Doc.

03.3

**Abordagem Y**

**I. Visão Geral Abordagem X**

Tabela I – Visão Geral Abordagem Y

|  |  |  |  |
| --- | --- | --- | --- |
| **Abordagem Y** | | | |
| **Item** | **Sim** | **Não** | **Observação** |
| **Baseada em UML?** | X |  |  |
| **Perfil?** | X |  |  |
| **Processo?** | X |  |  |
| **Estereótipos?** | X |  | Estereótipos específicos padrões para todos os modelos. |
| **Diretrizes?** | X |  | Diretrizes específicas para cada modelo. |

**II. Estereótipos e Diretrizes**

Nesta seção são apresentados os estereótipos para aplicação em casos de uso, existentes no perfil da abordagem Y por meio da Tabela II, em seguida são apresentados exemplos do uso destes, seguidos pelas diretrizes para cada tipo de modelo.

Tabela II – Estereótipos da Abordagem Y para Casos de Uso.

|  |  |  |
| --- | --- | --- |
| **Estereótipos Abordagem Y** | | |
| **Para Casos de Uso  (Aplicados também aos demais modelos da abordagem)** | | |
| **Estereótipo** | **Utilização** | **Exemplo** |
| **<<variationPoint>>** | Representa o local em que ocorre uma variabilidade. Um ponto de variação está sempre associado a uma ou mais variantes. | Figura 1. |
| **<<mandatory>>** | A variante estará obrigatoriamente presente na configuração de qualquer produto da linha de produto. | Figura 1. |
| **<<optional>>** | A variante pode ou não estar presente na configuração de um produto da linha de produto. Variantes opcionais também podem ou não estar associadas a um ponto de variação. | Figura 2. |
| **<<alternative\_OR>>** | Estão sempre associadas aos pontos de variação. Pelo menos uma das variantes deverá ser escolhida para resolver o ponto de variação, ou seja, para estar presente na configuração de um produto da linha de produto. | Figura 1. |
| **<<alternative\_XOR>>** | Estão sempre associadas aos pontos de variação. Somente uma das variantes deverá ser escolhida para resolver o ponto de variação. | - |
| **<<variability>>** | Indica uma variabilidade existente em um modelo UML. | Figura 1 e 2. |
| **<<requires>>** | Indica um relacionamento de dependência (em UML) entre variantes no qual a variante dependente (origem da dependência) só existirá em uma configuração se a variante relacionada (destino da dependência) existir. | - |
| **<<mutex>>** | Indica um relacionamento de dependência (em UML) entre variantes no qual a variante dependente (origem da dependência) só existirá em uma configuração se a variante relacionada (destino da dependência) obrigatoriamente não existir. São conhecidas como variantes mutuamente exclusivas. | - |

**II.1 Exemplos**

***Casos de Uso***

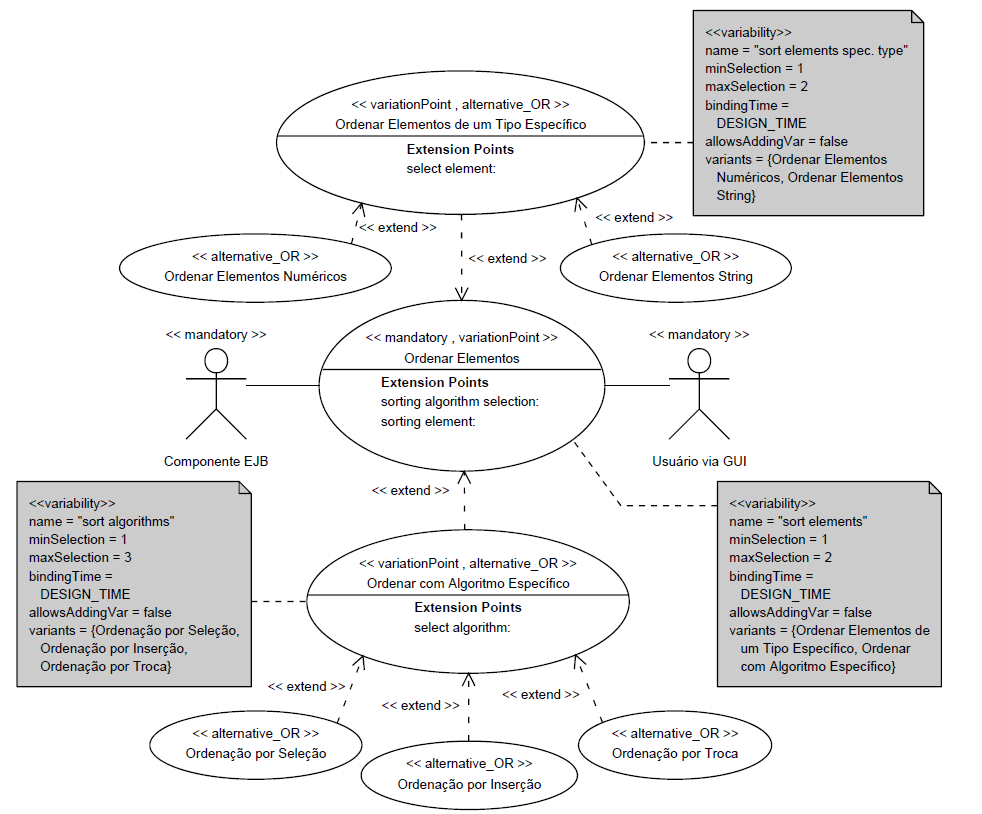


Figura 1 – Exemplo de Diagrama de Casos de Uso com a Abordagem Y.

Na Figura 1 observamos a aplicação da abordagem Y, e seus elementos. Passamos a analisar cada um deles, bem como as diretrizes presentes no processo da abordagem Y, que auxiliam sua utilização em outras LPs:

O caso de uso **Ordenar Elementos** é obrigatório <<*mandatory*>> e também é um ponto de variação <<*variationPoint*>>. Temos também dois casos de uso <<*alternative\_OR*>> que são variantes do ponto de variação **Ordenar Elementos**. O que é facilmente notado por meio do item ***variants*** da nota da variabilidade de nome “*sort elements*”.

As variabilidades são identificadas através de notas UML, estereotipada com <<*variability>>*. Nestas notas estão contidos os meta-atributos que seguem:

* ***Name***: nome da variabilidade;
* ***minSelection***: a quantidade mínima de variantes a serem selecionadas;
* ***maxSelection***: a quantidade máxima de variantes a serem selecionadas;
* ***bindingTime***: em qual momento será resolvida esta variabilidade ;
* ***allowsAddingVar***: se permite incluir novas variantes para resolver o ponto de variação; e
* ***variants***: quais as variantes para resolver o ponto de variação (casos de uso ligados ao ponto de variação).

Estas notas são inseridas em todas as variabilidades em caso de uso.

**Diretrizes para Identificação e Representação de Variabilidade** - Para a criação e classificação de variabilidades, as seguintes diretrizes presentes no processo da abordagem, sugerem ser seguidas:

VR.1 Variabilidades com variantes opcionais possuem multiplicidade ***minSelection*** ***= 0*** e ***maxSelection = 1NI***;

VR.2 Variabilidades com variantes exclusivas possuem multiplicidade ***minSelection = max-Selection = 1***;

VR.3 Variabilidades com variantes inclusivas possuem multiplicidade ***minSelection = 1*** e ***maxSelection = size(variants)*** em que ***size(x)*** é uma função que retorna a quantidade de elementos dacoleção ***x***;

VR.4 O valor ***bindingTime*** deve ser definido escolhendo-se um dos valores da classe de enumeração ***BindingTime***, que são: DESIGN\_TIME, LINK\_TIME, COMPLE\_TIME, RUNTIME;

VR.5 O valor *boobleano* do atributo ***allowsAddingVar*** deve ser analisado de acordo com a possibilidade de manter o ponto de variação aberto (*true*) ou fechado (*false*); e

VR.6 O valor da coleção ***variants*** é o conjunto formado pelas instâncias das variantes associadas ao ponto de variação ou variabilidade.

**Diretrizes para Casos de Uso -** As diretrizes especificadas para auxiliar na identificação das variabilidades em casos de uso são expressas abaixo:

UC.1 Elementos de modelos de casos de uso relacionados aos mecanismos de extensão e de pontos de extensão sugerem pontos de variação com variantes associadas, as quais podem ser inclusivas ou exclusiva;

UC.3 Em modelos de caso de uso relacionadas com a associação de inclusão (<<*include*>>) ou associados a atores sugerem variantes obrigatórias ou opcionais;

UC.5 Variantes que, ao serem selecionadas para fazer parte de um produto, exigem a presença de outra(s) determinada(s) variante(s) devem ter seus relacionamentos de dependência marcados com o estereótipo <<*requires*>>;

UC.6 Variantes mutuamente exclusivas para um determinado produto devem ter seus relacionamentos de dependência marcados com o estereótipo <<*mutex*>>.

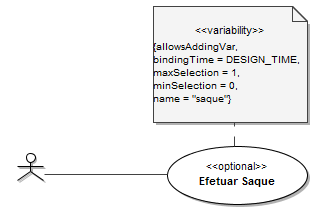


Figura 2 – Exemplo de Identificação de Variabilidade em Casos de Uso

Na Figura 2, que representa fragmento de um caso de uso, notamos que a variabilidade pode ser definida em um caso de uso único, como é o caso da variabilidade chamada *“saque”* identificada pelo comentário da UML estereotipado com <<*variability*>>.

***Relações de Extensão (Extend)***

Na Figura 3 podemos notar como a aplicação da abordagem é realizada para um caso de uso com associações de extensão (<<*extend>>*). Notamos que existe uma variante obrigatória, que não é considerada no meta-atributo ***minSelection*** e ***maxSelection***, por justamente ser necessária e obrigatoriamente estar presente na variabilidade.

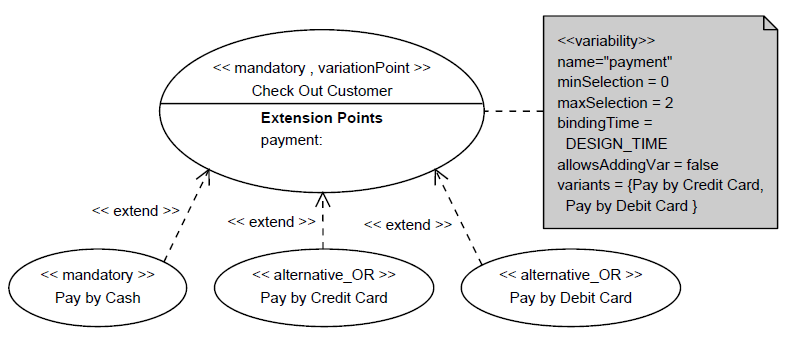


Figura 3 – Exemplo de Diagrama de Casos de Uso com Representação de Variabilidades para associações de extensão

***Relações de Inclusão (Include)***

Como indicado na diretriz UC.3 para a identificação de variabilidades: em modelos de caso de uso relacionadas com a associação de inclusão (<<*include*>>) ou associados a atores sugerem variantes obrigatórias ou opcionais.

**III. Exercício**

Dada a descrição da linha de produto de software pedagógica *Arcade Game Maker* (AGM), desenvolvida pelo *Software Engineering Institute* (SEI) apresente por meio de um diagrama de casos de uso, com a aplicação da **Abordagem Y** todas as variabilidades e similaridades descritas.

A LPS AGM é focada no desenvolvimento de jogos eletrônicos para celular, podendo gerar produtos com até três jogos diferentes: **Brickles, Pong** e/ou **Bowling**, mas pelo menos um deles deverá existir. Estes produtos podem apresentar configurações e elementos opcionais, como verificar a melhor pontuação obtida anteriormente (**Check Previous Best Score**), que necessitará obrigatoriamente, se selecionada, do caso de uso salvar a pontuação (**Save Score**). Todos os jogos devem possuir instalador (**Install Game**), também devem permitir salvar o jogo (**Save Game**), sair do mesmo (**Exit Game**), desinstalar (**Unistall Game**), e seja qualquer um dos três jogos, é necessária a inicialização (**initialization**) e a animação de loop (**Animation Loop**).