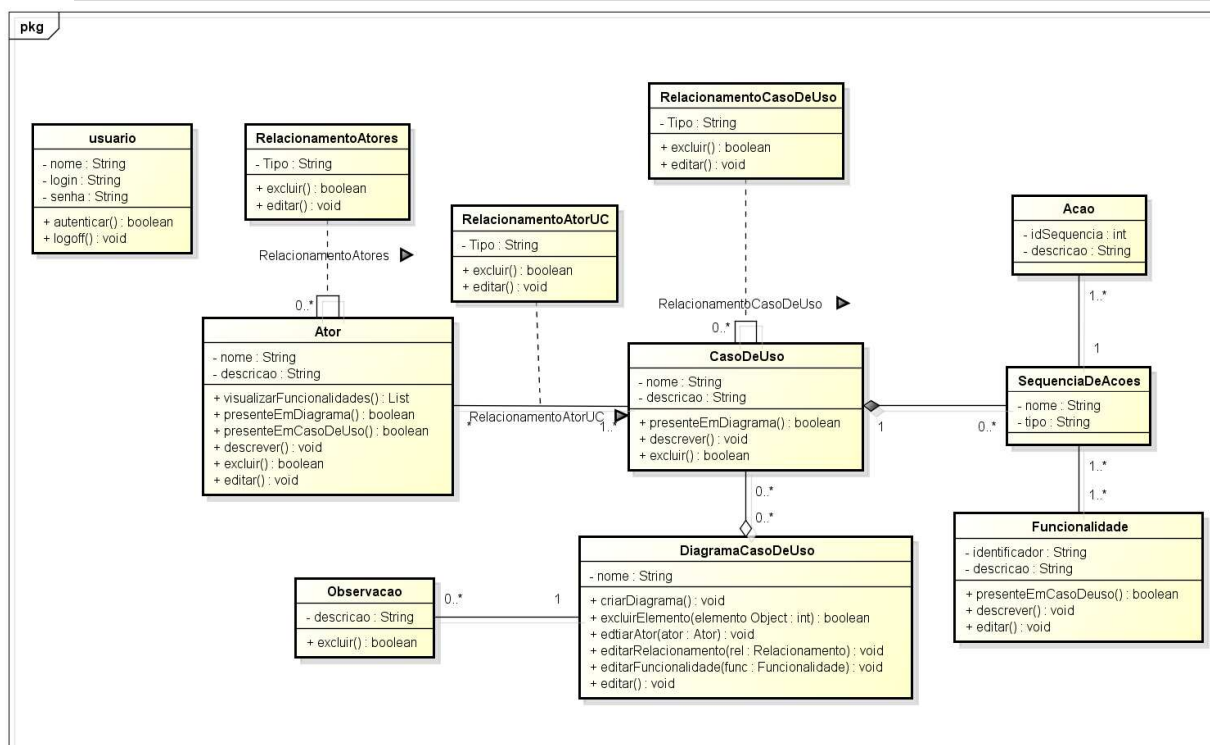


Atenção: Como a AD é individual, caso seja constatado que provas de alunos distintos são cópias uma das outras, independentemente de qualquer motivo, a todas será atribuída a nota ZERO. As soluções para as questões podem sim, ser buscadas por grupos de alunos, mas a redação final das respostas para as questões da prova tem que ser individual!

1. Utilizando a notação UML e considerando o documento de requisitos para a Ferramenta de Construção de Modelos de Casos de Uso(em anexo), apresente (7,0):

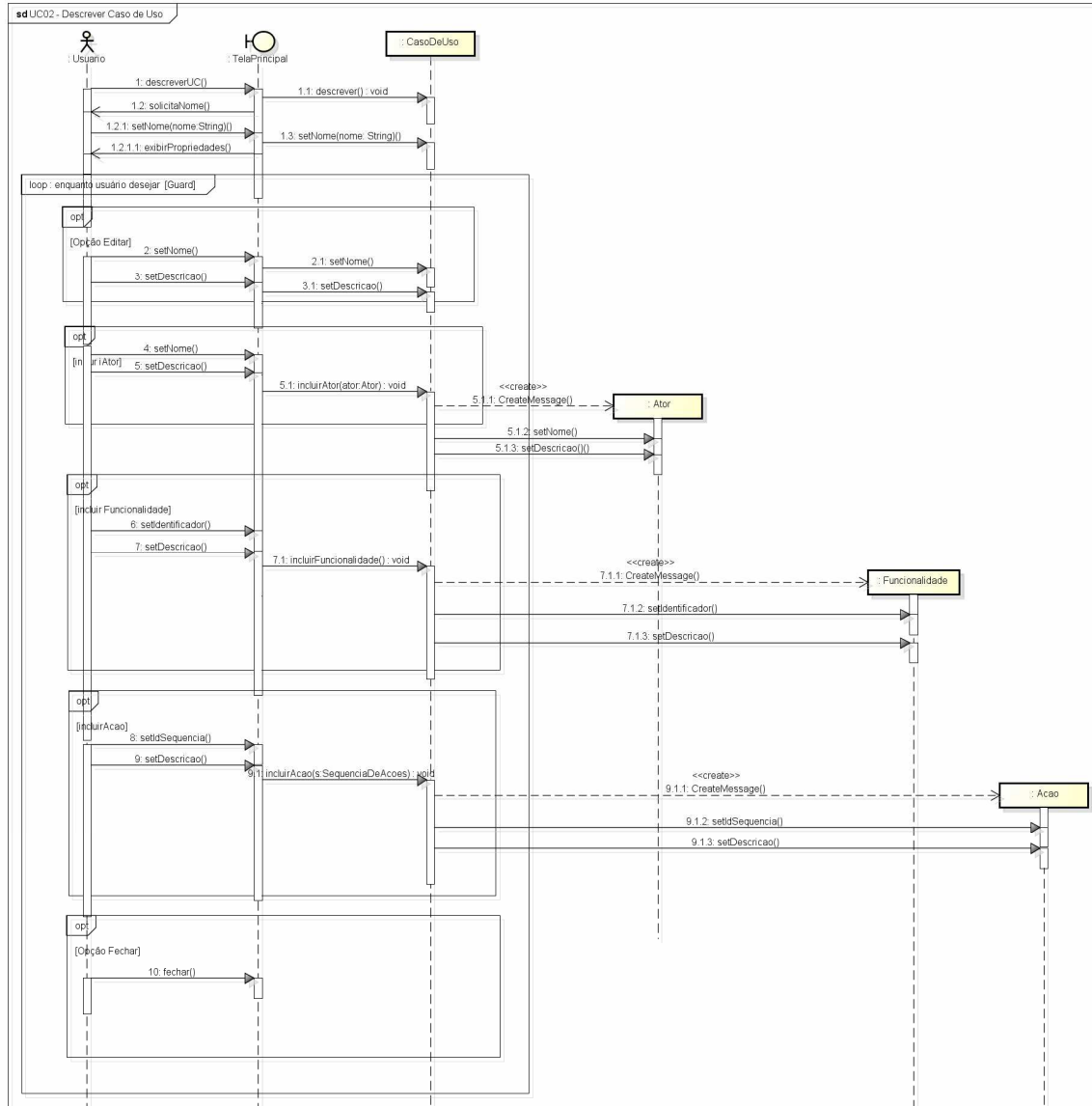
- a. o diagrama de classes para o sistema (2,0 pontos)

O diagrama de classes depende da perspectiva (ponto de vista) utilizada pelo engenheiro de software. Entretanto, alguns conceitos principais devem estar representados. O diagrama a seguir representa uma possível solução para um diagrama de classes para o sistema.



b. os diagramas de sequência para os casos de uso UC02, UC04 e UC10 (1,5 ponto)

Um exemplo de diagrama de sequência do UC02 é mostrado a seguir. É importante o aluno perceber que as classes e métodos utilizados no diagrama de sequência devem coincidir com o diagrama de classes (exceto os métodos get's e set's que geralmente são omitidos no diagrama de classes).



- c. Que técnicas de inspeção podem ser usadas para revisar estes diagramas? Que critérios você usou para indicar estas técnicas e que tipos de defeito elas podem ajudar a encontrar? Dê um exemplo aplicando uma das técnicas indicadas nos diagramas que você criou (1,0 ponto)

Poderiam ser utilizadas OORT's (Técnicas de Leitura Orientada a objetos). Pois elas apóiam a inspeção de projetos orientados a objetos. Essas técnicas podem revelar defeitos de omissão e ambiguidade, por exemplo.

Um exemplo de aplicação seria verificar se todos os métodos utilizados no diagrama de sequências estão presentes no diagrama de classes (*OORT's – Leitura Horizontal v3.0 – Leitura 1 – Diagramas de Sequencia x Classes*).

Outras técnicas de leitura como *ad-hoc* (leitura de acordo com a experiência do desenvolvedor) ou um *checklist* preparado para modelos de projeto também poderiam ser utilizadas.

- d. Apresente o conjunto de casos de testes para realizar o teste do UC04 – *Criar Diagrama de Caso de Uso*, indicando qual abordagem foi utilizada para gerar esta informação (1,5 pontos)

Apesar de não termos acesso ao código fonte do sistema em questão, nós temos acesso a descrição dos casos de uso. Por este motivo, podemos utilizar algum critério da técnica funcional.

Por exemplo, poderia ser definido um critério onde todos os fluxos do caso de uso fossem executados. Incluindo necessariamente as sequencias alternativas de ações (fluxos alternativos). A partir alguma abordagem específica (particionamento por equivalência, análise valor-limite ou grafos causa-efeito) poderia ser aplicada para identificar os casos de teste válidos e inválidos que deveriam ser usados para testar o UC.

- e. Utilize as métricas NOC, DIT e CBO para identificar as classes que poderiam ser mais propensas à falha no seu projeto. Indique as classes e o motivo desta indicação (1 ponto)

Esta questão depende do diagrama de classes construído.

Para o diagrama de classes apresentado como exemplo temos os seguintes valores para essas métricas:

Classe/Métrica	NOC	DIT	CBO
Usuario	0	0	0
Ator	0	0	3
CasoDeUso	0	0	5
SequenciaDeAcoes	0	0	3
Acao	0	0	1
Funcionalidade	0	0	1
DiagramaCasoDeUso	0	0	2
Observacao	0	0	1
RelacionamentoAtores	0	0	1
RelacionaemtnoAtorUC	0	0	1
RelacionamentoCasoDeUso	0	0	1

As métricas NOC e DIT ficaram com valor 0 para todas as classes, pois não temos herança em nosso diagrama de exemplo.

As classes CasoDeUso, SequenciaDeAcoes e Ator são aparentemente mais propensas à falha, pois apresentam o maior CBO.

Classes que possuem valor de Acoplamento alto em um projeto são usualmente mais propensas à falhas, pois possuem muitas relações de dependência com outras classes/objetos do projeto.

2. Quais são as três características fundamentais de um bom programa? Apresente e justifique a relação (ou ordem) de importância entre estas características. (valor: 1,0 ponto; máximo: 5 linhas).

Primeiro, um programa de estar correto, ou seja, deve fazer o que foi projetado para fazer e não fazer o contrário. Segundo, como modificar um sistema geralmente é mais caro do que construir, devemos desenvolver o sistema pensando em quem vai cuidar de sua manutenção no futuro. Assim, o programa deve ser claro. Finalmente, o programa deve ser eficiente do ponto de vista de memória e consumo de recursos computacionais.

3. Relacione o gerenciamento de tempo com o gerenciamento de escopo em projetos de desenvolvimento de software. Como as atividades de cada processo de gerenciamento geram insumos que são utilizados por outras atividades dos mesmos processos? Que insumos são estes e como são utilizados? (Valor: 2,0 pontos; máximo: 10 linhas)

O gerenciamento de escopo tem como objetivo identificar os trabalhos que devem ser realizados no contexto do projeto e aqueles que não devem ser realizados. O gerenciamento de escopo produz a WBS do projeto, que apresenta seus pacotes de trabalho. Estes pacotes deverão ser detalhados em atividades, que serão posteriormente sequenciadas e formarão o cronograma do projeto, já no contexto do gerenciamento de tempo.

Documento Especificação de Requisitos da Ferramenta de construção de Modelos de Casos de Uso.

Objetivo:

Construção de uma ferramenta capaz de auxiliar a construção de modelos de casos de uso de sistemas, capturando as necessidades observadas na aplicação da técnica de leitura PBR.

Glossário:

- *PBR: técnica de leitura baseada em perspectiva para inspeção de documentos de requisitos de software.*
- *Funcionalidades: conjunto de requisitos funcionais que caracterizam um sistema.*
- *Atores ou Participantes: são os usuários, dispositivos ou outros sistemas que interagem com o sistema.*
- *Caso de uso: um cenário de utilização de um sistema. A técnica PBR enfatiza a importância de identificar funcionalidades do sistema que estão sendo utilizadas e relacioná-las com atores que interagem com o sistema nesses cenários.*
- *Ação: é o modo pelo qual um agente (o sistema ou atores) atua sobre outro de modo a produzir um efeito. Uma ação pode ser de sistema ou de ator. Ações de atores são realizadas por atores. Ações de sistema são realizadas pelo sistema.*
- *Sequencia de Ações: são as sequencias (conjunto) de ações que caracterizam um caso de uso, fazendo com que o sistema comporte-se de determinada maneira. Sequencias de ações podem ser classificadas como naturais ou alternativas. Sequencia natural é a sequencia normal de ações que leva a execução de um caso de uso ao seu resultado esperado. Sequencias Alternativas são aquelas que manipulam um comportamento não esperado para a conclusão de sucesso de um caso de uso.*
- *Relacionamentos: expressam uma relação entre um ator e um caso de uso, ou entre casos de uso, ou entre atores. O único relacionamento entre atores é do tipo generalização. Relacionamentos entre atores e casos de*

uso caracterizam um determinado contexto de utilização do sistema por atores. Relacionamentos entre casos de uso são dos tipos generalizações, extensões, ou inclusões.

- *Diagrama de caso de uso: representação gráfica da interação entre os atores do sistema com os casos de uso e relacionamentos entre casos de uso.*
- *Observações: descrições textuais com algum tipo de informação presentes em determinados diagramas de caso de uso.*
- *Elementos do modelo de casos de uso: são diagramas de caso de uso, casos de uso, relacionamentos, atores, ações, sequência de ações e funcionalidades definidas durante a modelagem.*

Requisitos Funcionais:

Requisito Funcional 1

A Ferramenta deve permitir que o usuário descreva os atores identificados por meio da análise do documento de requisitos.

Requisito Funcional 2

A Ferramenta deve permitir que o usuário descreva as funcionalidades identificadas por meio da análise do documento de requisitos.

Requisito Funcional 3

A Ferramenta deve permitir que o usuário descreva os casos de uso identificados por meio da análise de documento de requisitos.

Requisito Funcional 4

A Ferramenta deve permitir que o usuário defina as sequências de ações que caracterizam um caso de uso.

Requisito Funcional 5

A Ferramenta deve permitir que o usuário determine quais participantes estão relacionados com um caso de uso do sistema.

Requisito Funcional 6

A Ferramenta deve permitir que o usuário determine quais funcionalidades estão associadas com um caso de uso do sistema.

Requisito Funcional 7

A Ferramenta deve permitir que o usuário visualize quais funcionalidades estão associadas com determinado participante do sistema.

Requisito Funcional 8

Todos os elementos do modelo (diagramas, casos de uso, atores, relacionamentos, ações, sequência de ações, funcionalidades e observações) devem poder ser descritos, editados ou excluídos pelo usuário.

Requisito Funcional 9

A ferramenta deve possuir mecanismos que permitam ao usuário criar somente os seguintes relacionamentos:

- *Inclusões, extensões ou generalizações entre casos de uso.*
- *Generalizações entre atores.*
- *Associações entre atores e casos de uso.*

Requisito Funcional 10

A ferramenta deve permitir que o usuário crie diagramas de casos de uso, utilizando elementos identificados durante a aplicação da PBR ou com novos elementos que podem ser descritos durante a confecção de diagramas.

Requisito Funcional 11

A ferramenta deve possuir mecanismos para verificação e identificação dos casos de uso, atores, ou relacionamentos descritos e que não aparecem em nenhum diagrama de casos de uso.

Requisito Funcional 12

A ferramenta deve possuir mecanismos para verificação e identificação de atores ou funcionalidades que não estão relacionados com nenhum caso de uso.

Requisitos de Dados:

Requisito de Dados 1

Um ator deve possuir um nome, um identificador, e pode possuir uma descrição.

Descrição de Dados:

```
Ator{  
  nome; (cadeia de no máximo 80 caracteres alfanuméricos)  
  descrição: (cadeia de no máximo 5000 caracteres  
             alfanuméricos,opcional)  
}
```

Requisito de Dados 2

Uma ação deve possuir um identificador de seqüência que determina a ordem em que as ações ocorrem em uma determinada seqüência de ações, e uma descrição.

Descrição de Dados:

```
Ação{  
  identificadorDeSequencia;(cadeia de no máximo 10 caracteres  
  alfanuméricos);  
  descrição; (cadeia de no máximo 5000 caracteres alfanuméricos)  
}
```

Requisito de Dados 3

Uma sequencia de ações é composta por ações, e cada sequencia deve ser distinguida como uma sequencia de ações alternativas ou não (natural). Se for uma sequencia alternativa deve possuir um nome.

Descrição de Dados:

```
SequênciaDeAcoes{
```


nome;(somente se for uma sequencia alternativa, cadeia de no máximo 80 caracteres alfanuméricos);
tipo;(assume somente os valores "alternativa" ou "natural")
Ações;(um conjunto de Ações);
}

Requisito de Dados 4

Um caso de uso deve possuir um nome e pode ser composto por uma sequencia natural de ações e um número qualquer de seqüências alternativas de ações.

Descrição de Dados:

Caso de Uso{

nome;(cadeia de no máximo 80 caracteres alfanuméricos)

Sequência Natural de Ações;(seqüência de ações que só pode ser a natural)

Descrição: (cadeia de no máximo 5000 caracteres alfanuméricos, opcional);

Sequências Alternativas;(conjunto de sequencias de ações que só podem ser alternativas)

}

Requisito de Dados 5

Um relacionamento é uma representação única entre um ator e um caso de uso, entre casos de uso ou entre atores. Um relacionamento entre casos de uso deve possuir um tipo, que define o estereótipo da relação entre tais entidades e este tipo pode assumir os valores "extensão", "inclusão", "generalização". Um relacionamento entre atores possui um tipo para definir o estereótipo da relação entre tais entidades com o valor "generalização". Relacionamentos entre atores e casos de uso possuem um tipo para definir o estereótipo da relação com o valor "associação".

Descrição de Dados:

Relacionamento{

primeiroIdentificador;(identificador de um caso de uso ou ator, cadeia de no máximo 100 caracteres alfanuméricos)

segundoIdentificador; (identificador de um caso de uso ou ator, cadeia de no máximo 100 caracteres alfanuméricos)

tipo;(Assume os valores "inclusão", "extensão" e "generalização" para relacionamento entre casos de uso. Assume o valor "generalização" para relacionamento entre atores)

}

Requisito de Dados 7

Uma funcionalidade deve possuir um identificador e uma descrição com sua definição.

Descrição de Dados:

*Funcionalidade{
identificador;(cadeia de no máximo 100 caracteres alfanuméricos);
descrição;(cadeia de no máximo 5000 caracteres alfanuméricos)
}*

Requisito de Dados 8

Um diagrama de casos de uso deve possuir um nome e é composto de casos de uso, atores, relacionamentos e observações.

Descrição de Dados:

*Diagrama de Casos de Uso{
nome;(cadeia de no máximo 80 caracteres alfanuméricos)
Casos de Uso;(conjunto de casos de uso)
Atores;(conjunto de atores)
Relacionamentos;(conjunto de relacionamentos)
Observações;(conjunto de observações)
}*

Requisito de Dados 9

Observações devem possuir uma descrição.

Descrição de Dados:

*Observação{
descrição;(cadeia de no máximo 5000 caracteres alfanuméricos)
}*

Restrições:

Restrição 1

Um ator deve estar relacionado a um ou mais casos de uso através de relacionamentos do tipo "associação".

Restrição 2

Um ator pode estar relacionado a qualquer número de atores, através de relacionamentos do tipo "generalização".

Restrição 3

Um caso de uso pode estar relacionado a qualquer número de casos de uso, através de relacionamentos dos tipos "generalização", "inclusão", "extensão".

Restrição 4

Um caso de uso pode estar relacionado a qualquer número de atores.

Restrição 5

Uma sequência de ações deve estar relacionada a um único caso de uso.

Restrição 6

Uma ação deve pertencer a uma única sequência de ações.

Restrição 7

Uma sequência de ações deve possuir pelo menos uma ação.

Restrição 8

Um diagrama de casos de uso pode possuir qualquer número de casos de uso, atores, relacionamentos e observações.

Restrição 9

Uma observação só pode estar presente em um único diagrama de casos de uso.

Restrição 10

Uma funcionalidade deve relacionar-se com pelo menos um caso de uso.

Restrição 11

Um caso de uso deve estar relacionado com pelo menos uma funcionalidade.

ANEXO II**Descrição de Casos de Uso da Ferramenta****UC01 - Caso de Uso: Descrever Ator**

Ator (es): Usuário

Início: O usuário quer descrever um ator.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Um novo ator adicionado ao modelo.

Funcionalidades envolvidas: 1, 8.

Seqüência Natural de Ações:

Ator	Sistema
1- Usuário requisita a definição de um ator.	
	2- Solicita um nome para o novo ator.
3- Usuário fornece um nome para o ator, a menos que: 3.a- O usuário desiste de descrever o ator.	
	4- Exibe propriedades do novo ator para serem editadas.
5- Usuário edita as propriedades desejadas, a menos que: 5.a- O usuário desiste de descrever o ator.	
	6- Adiciona um ator ao modelo.

Sequências Alternativas de Ações:**Sequencia Alternativa 3.a ou 5.a: O usuário desiste de descrever o ator.**

Ator	Sistema
	3.a.1 ou 5.a.1-Pergunta se deseja realmente desistir.
3.a.2 ou 5.a.2 -Usuário responde.	
	3.a.3 ou 5.a.3- Se resposta foi "sim" não adiciona ator ao modelo, senão CONTINUE (volta para 4).

UC02 - Caso de Uso: Descrever Caso de Uso

Ator (es): Usuário

Início: O usuário quer descrever um caso de uso.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Um novo caso de uso adicionado ao modelo.

Funcionalidades envolvidas: 3, 4, 6, 7, 8.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário requisita a definição de um caso de uso.	
	2- Solicita um nome para o novo caso de uso.
3- Usuário fornece um nome para o caso de uso, a menos que: 3.a- O usuário desiste de descrever o caso de uso.	
	4- Exibe propriedades do novo caso de uso para serem editadas. Disponibiliza a inclusão, exclusão e edição de ações. Disponibiliza inclusão e exclusão de atores e funcionalidades existentes no modelo associados ao novo caso de uso.
5- Enquanto (o usuário desejar): <ul style="list-style-type: none"> • Ele edita as propriedades desejadas. • Inclui, adiciona ou exclui atores e funcionalidades desejados associados ao caso de uso. • Adiciona, exclui ou edita ações. • A menos que: 5.a- O usuário desiste de descrever o caso de uso. Fim do Eqto;	
	6- Adiciona um caso de uso ao modelo.

Sequencias Alternativas de Ações:

Especificação de Requisitos

Página 13 de 24

Copyright WladimirChapetta/Guilherme Travassos – COPPE/UFRJ

Sequencia Alternativa 3.a ou 5.a: O usuário desiste de descrever o caso de uso.

Ator	Sistema
	3.a.1 ou 5.a.1-Pergunta se deseja realmente desistir.
3.a.2 ou 5.a.2 -Usuário responde.	
	3.a.3 ou 5.a.3- Se resposta foi “sim” não adiciona caso de uso ao modelo, senão CONTINUE (volta para 4).

UC03-Caso de Uso: Descrever Funcionalidade

Ator (es): Usuário

Início: O usuário quer descrever uma funcionalidade.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Uma nova funcionalidade adicionada ao modelo.

Funcionalidades envolvidas:

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário requisita a definição de uma funcionalidade.	
	2- Solicita um nome para a nova funcionalidade.
3- Usuário fornece um nome para a funcionalidade, a menos que: 3.a-O usuário desiste de descrever a funcionalidade.	
	4- Exibe propriedades da nova funcionalidade para serem editadas
5- Usuário edita as propriedades desejadas, a menos que: 5.a- O usuário desiste de descrever a funcionalidade.	
	6- Adiciona uma funcionalidade ao modelo.

Sequências Alternativas de Ações:

Sequência Alternativa 3.a ou 5.a: O usuário desiste de descrever a funcionalidade.

Ator	Sistema
	3.a.1 ou 5.a.1-Pergunta se deseja realmente desistir.
3.a.2 ou 5.a.2 -Usuário responde.	
	3.a.3 ou 5.a.3- Se resposta foi "sim" não adiciona funcionalidade ao modelo, senão CONTINUE (volta para 4).

UC04-Caso de Uso: Criar Diagrama de Caso de Uso

Ator (es): Usuário

Início: O usuário quer criar um novo diagrama de casos de uso.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Um novo diagrama de caso de uso é adicionado ao modelo.

Funcionalidades envolvidas: 8, 9, 10, 11, 12.

Sequência Natural de Ações:

Ator	Sistema
1- Usuário requisita a criação de um diagrama de casos de uso.	
	2- Solicita um nome para o novo diagrama.
3- Usuário fornece um nome para o diagrama, a menos que: 3.a-O usuário desiste de criar o diagrama.	
	4- Exibe o diagrama recém criado, para ser confeccionado.

<p>5- Enquanto o usuário desejar:</p> <ul style="list-style-type: none"> • Ele confecciona diagrama com os elementos já existentes do modelo. • A menos que: • 5.a- O usuário desiste de criar o diagrama. • 5.b- O usuário queira descrever um novo ator. • 5.c- O usuário queira descrever um novo caso de uso. • 5.d- O usuário queira descrever uma nova funcionalidade. • 5.e- O usuário queira editar um ator existente. • 5.f- O usuário queira editar um caso de uso existente. • 5.g- O usuário queira editar uma funcionalidade existente. • 5.h- O usuário queira criar um relacionamento. • 5.i- O usuário queira editar um relacionamento existente. <p>Fim do Eqto;</p>	
	6- Salva o diagrama resultante.

Sequências Alternativas de Ações:**Sequência Alternativa 3.a ou 5.a: O usuário desiste de descrever o ator.**

Ator	Sistema
	3.a.1 ou 5.a.1- Pergunta se deseja realmente desistir.
3.a.2 ou 5.a.2 -Usuário responde.	
	3.a.3 ou 5.a.3- Se resposta foi “sim” não adiciona o diagrama , senão CONTINUE (volta para 5).

Sequência alternativa 5.b: O usuário queira definir um novo ator

Ator	Sistema
	5.b.1- Inclui caso de uso “Descrever Ator”.
	5.b.2- CONTINUE(volta para 5).

Sequência alternativa 5.c: O usuário queira definir um novo caso de uso

Ator	Sistema
	5.c.1- Inclui caso de uso "Descrever Caso de Uso".
	5.c.2- CONTINUE(volta para 5).

Sequencia alternativa 5.d: O usuário queira definir uma nova funcionalidade

Ator	Sistema
	5.d.1- Inclui caso de uso "Descrever Funcionalidade".
	5.d.2- CONTINUE(volta para 5).

Sequencia alternativa 5.e: O usuário queira editar um ator existente

Ator	Sistema
	5.b.1- Inclui caso de uso "Editar Ator".
	5.b.2- CONTINUE(volta para 5).

Sequencia alternativa 5.f: O usuário queira editar um caso de uso existente

Ator	Sistema
	5.c.1- Inclui caso de uso "Editar Caso de Uso".
	5.c.2- CONTINUE(volta para 5).

Sequencia alternativa 5.g: O usuário queira editar uma funcionalidade existente

Ator	Sistema
	5.d.1- Inclui caso de uso "Descrever Funcionalidade".
	5.d.2- CONTINUE(volta para 5).

Sequencia alternativa 5.h: O usuário queira criar um relacionamento existente

Ator	Sistema
	5.c.1- Inclui caso de uso "Criar Relacionamento".
	5.c.2- CONTINUE(volta para 5).

Sequencia alternativa 5.i: O usuário queira editar um relacionamento existente

Ator	Sistema
	5.c.1- Inclui caso de uso "Editar Relacionamento".
	5.c.2- CONTINUE(volta para 5).

UC05-Caso de Uso: Excluir Elemento

Ator (es): Usuário

Especificação de Requisitos

Página 17 de 24

Copyright WladimirChapetta/Guilherme Travassos – COPPE/UFRJ

Início: O usuário quer excluir um elemento do modelo.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Um elemento é excluído do modelo.

Funcionalidades envolvidas: 8.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário seleciona o elemento a ser excluído.	
	2- Pede confirmação de exclusão do elemento.
3- Usuário confirma a exclusão, a menos: - O usuário desista de excluir o elemento.	
	4- Exclui o elemento.

Sequencias Alternativas de Ações:

1-Sequencia alternativa: O usuário desiste de excluir o elemento.

Ator	Sistema
	3.a-Pergunta se deseja realmente desistir.
3.b –Usuário responde.	
	3.c ou 5.c- Se resposta foi “sim” não exclui o elemento, senão CONTINUE (volta para 3).

UC06-Caso de Uso: Editar Ator

Ator (es): Usuário

Início: O usuário quer editar um ator do modelo.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Um ator do modelo tem suas propriedades alteradas.

Funcionalidades envolvidas: 4, 5, 6, 7, 8.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário seleciona o ator a ser editado.	
	2- Exibe propriedades do ator selecionado.

3- Usuário edita as propriedades desejadas.	
	4- Se alguma propriedade foi alterada, pergunta ao usuário se deseja salvar as alterações.
5- Usuário responde.	
	6- Se usuário responde “sim”, salvar as alterações realizadas.

UC07-Caso de Uso: Editar Relacionamento

Ator (es): Usuário

Início: O usuário quer editar um relacionamento do modelo.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo e o usuário deve estar trabalhando na confecção de um diagrama de caso de uso.

Pós-condições: Um relacionamento do modelo tem suas propriedades alteradas.

Funcionalidades envolvidas: 4, 5, 6, 7, 8.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário seleciona o relacionamento a ser editado.	
	2- Exibe propriedades do relacionamento selecionado.
3- Usuário edita as propriedades desejadas.	
	4- Se alguma propriedade foi alterada, pergunta ao usuário se deseja salvar as alterações.
5- Usuário responde.	
	6- Se usuário responde “sim”, salvar as alterações realizadas.

UC08-Caso de Uso: Editar Caso de Uso

Ator (es): Usuário

Início: O usuário quer editar um caso de uso do modelo.

Condições:

Especificação de Requisitos

Página 19 de 24

Copyright WladimirChapetta/Guilherme Travassos – COPPE/UFRJ

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Um caso de uso do modelo tem suas propriedades, suas associações ou ações alteradas.

Funcionalidades envolvidas: 4, 5, 6, 7, 8.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário seleciona do caso de uso a ser editado.	
	2- Exibe propriedades, as funcionalidades e atores associados, e a seqüência de ações do caso de uso selecionado.
3- Usuário edita as propriedades, as associações e ações desejadas.	
	4- Se alguma propriedade, associação ou ação foi alterada, pergunta ao usuário se deseja salvar as alterações.
5- Usuário responde.	
	6- Se usuário responde "sim", salvar as alterações realizadas.

UC09-Caso de Uso: Editar Funcionalidade

Ator (es): Usuário

Início: O usuário quer editar uma funcionalidade do modelo.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Uma funcionalidade do modelo tem suas propriedades alteradas.

Funcionalidades envolvidas: 4, 5, 6, 7, 8.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário seleciona a funcionalidade a ser editado.	
	2- Exibe propriedades da funcionalidade selecionada.
3- Usuário edita as propriedades desejadas.	

Especificação de Requisitos

Página 20 de 24

Copyright WladimirChapetta/Guilherme Travassos – COPPE/UFRJ

	4- Se alguma propriedade foi alterada, pergunta ao usuário se deseja salvar as alterações.
5- Usuário responde.	
	6- Se usuário responde "sim", salvar as alterações realizadas.

UC10-Caso de Uso: Editar Diagrama de Caso de Uso

Ator (es): Usuário

Início: O usuário quer editar um diagrama de casos de uso.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo.

Pós-condições: Um diagrama de caso de uso recentemente editado modelo.

Funcionalidades envolvidas: 8, 9, 10 , 11, 12.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário seleciona o diagrama de casos de uso a ser editado.	
	2- Exibe o diagrama selecionado.

<p>3- Enquanto o usuário desejar: <i>Ele edita propriedades e elementos contidos no diagrama.</i> <i>A menos que:</i> 3.a- O usuário desiste de editar o diagrama. 3.b- O usuário queira descrever um novo ator. 3.c- O usuário queira descrever um novo caso de uso. 3.d- O usuário queira descrever uma nova funcionalidade. 3.e- O usuário queira editar um ator existente. 3.f- O usuário queira editar um caso de uso existente. 3.g- O usuário queira editar uma funcionalidade existente. 3.h- O usuário queira criar um relacionamento. 3.i- O usuário queira editar um relacionamento existente.</p>	
	4- Se alguma propriedade ou elemento foi alterado, pergunta ao usuário se deseja salvar as alterações.
5- Usuário responde.	
	6- Se usuário responde "sim", salvar as alterações realizadas.

Sequências Alternativas de Ações:**Sequência Alternativa 3.a: O usuário desiste de editar o diagrama.**

Ator	Sistema
	3.a.1-Pergunta se deseja realmente desistir.
3.a.2-Usuário responde.	
	3.a.3- Se resposta foi "sim" não adiciona o diagrama , senão CONTINUE (volta para 5).

Sequência alternativa 3.b: O usuário queira definir um novo ator

Ator	Sistema
	5.b.1- Inclui caso de uso "Descrever Ator".

	5.b.2- CONTINUE(volta para 5).
--	---------------------------------

Sequencia alternativa 3.c: O usuário queira definir um novo caso de uso

Ator	Sistema
	5.c.1- Inclui caso de uso "Descrever Caso de Uso".
	5.c.2- CONTINUE(volta para 5).

Sequencia alternativa 3.d: O usuário queira definir uma nova funcionalidade

Ator	Sistema
	5.d.1- Inclui caso de uso "Descrever Funcionalidade".
	5.d.2- CONTINUE(volta para 5).

Sequencia alternativa 3.e: O usuário queira editar um ator existente

Invariantes: Nenhum

Ator	Sistema
	5.b.1- Inclui caso de uso "Editar Ator".
	5.b.2- CONTINUE(volta para 5).

Sequencia alternativa 3.f: O usuário queira editar um caso de uso existente

Ator	Sistema
	5.c.1- Inclui caso de uso "Editar Caso de Uso".
	5.c.2- CONTINUE(volta para 5).

Sequencia alternativa 3.g: O usuário queira editar uma funcionalidade existente

Invariantes: Nenhum

Ator	Sistema
	5.d.1- Inclui caso de uso "Descrever Funcionalidade".
	5.d.2- CONTINUE(volta para 5).

Sequencia alternativa 3.h: O usuário queira criar um relacionamento existente

Ator	Sistema
	5.c.1- Inclui caso de uso "Criar Relacionamento".
	5.c.2- CONTINUE(volta para 5).

Sequencia alternativa 3.i: O usuário queira editar um relacionamento existente

Ator	Sistema
	5.c.1- Inclui caso de uso "Editar Relacionamento".
	5.c.2- CONTINUE(volta para 5).

UC11 - Caso de Uso: Criar Relacionamento

Ator (es): Usuário

Início: O usuário quer criar um relacionamento entre um ator e um caso de uso, ou entre casos de uso.

Condições:

Invariantes: A ferramenta deve estar ativa.

Pré-condições: Para a ferramenta, um modelo de casos de uso deve estar ativo, e o usuário deve estar trabalhando na confecção de um diagrama de caso de uso.

Pós-condições: Um novo relacionamento adicionado ao modelo, entre dois casos de uso ou entre um caso de uso e um ator.

Funcionalidades envolvidas: 1, 8.

Sequencia Natural de Ações:

Ator	Sistema
1- Usuário requisita a criação de um relacionamento.	
2- Usuário escolhe os elementos (casos de uso ou ator): 2.a- O usuário desiste de descrever o relacionamento.	
	3- Adiciona um relacionamento ao modelo, e o exibe no diagrama.

Sequencias Alternativas de Ações:

Sequencia Alternativa 2.a: O usuário desiste de descrever o relacionamento.

Ator	Sistema
	2.a.1-Pergunta se deseja realmente desistir.
2.a.2 -Usuário responde.	
	2.a.3- Se resposta foi "sim" não adiciona ator ao modelo, senão CONTINUE (volta para 4).