

Guia para Modelagem de Casos de Uso
Metodologia CELEPAR

Agosto 2009

Sumário de Informações do Documento

Documento: guiaModelagemCasosUso.odt	**Número de páginas**: 14

Versão	**Data**	**Mudanças**	**Autor**
1.0	25/04/07	Criação do artefato	Ariel Bolzan Witczak
1.0	14/05/07	Revisão	Ariel e Danielle Mayer
1.0	31/05/07	Revisão	Ariel, Cleverson Budel, Danielle, Luciana Reis, Karina Curcio
1.0	25/05/09	Revisão de formataçãoGuia para Modelagem de Casos de Uso	Danielle Mayer

SUMÁRIO

1.INTRODUÇÃO....4

1.1 Motivação....4
1.2 Benefícios....4
1.3 Composição do Modelo de Caso de Uso....5

2.MODELAGEM DE CASOS DE USO....5

3.CASOS DE USO....6

4.NOMEANDO CASOS DE USO....6

5.CRIANDO UM MODELO DE CASOS DGUIA PARA MODELAGEM DE CASOS DE USOE USO....7

6.IDENTIFICANDO ATORES....7

7.IDENTIFICANDO CASOS DE USO....8

8.RELACIONAMENTOS....9

8.1 Generalização de Ator....9
8.2 Por que estruturar o modelo de Caso do Uso?....10
8.3 Por que usar um relacionamento <<include>>?....10
8.4 Porque usar um relacionamento <<extend>>?....11
8.5 Porque usar um relacionamento de generalização?....11
8.6 Porque usar um relacionamento de generalização de Ator?....11

9.DECOMPOSIÇÃO FUNCIONAL....11

9.1 Quando evitar a Decomposição Funcional....11

10.CHECKLIST....12

10.1 Para Casos de Uso....12
10.2 Para Atores....13
10.3 Para o Modelo:....14

1. INTRODUÇÃO

Este guia tem o objetivo de orientar a atuação dos Analistas de Requisitos no momento de realizar a Modelagem dos Casos de Uso, ou seja, modelar o contexto do sistema.

1.1 Motivação

Traduz a expectativa dos envolvidos no que concerne as funcionalidades da aplicação. Com a modelagem de Caso de Uso, evita-se problemas comuns enfrentados durante o desenvolvimento tais como:

- As necessidades do negócio não condizem com as necessidades dos usuários;
- Requisitos confusos;
- Falta de integração entre módulos;
- Dificuldade de manutenção;
- Descoberta tardia de falhas;
- Baixa qualidade ou experiência do usuário final;
- Baixa performance;
- Esforços não coordenados da equipe.

1.2 Benefícios

A modelagem de Caso de Uso ilustra as necessidades do usuário, facilita a compreensão geral, captura a estrutura e o comportamento do sistema e apresenta a integração que há entre os elementos, promovendo uma comunicação clara e concisa.

1.3 Composição do Modelo de Caso de Uso

Um modelo de Caso de Uso é formado por vários elementos, Figura 1, neste momento será citada a importância dos Atores e dos Casos de Uso, maiores detalhes serão descritos posteriormente.

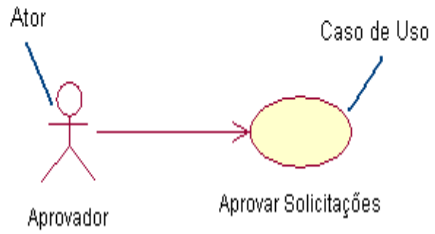


Figura 1: Modelo de Caso de Uso Básico

Ator: Alguém ou alguma coisa fora do sistema que interage com o mesmo.

Caso de Uso: O que um Ator quer que o sistema faça.

2. MODELAGEM DE CASOS DE USO

Os Casos de Uso mapeiam o comportamento do sistema, identificam quem e o quê interage com o sistema e também o que o sistema deve fazer.

O modelo de Caso de Uso (conjunto de Casos de Uso) é uma maneira de verificar todos os requisitos capturados e validar as futuras funcionalidades do sistema.

É correto afirmar que o modelo de Casos de Uso representa as futuras funcionalidades do sistema (Casos de Uso) e sua interação com o meio externo (Atores).

O modelo de Casos de Uso é utilizado em todas as fases do ciclo de desenvolvimento por todos os membros da equipe. Deve ser compreendido pela equipe de desenvolvimento (analistas, designers, desenvolvedores, equipe de teste) e também pelas pessoas externas ao projeto (clientes,

usuários).

3. CASOS DE USO

Um Caso de Uso define uma sequencia de ações executadas pelo sistema retornando valores que podem ser observados pelos Atores. Um Ator é um papel que um usuário desempenha em relação ao sistema.

Cada Caso de Uso descreve ações que o sistema deve executar para retornar algum valor para o Ator, demostra as funcionalidades do sistema utilizadas pelos Atores. O Caso de Uso representa um fluxo completo de eventos, diálogo entre os Atores e o sistema.

A diferença fundamental entre Ator e usuário é que o Ator representa um tipo particular de usuário. Por exemplo, temos os usuários do sistema de notas: Pedro e João, porém eles são representados pelo Ator Aluno.

O canal de comunicação entre Atores e Casos de Uso é chamado de associação.

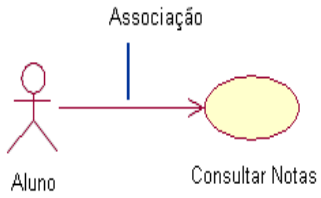


Figura 2: Comunicação entre Atores e Casos de Uso

Há vários tipos de associação que serão descritas posteriormente.

4. NOMEANDO CASOS DE USO

É importante que o nome do Caso de Uso seja representativo. Para isso deve:

- O nome deve ser único para todo sistema;
- Indicar qual o objetivo do Caso de Uso;
- Utilizar uma forma ativa: começar com um verbo;

- Imaginar uma lista de tarefas a serem executadas.

5. CRIANDO UM MODELO DE CASOS DE USO

Para encontrar os Atores e Casos de Uso do sistema, deve-se considerar os requisitos coletados junto aos clientes e usuários. Atores e Casos de Uso devem ser identificados e ilustrados no diagrama de Casos de Uso, assim obtém-se o primeiro esboço do fluxo.

Passos para criar um Modelo de Casos de Uso:

- Encontrar Atores e Casos de Uso;
- Identificar e descrever Atores;
- Identificar e descrever Casos de Uso;
- Elencar e priorizar todos os Casos de Uso.

6. IDENTIFICANDO ATORES

Para identificar os Atores do sistema basta fazer a seguinte pergunta: **"Quem está fazendo a atual interação?".** (Lembrando que o Ator é quem interage diretamente com o sistema).

Identificando Atores:

- Quem/O quê utiliza o sistema?
- Quem/O quê obtém informações deste sistema?
- Quem/O quê fornece informações para o sistema?
- Em que lugar da empresa o sistema será utilizado?
- Quem/O quê fornece suporte e manutenção ao sistema?
- Quais outros sistemas utilizam este sistema?

Atores representam diferentes tipos de usuário. Um Ator pode fornecer informação, receber informação ou fornecer e receber informação.

O Ator deve sempre ter um nome, uma descrição e relacionamentos com os Casos de Uso.

Ator é um papel, não uma pessoa ou coisa em particular. O nome do Ator deve representar da maneira mais clara possível o papel que ele exerce. Por exemplo, se determinado Caso de Uso é inicializado em um determinado tempo, como mensalmente ou no final do dia, a modelagem terá um caso especial de Ator: *scheduler* ou *time*.

7. IDENTIFICANDO CASOS DE USO

Os Casos de Uso descrevem o que o Ator quer que o sistema faça para fornecer algum valor para o Ator. A melhor forma para identificar Casos de Uso é considerar o quê cada Ator requisita do sistema.

Identificando Casos de Uso:

- Quais são os objetivos de cada Ator?
- Por que o Ator quer utilizar o sistema?
- Este Ator irá criar, armazenar, mudar ou remover dados deste sistema? Em caso afirmativo, por quê?
- O Ator necessitará informar ao sistema eventos externos ou mudanças?
- O Ator precisa ser informado sobre certas ocorrências?
- O sistema suporta todo o negócio com seus comportamentos corretos?

Ao fazer a primeira lista dos Casos de Uso, verifique se todas as funcionalidades requisitadas estão sendo tratadas, atente para não esquecer de mapear nenhuma funcionalidade. Cada Caso de Uso deve ter uma breve descrição que explica seu propósito.

8. RELACIONAMENTOS

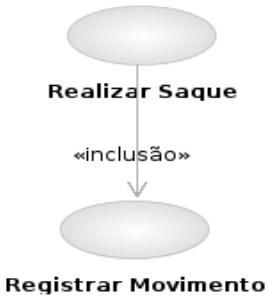
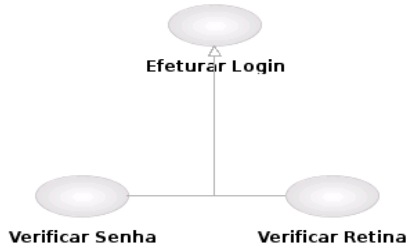
Inclusão Relacionamento que obriga a execução de determinado Caso de Uso antes da realização de outro, ou seja, dependência entre Casos de Uso. **Ex**: Sempre que um saque ocorrer, obrigatoriamente o histórico deve ser mantido.	
Extensão Relacionamento opcional de um Caso de Uso com outro, não há relação de dependência. **Ex**: Ao realizar um cadastro, o usuário pode ou não visualizar o produto.	
Generalização Relacionamento usado quando há um Caso de Uso com características comuns no comportamento, na estrutura, e na finalidade para dois ou mais Casos de Uso.	

Tabela 1: Relacionamentos.

8.1 Generalização de Ator

É possível que Atores possuam características comuns, Atores múltiplos podem ter os papéis ou finalidades comuns que interagem com um Caso do Uso.

Um usuário pode acumular diversos papéis em relação ao sistema, um usuário pode corresponder a diversos Atores.

Para fazer o modelo das situações citadas nos parágrafos anteriores é necessário representar relacionamentos entre Atores. O papel compartilhado é modelado como um Ator pai, conforme a Figura 3.

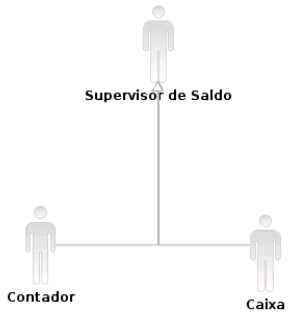


Figura 3: Generalização de Atores

8.2 Por que estruturar o modelo de Caso do Uso?

- Para simplificar os Casos de Uso originais = Mais fácil de compreender + Mais fácil de manter;
- Reuso de Casos de Uso.

8.3 Por que usar um relacionamento <<include>>?

- Caso de Uso comum e obrigatório para dois ou mais Casos de Uso;
- Para simplificar o fluxo de eventos em um Caso de Uso.

8.4 Porque usar um relacionamento <<extend>>?

- Para executar somente Casos de Uso sob determinadas circunstâncias (opcional);
- Para simplificar o fluxo de eventos em um Caso de Uso.

8.5 Porque usar um relacionamento de generalização?

- Para definir um comportamento comum entre dois ou mais casos do uso.

8.6 Porque usar um relacionamento de generalização de Ator?

- Para simplificar associações entre muitos Atores e Casos de Uso;
- Para mostrar que os Casos de Uso filhos podem executar todo o comportamento do Ator pai.

9. DECOMPOSIÇÃO FUNCIONAL

Todos têm a tendência de dividir um grande problema em partes menores para obter maior controle e encontrar uma solução completa para o problema original. Este tipo de situação, no caso de sistemas, pode ocasionar perda de contexto nas partes menores. Quanto maior o número de partes pequenas, maior é o número de interfaces necessário para solucionar cada uma das partes.

Os requisitos devem ser levantados em alto nível e abordados em um contexto que facilite o entendimento dos envolvidos.

9.1 Quando evitar a Decomposição Funcional

Sintomas:

- Casos de Uso muito pequenos;
- Número grande de Casos de Uso;
- Nome com operações de baixo nível (por ex.: Inserir Cartão – CRUD);
- Dificuldade para entender o modelo como um todo.

Ações Corretivas:

- Procurar por um contexto maior:

- Por que este sistema está sendo construído?

- Colocar-se no lugar do usuário:
  - Quais os objetivos do usuário?
  - Qual objetivo este Caso de Uso satisfaz?
  - Quais valores este Caso de Uso adiciona?
  - Qual a história por trás deste Caso de Uso?

10. CHECKLIST

10.1 Para Casos de Uso

- O modelo de Casos de Uso apresenta claramente o comportamento do sistema?
- É fácil compreender o propósito do sistema apenas revisando o modelo de Casos de Uso?
- Todos os Casos de Uso foram identificados? Todos endereçam a um comportamento requisitado?
- Todos os requisitos funcionais estão mapeados em pelo menos um Caso de Uso?
- O modelo de Casos de Uso contém algum comportamento supérfluo? Todos os Casos de Uso tem uma justificativa para existir? São necessários?
- Os Casos de Uso têm nomes únicos e intuitivos? Se não tiverem, convém alterar o nome para evitar que sejam esquecidos ou confundidos no futuro.
- Os clientes e usuários compreendem os nomes e descrições de cada Caso de Uso (direta ou indiretamente)?

- O atributo “Breve descrição do Caso de Uso” permite entender o propósito do mesmo? Cumpre seu objetivo?

- Cada Caso de Uso está envolvido com pelo menos um Ator?

- Algum Caso de Uso tem comportamentos ou fluxo de eventos similares?

- Os requisitos descobertos tem uma fonte e justificativa válida?

- O requisito está adequado aos objetivos de negócio do projeto?

- O requisito está em conflito com alguma restrição do domínio, política da empresa, regulamento ou lei?

- Todos os envolvidos foram consultados?

- Toda a informação capturada no Documento de Visão foi avaliada e tratada como requisito?

10.2 Para Atores

- Todos os Atores foram encontrados? Todos os papéis foram identificados no ambiente do sistema?

- Cada Ator está envolvido com pelo menos um Caso de Uso?

- Existe algum Ator que execute papéis semelhantes no sistema? Em caso afirmativo, eles poderiam ser representados por um único Ator?

10.3 Para o Modelo:

- Todos os Casos de Uso necessários foram identificados?

- Algum Caso de Uso desnecessário está presente no projeto?

- Os Casos de Uso acontecem na sequencia adequada?

- O modelo de Casos de Uso permite uma boa compreensão?