, é comum apresentar os modelos do sistema em um número de diferentes visões. Em uma abordagem guiada por casos de uso, pode-se construir uma visão para cada caso de uso, isto é, em cada visão são modelados apenas aqueles elementos que participam de um caso de uso específico.

Essa abordagem é especialmente útil para a modelagem comportamental feita utilizando diagramas de atividade e de sequência. Um particular elemento (uma classe, p.ex.) pode, é claro, participar de vários casos de uso. Isto significa que um modelo do sistema completo só é visto através de um conjunto de visões. Para se definir todas as responsabilidades de um elemento, deve-se olhar os casos de uso onde esse elemento tem um papel.

É importante destacar que a modelagem casos de uso pode (e deve) ser realizada com algum grau de paralelismo em relação à modelagem conceitual estrutural. A identificação de conceitos relevantes para tratar um caso de uso pode ajudar a descobrir outros casos de uso relevantes, sobretudo de natureza cadastral. Assim, uma vez iniciada a descrição dos casos de uso, a modelagem conceitual estrutural pode ser também iniciada.

Além de serem uma ferramenta essencial na especificação dos requisitos funcionais de um sistema, casos de uso têm um papel fundamental no planejamento e controle de projetos iterativos. Casos de uso podem ser usados para definir o escopo de uma iteração do projeto ou mesmo do projeto como um todo. Neste contexto, uma técnica interessante para administrar discussões de escopo são as listas dentro / fora (COCKBURN, 2005). Uma lista dentro / fora é, na verdade, uma tabela com três colunas. A primeira coluna enumera casos de uso; as duas outras colunas são rotuladas “Dentro” e “Fora”. Sempre que não for claro se um caso de uso está dentro ou fora do escopo da discussão (projeto ou iteração), ele é incluído na tabela e deve-se perguntar aos interessados se o caso de uso está dentro ou fora do escopo. Assim, é possível capturar as diferentes visões dos diferentes interessados, sendo essas visões muitas vezes conflitantes. Identificados conflitos, os mesmos devem ser negociados e resolvidos.

Ainda no que se refere ao planejamento e controle de projetos iterativos, pode ser uma boa estratégia priorizar os casos de uso, de modo a definir o que considerar ou não em uma iteração (ou mesmo no projeto). Para os casos de uso considerados dentro do escopo do projeto, pode-se indicar em qual versão o caso de uso deveria ser tratado. Por exemplo, se foram planejados três iterações para o desenvolvimento de um certo sistema, os interessados poderiam indicar em qual versão (1, 2 ou 3) cada caso de uso deveria ser tratado. Essas listas de prioridades são usadas como ponto de partida para a negociação e o planejamento das iterações do projeto.

Leitura Complementar

Os capítulos 7 e 8 de (BLAHA; RUMBAUGH, 2006) – *Modelagem de Interações e Modelagem Avançada de Interações*, respectivamente – abordam a modelagem de casos de uso. Mais especificamente, recomenda-se a leitura da seção 7.1 (*Modelos de Casos de Uso*), que dá uma visão geral de atores, casos de uso e diagramas de casos de uso, e da seção 8.1 (*Relações entre Casos de Uso*), que discute as relações de inclusão, extensão e generalização e especialização entre casos de uso.

O Capítulo 15 de (OLIVÉ, 2007) – *Use Cases* – dá uma visão geral da modelagem de casos de uso, discutindo de maneira breve, mas bastante didática, os conceitos de ator e de caso de uso, a especificação de casos de uso e os relacionamentos entre casos de uso.

O livro “*Escrevendo Casos de Uso Eficazes: Um guia prático para desenvolvedores de software*” (COCKBURN, 2005) é inteiramente dedicado ao processo de escrita de casos de uso. Esse livro é uma ótima referência para os interessados em aperfeiçoar seu processo de escrita de casos de uso, contendo diversas diretrizes incorporadas nestas notas de aula.

Em (WAZLAWICK, 2004), tanto o Capítulo 2 (*Concepção*) quanto o Capítulo 3 (*Expansão dos Casos de Uso*) abordam a modelagem de casos de uso.

Em (BOOCH; RUMBAUGH; JACOBSON, 2006), merecem atenção os capítulos 17 (*Casos de Uso*) e 18 (*Diagramas de Casos de Uso*). As notações da UML para diagramas de casos de uso são tratadas com mais detalhes do que nas demais referências citadas anteriormente, precisamente por se tratar este de um livro sobre a UML.

Referências do Capítulo

- BLAHA, M., RUMBAUGH, J., *Modelagem e Projetos Baseados em Objetos com UML 2*, Elsevier, 2006.
- BOOCH, G., RUMBAUGH, J., JACOBSON, I., *UML Guia do Usuário*, 2a edição, Elsevier Editora, 2006.
- COCKBURN, A., *Escrevendo Casos de Uso Eficazes: Um guia prático para desenvolvedores de software*, Porto Alegre: Bookman, 2005.
- OLIVÉ, A., *Conceptual Modeling of Information Systems*, Springer, 2007.
- WAZLAWICK, R.S., *Análise e Projeto de Sistemas de Informação Orientados a Objetos*, Elsevier, 2004.