

Capítulo

5. Estudo de Caso: Mobile Media

Este capítulo apresenta a utilização da abordagem **G2FM** (apresentada no capítulo anterior) com um sistema de Linha de Produtos de Software, o **Mobile Media**.

Neste capítulo apresentamos a utilização da abordagem **G2FM** (apresentada no Capítulo anterior) no contexto de uma linha de produtos de software denominado de **MobileMedia** [Figueiredo, E. et al., 2008]. O principal objetivo desse sistema é manipular fotos, música e vídeo em dispositivos móveis como, por exemplo, telefones celulares. A Figura 5.1 apresenta a tela da funcionalidade de adicionar foto. Para desenvolver o **MobileMedia**, os desenvolvedores estenderam a implementação base de uma LPS anterior chamada MobilePhoto [Young, T., 2005] incluindo novas *features* obrigatórias, opcionais e alternativas [Figueiredo, E. et al., 2008].



Figura 5.1 – Tela do sistema MobileMedia da funcionalidade de adicionar foto

Este estudo de caso tem o propósito de: (i) mostrar como a abordagem **G2FM** pode ser usada para extrair *features* de um modelo *i**; e (ii) avaliar se nossa proposta contribuirá para a escolha e justificativa da configuração de um produto específico.

Escolhemos esse sistema para realização do nosso estudo de caso por diversas razões: (i) é uma LPS que vem sendo utilizada em diversos estudos de caso [Procad, 2009]; (ii) é representativo para o domínio de dispositivos móveis, pois possui diversos pontos variáveis

relacionados a plataformas heterogêneas; e (iii) apresenta muitas *features* alternativas e opcionais [Figueiredo, E. et al., 2008].

5.1. Construção do modelo SR (opcional)

A primeira atividade do nosso processo é a construção do modelo SR do *framework* *i**, destacada na Figura 5.2. Vimos anteriormente que esta atividade é opcional, desde que já exista um modelo SR para o sistema a ser modelado. Como o sistema **MobileMedia** não possui este tipo de modelo, então faz-se necessária a execução desta atividade. O documento de casos de uso do sistema é o artefato consumido por esta atividade que, para nosso estudo de caso, é a *release* 1 do documento de casos de uso do **MobileMedia**.

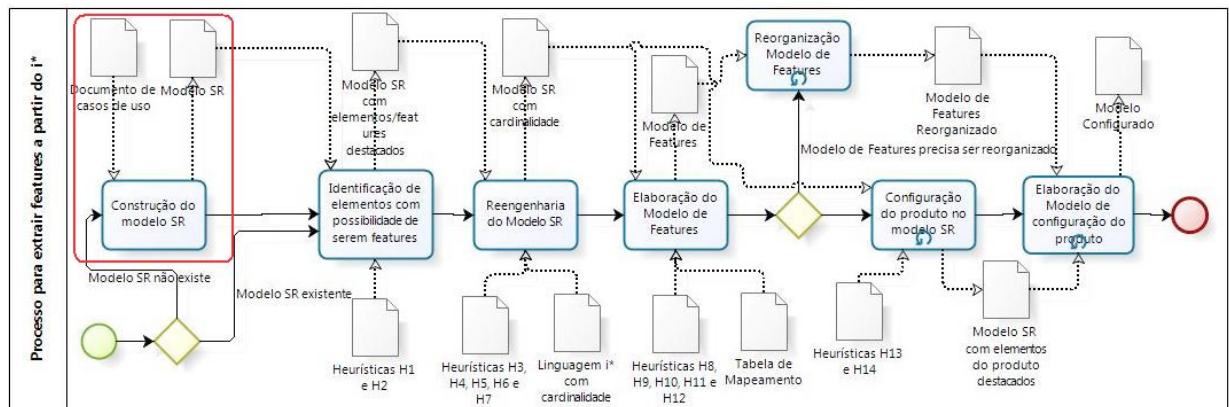


Figura 5.2 – Primeira atividade do processo G2FM

O documento de casos de uso do **MobileMedia** é composto por oito *releases*, disponibilizadas de forma incremental, ou seja, a cada nova *release* são adicionadas novas funcionalidades. Optamos por aplicar o nosso processo na *release* 1, que contem os seguintes casos de uso: **Adicionar Foto**, **Apagar Foto**, **Visualizar Conteúdo**, **Adicionar Álbum**, **Apagar Álbum** e **Fornecer Legenda**. Os passos definidos para esta atividade são aplicados a cada caso de uso.

5.1.1. Caso de Uso: Adicionar Foto

O modelo SR deste caso de uso já foi modelado no capítulo anterior, durante a apresentação da abordagem **G2FM**, visualizado na Figura 4.11.

5.1.2. Caso de Uso: Apagar Foto

A especificação original do caso de uso **Apagar Foto** encontra-se na Figura 5.3 abaixo.

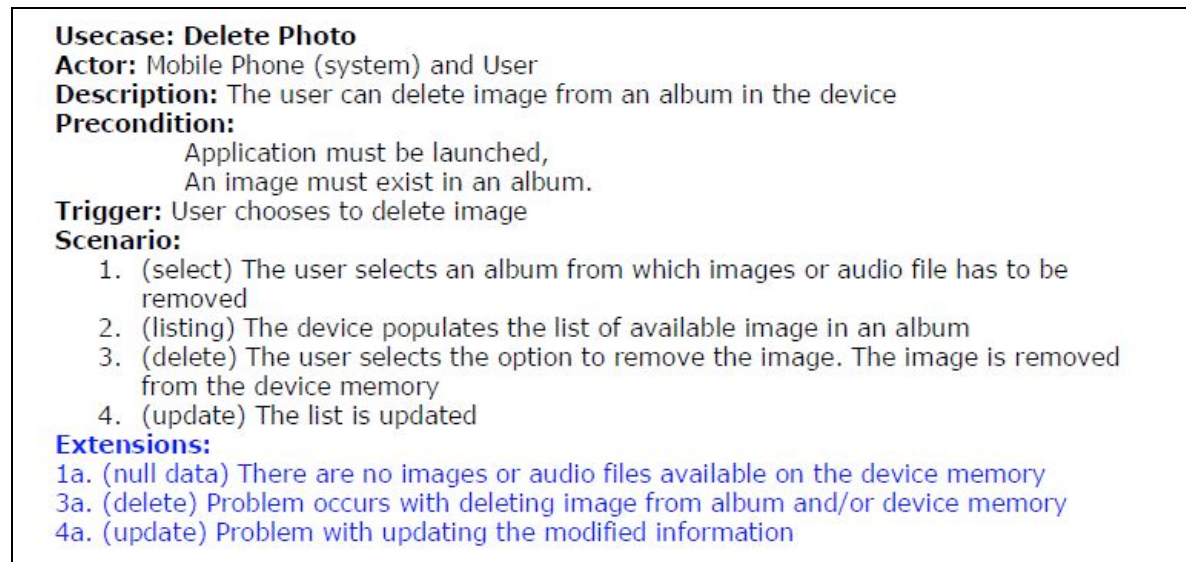


Figura 5.3 – Especificação original do caso de uso Apagar Foto

A seguir, encontram-se os passos da primeira atividade do processo, aplicados ao caso de uso Apagar Foto. O modelo SR do caso de uso **Apagar Foto** é apresentado na Figura 5.4.

Passo 1.1 - Identificação e modelagem dos atores e objetivos principais: de acordo com a especificação do caso de uso **Apagar Foto**, os atores identificados são **Telefone Móvel (sistema)** e **Usuário**, e seus principais objetivos são **Gerenciamento de Mídia** e **Manipulação de Mídia**, respectivamente.

Passo 1.2 – Construção do modelo i* operacional:

- **Regra 1 – Modelagem de atividades:** de acordo com esta regra, o caso de uso (ou atividade) **Apagar Foto** é modelado como uma tarefa associada ao objetivo principal dos atores **Telefone Móvel (sistema)** e **Usuário**, através do relacionamento meio-fim.
- **Regra 2 – Modelagem de tarefas:** os passos do cenário principal do caso de uso em questão são mapeados em sub-tarefas e relacionados à tarefa **Apagar Foto** através do relacionamento de decomposição de tarefas. As tarefas iniciadas no ator **Te-**

telefone Móvel (sistema) seriam: **Listar Fotos disponíveis**, **Remover Foto** e **Atualizar lista de fotos**, porém, da mesma forma que o caso de uso **Adicionar Foto**, não vemos sentido em modelar **Atualizar lista de fotos** como tarefa, pois, entendemos como uma pós-condição do caso de uso. Assim, o ator **Telefone Móvel (sistema)** tem as tarefas **Listar Fotos disponíveis** e **Remover Foto**. As tarefas iniciadas no ator **Usuário** são: **Selecionar Álbum**, **Selecionar opção de apagar fotos** e **Selecionar Foto a ser removida**. Esta última tarefa não aparece explicitamente nos passos do caso de uso, mas entendemos que ela é necessária para a realização do caso de uso, justificando a modelagem desta tarefa. Mais adiante, na Regra 5, com a modelagem das tarefas alternativas, a tarefa **Remover Foto** sofrerá modificação.

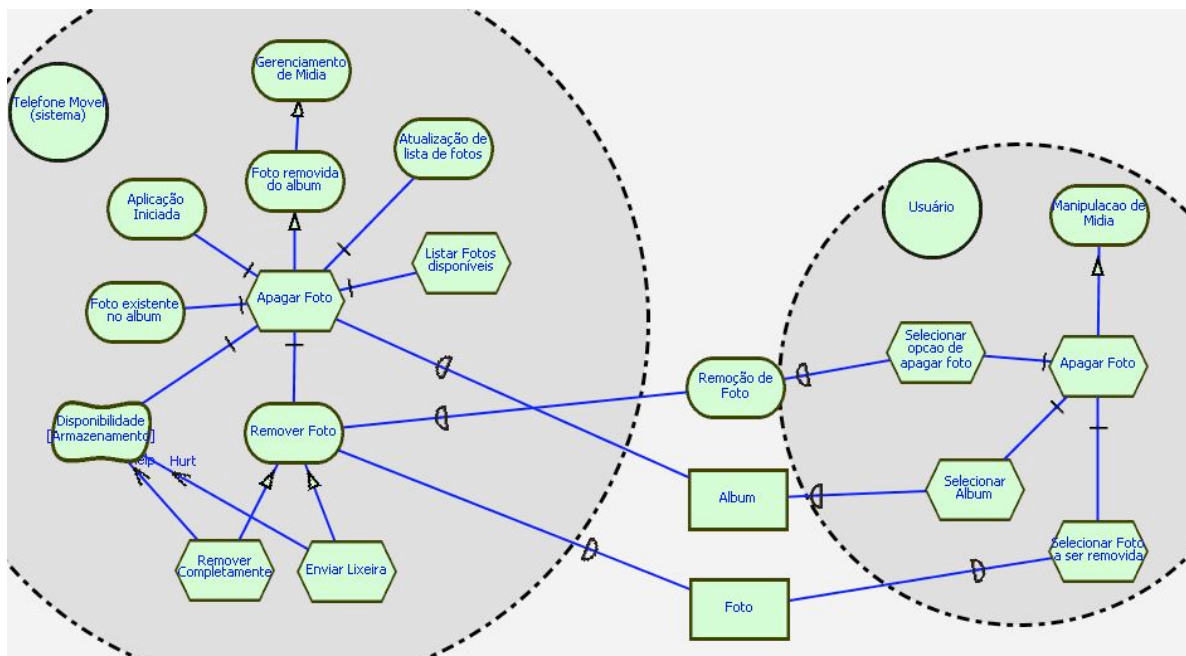


Figura 5.4 – Modelo SR do caso de uso Apagar Foto

- **Regra 3 – Modelagem de tarefas que produzem/fornecem recursos:** a tarefa **Selecionar foto a ser removida** (*dependee*) fornece o recurso **Foto** (*dependum*) que vai ser consumido pela tarefa **Remover Foto** (*dependor*).
- **Regra 4 – Modelagem de tarefas que consomem recursos:** a tarefa **Selecionar Álbum** (*dependor*) depende que o recurso **Álbum** (*dependum*) esteja disponível para que essa tarefa seja realizada. Como este recurso não foi produzido pela regra anterior, então, será produzido pela tarefa **Apagar Foto** (*dependee*).

- **Regra 5 – Modelagem de tarefas alternativas:** apesar da especificação original do caso de uso **Apagar Foto** (vide Figura 5.3) conter apenas cenários de exceção, sentimos a necessidade de adicionar um cenário alternativo ao passo 3 do cenário principal: “3. *(delete)* The user chooses to send the image to the trash.”. A modelagem deste cenário alternativo fica da seguinte forma: a tarefa **Remover Foto** torna-se um objetivo que pode ser alcançado pelas seguintes tarefas alternativas: **Remover Completamente** e **Enviar Lixeira**.
- **Regra 6 – Modelagem de pré-condições, pós-condições e eventos de *trigger*:** as pré-condições da tarefa **Apagar Foto** são **Aplicação Iniciada** e **Foto existente no álbum**. De acordo com esta regra, essas pré-condições são modeladas como objetivos e relacionados através do relacionamento de decomposição de tarefas, no ator que representa o sistema. A especificação original do caso de uso **Apagar Foto** não contém pós-condições, porém, como vimos anteriormente na Regra 2, supomos e assim, modelamos **Atualização de lista de fotos** como uma pós-condição do caso de uso. Da mesma forma que as pré-condições, essa pós-condição é modelada, no ator que representa o sistema, como um objetivo relacionado, através do relacionamento de decomposição de tarefas, com a tarefa que representa o caso de uso. O evento de *trigger* da tarefa **Apagar Foto** é **Remoção de Foto**. Esta *trigger* é um *dependum* do tipo objetivo, onde o *dependee* é a tarefa **Remover Foto** e o *dependor* é a tarefa **Selecionar opção de apagar foto**.
- **Regra 7 - Modelagem dos relacionamentos do tipo *includes* e *extends*:** esta regra não é aplicada neste caso de uso, pois, o mesmo não possui relacionamentos do tipo *includes* ou *extends*.

Passo 1.3 – Construção do modelo i* intencional:

- **Diretriz 1 – Análise dos elementos intencionais:**
 - Qual o estado final alcançado através da execução da tarefa “Apagar Foto”?
 - A resposta pode ser: “Foto removida do álbum selecionado”.
 - Que ator precisa atingir este objetivo (*dependor*)?
 - Resposta: o Telefone Móvel (sistema)
 - De que ator depende a obtenção do objetivo (*dependee*)?

Resposta: depende do usuário

Este objetivo (**Foto removida do álbum**) será inserido entre a tarefa e o objetivo principal do ator *depende*, ou seja, entre a tarefa **Apagar Foto** e o objetivo **Gerenciamento de Mídia** do ator **Telefone Móvel (sistema)**. Dessa forma, **Apagar Foto** se tornará o meio para obter o objetivo **Foto removida do álbum**, que por sua vez será um meio para alcançar o objetivo principal, **Gerenciamento de Mídia**.

- **Diretriz 2 – Análise de atributos de qualidade:** caso a foto seja removida completamente do celular, o sistema ganhará espaço de armazenamento. Assim, a tarefa **Remover Completamente** contribui positivamente para satisfazer o *softgoal* **Disponibilidade[Armazenamento]**, enquanto que a tarefa **Enviar Lixeira** contribui negativamente para a satisfação do mesmo *softgoal*.

5.1.3. Caso de Uso: Visualizar Conteúdo

A Figura 5.5 apresenta a especificação original do caso de uso **Visualizar Conteúdo**.

Usecase: View Content
Actor: Mobile Phone (system) and User
Description: The user can view a image or album content on the device memory
Precondition: Application must be launched A image must exist on an album
Trigger: User chooses to view an album or image in an album
Scenario: 1. (select)The user selects an album 2. (listing) The device populates the list of available images in an album 3. (view) User views the content of the album
Extensions: 1a. (null data) There are no images available on the device memory 3b. (view) View an image: 3a1. User selects an image from the list of images in an album. 3a2. Image is displayed on the screen 3c. (view) Problem occurs with retrieving 3c1. Problem in retrieving albums on the device 3c2. Problem in retrieving the images in an album

Figura 5.5 – Especificação original do caso de uso Visualizar Conteúdo

A seguir, encontram-se os passos da primeira atividade do processo, aplicados ao caso de uso Visualizar Conteúdo. A Figura 5.6 apresenta o modelo SR do caso de uso **Visualizar Conteúdo**.

Passo 1.1 - Identificação e modelagem dos atores e objetivos principais: de acordo com a especificação do caso de uso **Visualizar Conteúdo**, os atores identificados são **Telefone Móvel (sistema)** e **Usuário**, e seus principais objetivos são **Gerenciamento de Mídia** e **Manipulação de Mídia**, respectivamente.

Passo 1.2 – Construção do modelo i* operacional:

- **Regra 1 – Modelagem de atividades:** de acordo com esta regra, o caso de uso (ou atividade) **Visualizar Conteúdo** é modelado como tarefa e associada ao objetivo principal dos atores **Telefone Móvel (sistema)** e **Usuário**, através do relacionamento meio-fim.
- **Regra 2 – Modelagem de tarefas:** os passos do cenário principal do caso de uso em questão são mapeados em sub-tarefas e relacionados à tarefa **Visualizar Conteúdo** através do relacionamento de decomposição de tarefas. A tarefa iniciada no ator **Telefone Móvel (sistema)** é: **Listar Fotos disponíveis** e **Disponibilizar Foto**. Esta última tarefa não aparece explicitamente nos passos do caso de uso, mas foi modelada, pois achamos necessária para a realização do caso de uso. As tarefas iniciadas no ator **Usuário** são: **Selecionar Álbum** e **Visualizar Foto(s)**.
- **Regra 3 – Modelagem de tarefas que produzem/fornecem recursos:** a tarefa **Disponibilizar Foto** (*dependee*) fornece o recurso **Foto** (*dependum*) que vai ser consumida pela tarefa **Visualizar Foto(s)** (*dependor*). Mais adiante, na Regra 5, com a modelagem das tarefas alternativas, essa dependência sofrerá modificação.
- **Regra 4 – Modelagem de tarefas que consomem recursos:** a tarefa **Selecionar Álbum** (*dependor*) depende que o recurso **Álbum** (*dependum*) esteja disponível para que essa tarefa seja realizada. Como este recurso não foi produzido pela regra anterior, então, será produzido pela tarefa **Visualizar Conteúdo** (*dependee*).
- **Regra 5 – Modelagem de tarefas alternativas:** a especificação original do caso de uso **Visualizar Conteúdo** (vide Figura 5.5) contém o seguinte cenário alternativo: “3a1. User selects an image from the list of images in an album” e “3a2. Image is displayed on the screen”. A modelagem destes cenários alternativos fica da seguin-

te forma: a tarefa **Visualizar Foto(s)** do ator **Usuário** tem uma ligação de decomposição com o objetivo **Selecionar Foto**. Já a tarefa **Disponibilizar Foto** do ator **Telefone Móvel** tem uma ligação de decomposição com o objetivo **Foto disponibilizada na tela**. O primeiro objetivo (**Selecionar Foto**) tem as seguintes tarefas alternativas: **Selecionar pelo nome** e **Selecionar sequencialmente**. Nesse caso, verificamos que a dependência do recurso **Foto** passou a ser com o objetivo **Selecionar Foto**, ao invés de ser com a tarefa **Visualizar Foto(s)**. O objetivo **Foto disponibilizada na tela**, tem as tarefas **Disponibilizar tamanho original** e **Disponibilizar tamanho reduzido** como tarefas alternativas.

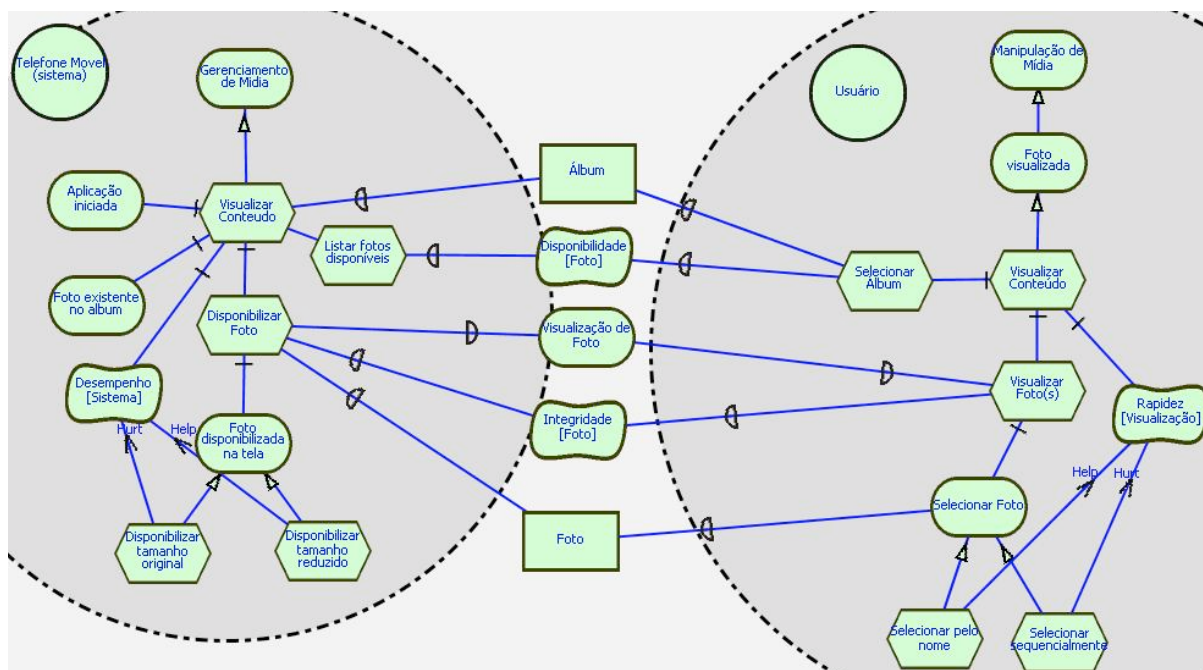


Figura 5.6 – Modelo SR do caso de uso Visualizar Conteúdo

- **Regra 6 – Modelagem de pré-condições, pós-condições e eventos de *trigger*:** as pré-condições da tarefa **Visualizar Conteúdo** são **Aplicação Iniciada** e **Foto existente no álbum**. De acordo com esta regra, essas pré-condições são modeladas como objetivos e relacionados através do relacionamento de decomposição de tarefas, no ator que representa o sistema. Este caso de uso não contém pós-condições. O evento de *trigger* deste caso de uso é **Visualização de Foto**. Esta *trigger* é um *dependum* do tipo objetivo onde o *dependee* é a tarefa **Visualizar Foto(s)** e o *depend* é a tarefa **Disponibilizar Foto**.

- **Regra 7 - Modelagem dos relacionamentos do tipo *includes* e *extends*:** esta regra não é aplicada neste caso de uso, pois, o mesmo não possui relacionamentos do tipo *includes* ou *extends*.

Passo 1.3 – Construção do modelo i* intencional:

- **Diretriz 1 – Análise dos elementos intencionais:**

- Qual o estado final alcançado através da execução da tarefa “Visualizar Conteúdo”?

A resposta pode ser: “Foto visualizada no álbum selecionado”.

- Qual ator precisa atingir este objetivo (*dependor*)?

Resposta: o usuário

- De que ator depende a obtenção do objetivo (*dependee*)?

Resposta: depende do Telefone Móvel (sistema)

Este objetivo (**Foto visualizada**) será inserido entre a tarefa e o objetivo principal do ator *dependor*, ou seja, entre a tarefa **Visualizar Conteúdo** e o objetivo **Manipulação de Mídia** do ator **Usuário**. Ou seja, **Visualizar Conteúdo** se tornará o meio para obter o objetivo **Foto visualizada**, que por sua vez será um meio para alcançar o objetivo principal, **Manipulação de Mídia**.

- **Diretriz 2 – Análise de atributos de qualidade:** para o sistema conseguir visualizar uma foto, o usuário depende que o Telefone Móvel (*dependee*) mantenha as fotos íntegras e disponíveis. Este desejo é representado pelos *softgoals* **Disponibilidade [Foto]** e **Integridade [Foto]**. Concluimos também que caso a foto seja visualizada sequencialmente, a visualização da mesma é mais lenta do que se for visualizada através do nome. Assim, a tarefa **Selecionar sequencialmente** contribui negativamente para satisfazer o *softgoal* **Rapidez [Visualização]**. Já a tarefa **Selecionar pelo nome** contribui positivamente para a satisfação do mesmo *softgoal*. O mesmo raciocínio é aplicado às tarefas alternativas **Disponibilizar tamanho original** e **Disponibilizar tamanho reduzido**. A primeira contribui negativamente para satisfazer o *softgoal* **Desempenho [Sistema]** e a segunda contribui positivamente para a satisfação do mesmo *softgoal*.

5.1.4. Caso de Uso: Adicionar Álbum

A especificação original do caso de uso **Adicionar Álbum** encontra-se na Figura 5.7. Comparando esta especificação com a do caso de uso **Adicionar Foto** (vide Figura 4.11), verificamos que são similares e assim, muito provavelmente, os modelos SR desses casos de uso também serão parecidos. O modelo SR do caso de uso **Adicionar Álbum** é apresentado na Figura 5.8

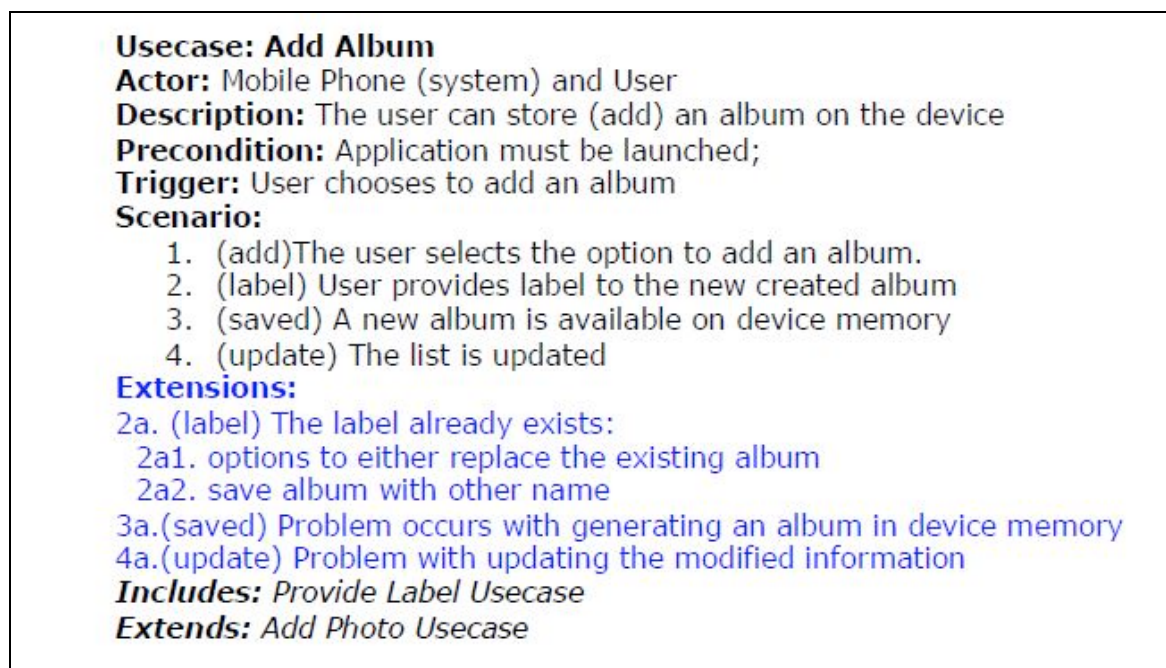


Figura 5.7 – Especificação original do caso de uso Adicionar Álbum

Passo 1.1 - Identificação e modelagem dos atores e objetivos principais: de acordo com a especificação do caso de uso **Adicionar Album**, os atores identificados são **Telefone Móvel (sistema)** e **Usuário**, e seus principais objetivos são **Gerenciamento de Mídia** e **Manipulação de Mídia**, respectivamente.

Passo 1.2 – Construção do modelo i* operacional:

- **Regra 1 – Modelagem de atividades:** de acordo com esta regra, o caso de uso **Adicionar Álbum** é modelado como tarefa e associada ao objetivo principal dos atores **Telefone Móvel (sistema)** e **Usuário**, através do relacionamento meio-fim.
- **Regra 2 – Modelagem de tarefas:** os passos do cenário principal do caso de uso em questão são mapeados em sub-tarefas e relacionados à tarefa **Adicionar Álbum**

através do relacionamento de decomposição de tarefas. As tarefas iniciada no ator **Telefone Móvel (sistema)** seriam: **Armazenar Álbum** e **Atualizar lista de álbuns**, porém, não vemos sentido em modelar **Atualizar lista de álbuns** como tarefa, pois, entendemos como uma pós-condição do caso de uso (vide Regra 6). Assim, o ator **Telefone Móvel (sistema)** tem apenas a tarefa **Armazenar Álbum**. As tarefas iniciadas no ator **Usuário** são: **Selecionar opção de adicionar álbum** e **Atribuir nome para álbum**.

- **Regra 3 – Modelagem de tarefas que produzem/fornecem recursos:** a tarefa **Atribuir nome para álbum** (*dependee*) fornece o recurso **Nome** (*dependum*) que vai ser consumido pela tarefa **Armazenar Álbum** (*dependor*).
- **Regra 4 – Modelagem de tarefas que consomem recursos:** a tarefa **Armazenar Álbum** consome o recurso **Nome**, que já foi modelado pela regra anterior.
- **Regra 5 – Modelagem de tarefas alternativas:** apesar da especificação original do caso de uso **Adicionar Álbum** (vide Figura 5.7) não conter cenários alternativos, mas apenas de exceção, sentimos a necessidade de acrescentar informações que achamos relevantes para enriquecer a especificação do caso de uso. Adicionamos cenário alternativo ao passo 3 do cenário principal: “3. (*saved*) The system saves automatically the new album on the device”. A modelagem deste cenário alternativo fica da seguinte forma: a tarefa **Armazenar Álbum** tem uma ligação de decomposição com o objetivo **Salvar Álbum** que tem as seguintes tarefas alternativas: **Salvar Automaticamente** e **Salvar pelo Usuário**.
- **Regra 6 – Modelagem de pré-condições, pós-condições e eventos de trigger:** a pré-condição da tarefa **Adicionar Álbum** é **Aplicação Iniciada**. De acordo com esta regra, essa pré-condição é modelada como objetivo e relacionado através do relacionamento de decomposição de tarefas, no ator que representa o sistema. A especificação original do caso de uso **Adicionar Álbum** não contém pós-condições, porém, como vimos anteriormente na Regra 2, supomos e assim, modelamos **Atualização de lista de álbuns** como uma pós-condição do caso de uso. Essa pós-condição é modelada da mesma forma que a pré-condição. O evento de *trigger* da tarefa **Adicionar Álbum** é **Adição de Álbum**. Esta *trigger* é um *dependum* do tipo objetivo onde o *dependee* é a tarefa **Armazenar Álbum** e o *dependor* é a tarefa **Selecionar opção de adicionar álbum**.

- **Regra 7 - Modelagem dos relacionamentos do tipo *includes* e *extends*:** o caso de uso **Adicionar Álbum** contém um relacionamento do tipo *includes* com o caso de uso **Fornecer Legenda** e um relacionamento do tipo *extends* com o caso de uso **Adicionar Foto**. De acordo com esta regra, **Fornecer Legenda** é modelada como sub-tarefa (através do relacionamento de decomposição de tarefas) da tarefa **Adicionar Álbum**. Nesse caso, verificamos que a dependência do recurso **Nome** passou a ser com a tarefa **Fornecer Legenda**, ao invés de ser com a tarefa **Armazenar Álbum**. **Adicionar Foto** é modelado como um objetivo relacionado ao objetivo **Gerenciamento de Mídia** através da ligação meio-fim. Mais adiante, na Diretriz 1, com a análise dos elementos intencionais, essa ligação sofrerá modificação.

Passo 1.3 – Construção do modelo i* intencional:

- **Diretriz 1 – Análise dos elementos intencionais:**
 - Qual o estado final alcançado através da execução da tarefa “Adicionar Álbum”?
A resposta pode ser: “Novo álbum incluído no celular”.
 - Qual ator precisa atingir este objetivo (*dependor*)?
Resposta: o Telefone Móvel (sistema)
 - De que ator depende a obtenção do objetivo (*dependee*)?
Resposta: depende do usuário

Este objetivo (**Novo álbum incluído**) será inserido entre a tarefa e o objetivo principal do ator *dependor*, ou seja, entre a tarefa **Adicionar Álbum** e o objetivo **Gerenciamento de Mídia** do ator **Telefone Móvel (sistema)**. Desse modo, **Adicionar Álbum** se tornará o meio para obter o objetivo **Novo álbum incluído**, que por sua vez será um meio para alcançar o objetivo principal, **Gerenciamento de Mídia**. Nesse caso, verificamos que o relacionamento do objetivo **Adicionar Foto** passou a ser com o objetivo **Novo álbum incluído**, ao invés de ser com o objetivo **Gerenciamento de Mídia**.

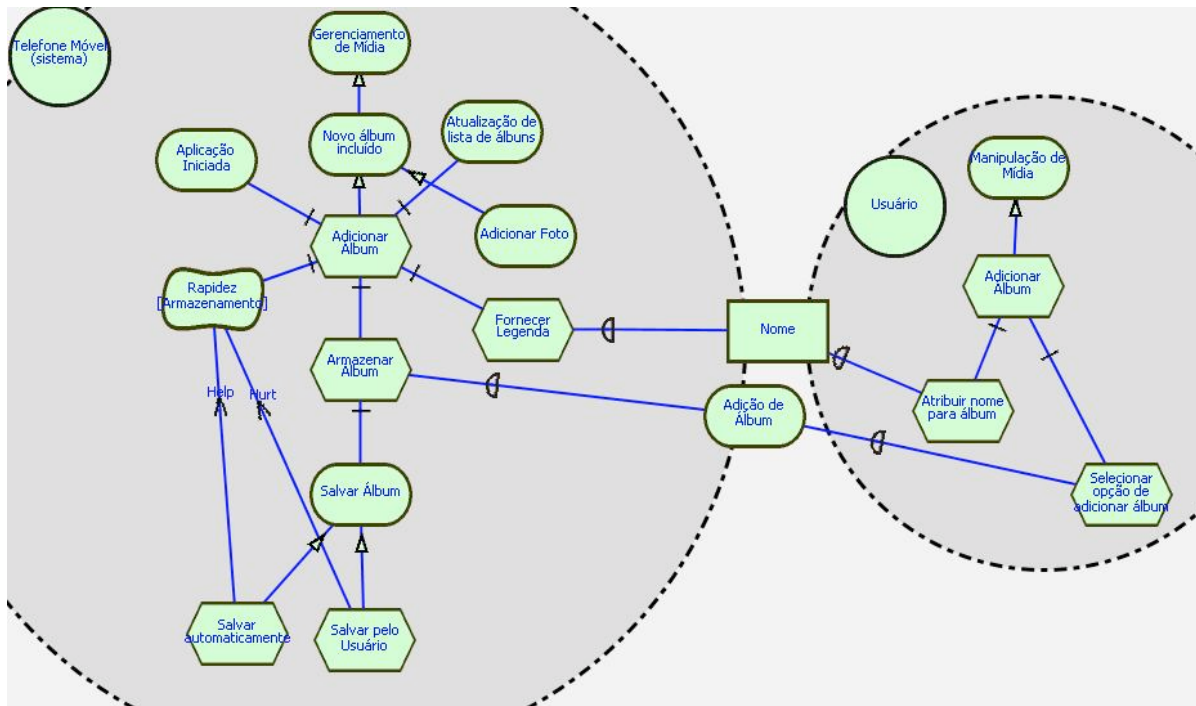


Figura 5.8 – Modelo SR do caso de uso Adicionar Álbum

- **Diretriz 2 – Análise de atributos de qualidade:** observamos que caso o álbum seja salvo pelo próprio sistema, ou seja, automaticamente, o armazenamento do mesmo é mais rápido do que se for salvo pelo usuário. Assim, a tarefa **Salvar Automaticamente** contribui positivamente para satisfazer o *softgoal* **Rapidez [Armazenamento]**. Já a tarefa **Salvar pelo Usuário** contribui negativamente para a satisfação do mesmo *softgoal*.

5.1.5. Caso de Uso: Apagar Álbum

Comparando a especificação original desse caso de uso (Figura 5.9) com a do caso de uso **Apagar Foto** (vide Figura 5.3), verificamos que são similares e assim, os modelos SR desses casos de uso, provavelmente, serão parecidos. A Figura 5.10 apresenta a configuração final do modelo SR do caso de uso **Apagar Álbum**.

Passo 1.1 - Identificação e modelagem dos atores e objetivos principais: de acordo com a especificação do caso de uso **Apagar Álbum**, os atores identificados são **Telefone Móvel (sistema)** e **Usuário**, e seus principais objetivos são **Gerenciamento de Mídia** e **Manipulação de Mídia**, respectivamente.

Passo 1.2 – Construção do modelo i* operacional:

- **Regra 1 – Modelagem de atividades:** de acordo com esta regra, o caso de uso **Apagar Álbum** é modelado como tarefa e associada ao objetivo principal dos atores **Telefone Móvel (sistema)** e **Usuário**, através do relacionamento meio-fim.

Usecase: Delete Album Actor: Mobile Phone (system) and User Description: The user can delete an album from the mobile phone Precondition: Application must be launched; The album must exist on the mobile phone, Trigger: User chooses to delete Album Scenario: 1. (listing) The albums stored in the mobile phone memory are listed down. 2. (delete) The user selects the option to delete the selected album 3. (delete) Album removed from the mobile phone 4. (update) The list is updated Extensions: 1a. (null data) There is no album available on the device 1b. (listing) Problem with listing down the albums 3a. (delete) Problem occurs while deleting the album 4a. (update) Problem with updating the modified information Extends: Delete Photo Usecase

Figura 5.9 – Especificação original do caso de uso Apagar Álbum

- **Regra 2 – Modelagem de tarefas:** os passos do cenário principal do caso de uso em questão são mapeados em sub-tarefas e relacionados à tarefa **Apagar Álbum** através do relacionamento de decomposição de tarefas. As tarefas iniciadas no ator **Telefone Móvel (sistema)** seriam: **Listar Álbuns existentes**, **Remover Álbum** e **Atualizar lista de álbuns**, porém, não vemos sentido em modelar **Atualizar lista de álbuns** como tarefa, pois, entendemos como uma pós-condição do caso de uso. Assim, o ator **Telefone Móvel (sistema)** tem as tarefas **Listar Álbuns existentes** e **Remover Álbum**. As tarefas iniciadas no ator **Usuário** são: **Selecionar opção de apagar álbum** e **Selecionar Álbum a ser removido**. Esta última tarefa não aparece explicitamente nos passos do caso de uso, mas foi modelada, pois achamos que ela é necessária para a realização do caso de uso. Mais adiante, na Regra 5, com a modelagem das tarefas alternativas, a tarefa **Remover Álbum** sofrerá modificação.

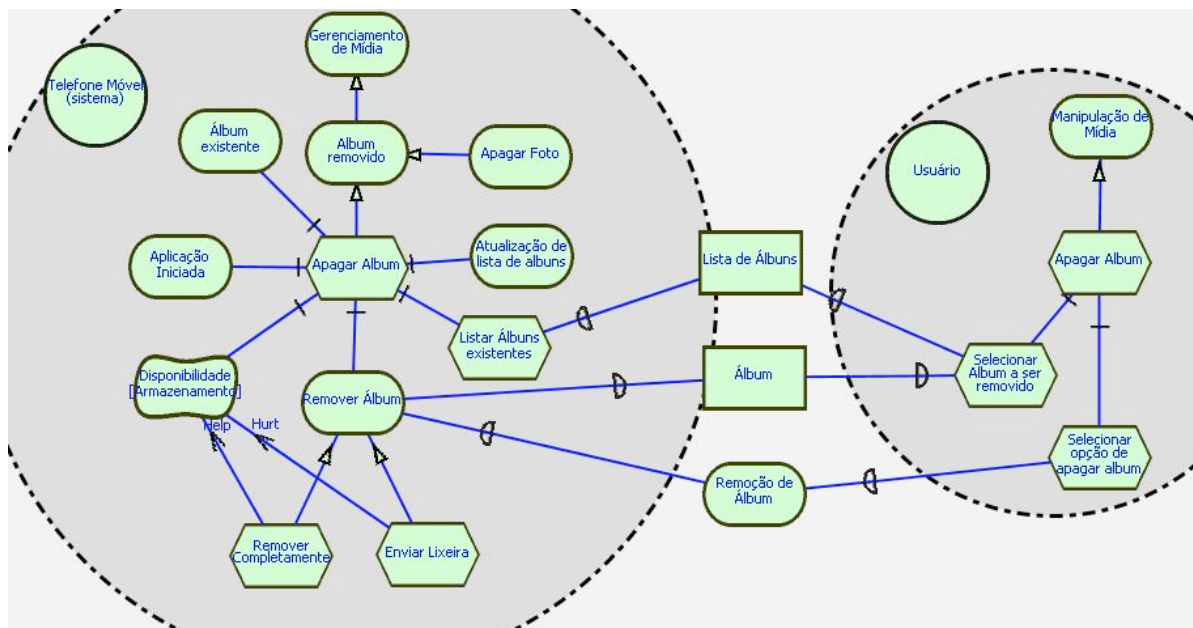


Figura 5.10 – Modelo SR do caso de uso Apagar Álbum

- **Regra 3 – Modelagem de tarefas que produzem/fornecem recursos:** a tarefa **Listar Álbuns existentes** (*dependee*) fornece o recurso **Lista de Álbuns** (*dependum*), que vai ser consumida pela tarefa **Selecionar Álbum a ser removido** (*dependor*). Esta tarefa (**Selecionar Álbum a ser removido**) por sua vez, fornece o recurso **Album** (*dependum*), que vai ser consumido pela tarefa **Remover Álbum** (*dependor*).
- **Regra 4 – Modelagem de tarefas que consomem recursos:** os recursos consumidos neste caso de uso (**Lista de Álbuns** e **Album**) foram produzidos pela regra anterior.
- **Regra 5 – Modelagem de tarefas alternativas:** apesar da especificação original do caso de uso **Apagar Álbum** (vide Figura 5.9) conter apenas cenários de exceção, sentimos a necessidade de adicionar cenários alternativos ao passo 3 do cenário principal: “3. (*delete*) The user chooses to send the album to the trash.”. A modelagem deste cenário alternativo fica da seguinte forma: a tarefa **Remover Álbum** torna-se um objetivo que pode ser alcançado pelas seguintes tarefas alternativas: **Remover Completamente** e **Enviar Lixeira**.
- **Regra 6 – Modelagem de pré-condições, pós-condições e eventos de *trigger*:** as pré-condições da tarefa **Apagar Álbum** são **Aplicação Iniciada** e **Álbum existente**. De acordo com esta regra, essas pré-condições são modeladas como objetivos e

relacionados através do relacionamento de decomposição de tarefas, no ator que representa o sistema. A especificação original do caso de uso **Apagar Álbum** não contém pós-condições, porém, como vimos anteriormente na Regra 2, supomos e assim, modelamos **Atualização de lista de álbuns** como uma pós-condição do caso de uso. Essa pós-condição é modelada da mesma forma que as pré-condições. O evento de *trigger* da tarefa **Apagar Álbum** é **Remoção de Álbum**. Esta *trigger* é um *dependum* do tipo objetivo, onde o *dependee* é a tarefa **Remover Álbum** e o *dependor* é a tarefa **Selecionar opção de apagar album**.

- **Regra 7 - Modelagem dos relacionamentos do tipo *includes* e *extends*:** o caso de uso **Apagar Álbum** contém um relacionamento do tipo *extends* com o caso de uso **Apagar Foto**. De acordo com esta regra, **Apagar Foto** é modelado como um objetivo relacionado ao objetivo **Gerenciamento de Mídia** através da ligação meio-fim. Mais adiante, na Diretriz 1, com a análise dos elementos intencionais, essa ligação sofrerá modificação.

Passo 1.3 – Construção do modelo i* intencional:

- **Diretriz 1 – Análise dos elementos intencionais:**
 - Qual o estado final alcançado através da execução da tarefa “Apagar Álbum”?
A resposta pode ser: “Álbum removido do celular”.
 - Qual ator precisa atingir este objetivo (*dependor*)?
Resposta: o Telefone Móvel (sistema)
 - De que ator depende a obtenção do objetivo (*dependee*)?
Resposta: depende do usuário

Este objetivo (**Álbum removido**) será inserido entre a tarefa e o objetivo principal do ator *dependor*, ou seja, entre a tarefa **Apagar Álbum** e o objetivo **Gerenciamento de Mídia** do ator **Telefone Móvel (sistema)**. Assim, **Apagar Álbum** se tornará o meio para obter o objetivo **Álbum removido**, que por sua vez será um meio para alcançar o objetivo principal, **Gerenciamento de Mídia**. Nesse caso, verificamos que o relacionamento do objetivo **Apagar Foto** passou a ser com o objetivo **Álbum removido**, ao invés de ser com o objetivo **Gerenciamento de Mídia**.

- **Diretriz 2 – Análise de atributos de qualidade:** caso o álbum seja removido completamente do celular, o sistema ganhará espaço de armazenamento. Assim, a tarefa **Remover Completamente** contribui positivamente para satisfazer o *softgoal* **Disponibilidade [Armazenamento]**, enquanto que a tarefa **Enviar Lixeira** contribui negativamente para a satisfação do mesmo *softgoal*.

5.1.6. Caso de Uso: Fornecer Legenda

A especificação original do caso de uso Fornecer Legenda encontra-se na Figura 5.11 e a Figura 5.12 apresenta o modelo SR deste caso de uso.

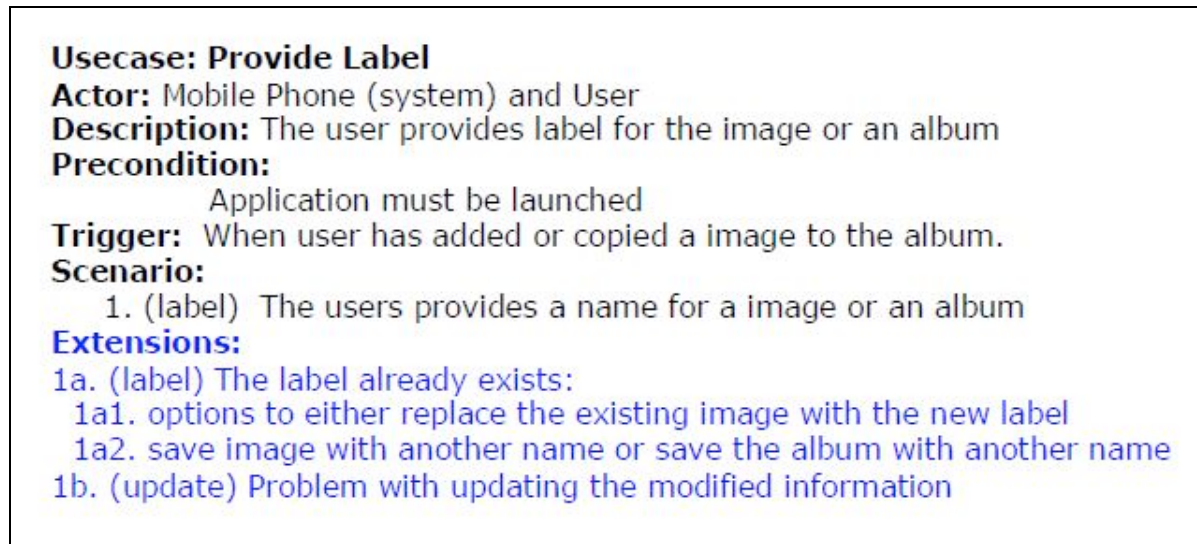


Figura 5.11 – Especificação original do caso de uso Fornecer Legenda

Passo 1.1 - Identificação e modelagem dos atores e objetivos principais: de acordo com a especificação do caso de uso **Fornecer Legenda**, os atores identificados são **Telefone Móvel (sistema)** e **Usuário**, e seus principais objetivos são **Gerenciamento de Mídia** e **Manipulação de Mídia**, respectivamente.

Passo 1.2 – Construção do modelo i* operacional:

- **Regra 1 – Modelagem de atividades:** de acordo com esta regra, o caso de uso **Fornecer Legenda** é modelado como tarefa e associada ao objetivo principal dos atores **Telefone Móvel (sistema)** e **Usuário**, através do relacionamento meio-fim.
- **Regra 2 – Modelagem de tarefas:** só existe um passo no cenário principal desse caso de uso que é mapeada na sub-tarefa **Atribuir nome**, no ator **Usuário**.

- **Regra 3 – Modelagem de tarefas que produzem/fornecem recursos:** a tarefa **Atribuir nome** (*dependee*) fornece o recurso **Nome** (*dependum*) que vai ser consumido pela tarefa **Fornecer Legenda** (*depende*). Mais adiante, na Regra 5, com a modelagem das tarefas alternativas, essa dependência sofrerá modificação.
- **Regra 4 – Modelagem de tarefas que consomem recursos:** o recurso consumido neste caso de uso (**Nome**) já foi modelado na regra anterior.
- **Regra 5 – Modelagem de tarefas alternativas:** apesar da especificação original do caso de uso **Fornecer Legenda** (vide Figura 5.11) não conter cenários alternativos, mas apenas de exceção, sentimos a necessidade de acrescentar informações que achamos relevantes para enriquecer a especificação do caso de uso. Assim, adicionamos um cenário alternativo ao passo 1 do cenário principal: “1. (*label*) *The system assigns a label automatically to the photo or album*”. A modelagem deste cenário alternativo fica da seguinte forma: a tarefa **Fornecer Legenda** tem uma ligação de decomposição com o objetivo **Nomear Álbum ou Foto** que por sua vez, tem as seguintes tarefas alternativas: **Nomear Automaticamente** e **Nomear pelo Usuário**. Nesse caso, verificamos que a dependência do recurso **Nome** passou a ser com a tarefa alternativa **Nomear pelo Usuário**, ao invés de ser com a tarefa **Fornecer Legenda**.
- **Regra 6 – Modelagem de pré-condições, pós-condições e eventos de *trigger*:** a pré-condição da tarefa **Fornecer Legenda** é **Aplicação Iniciada**. De acordo com esta regra, essa pré-condição é modelada, no ator que representa o sistema, como objetivo e relacionado através do relacionamento de decomposição de tarefas. O evento de *trigger* da tarefa **Fornecer Legenda** é **Adição de Foto/Adição de Álbum**. Esta *trigger* é um *dependum* do tipo objetivo onde o *dependee* é a tarefa **Fornecer Legenda** e o *depende* é a tarefa **Atribuir Nome**.
- **Regra 7 - Modelagem dos relacionamentos do tipo *includes* e *extends*:** esta regra não é aplicada neste caso de uso, pois, o mesmo não possui relacionamentos do tipo *includes* ou *extends*.

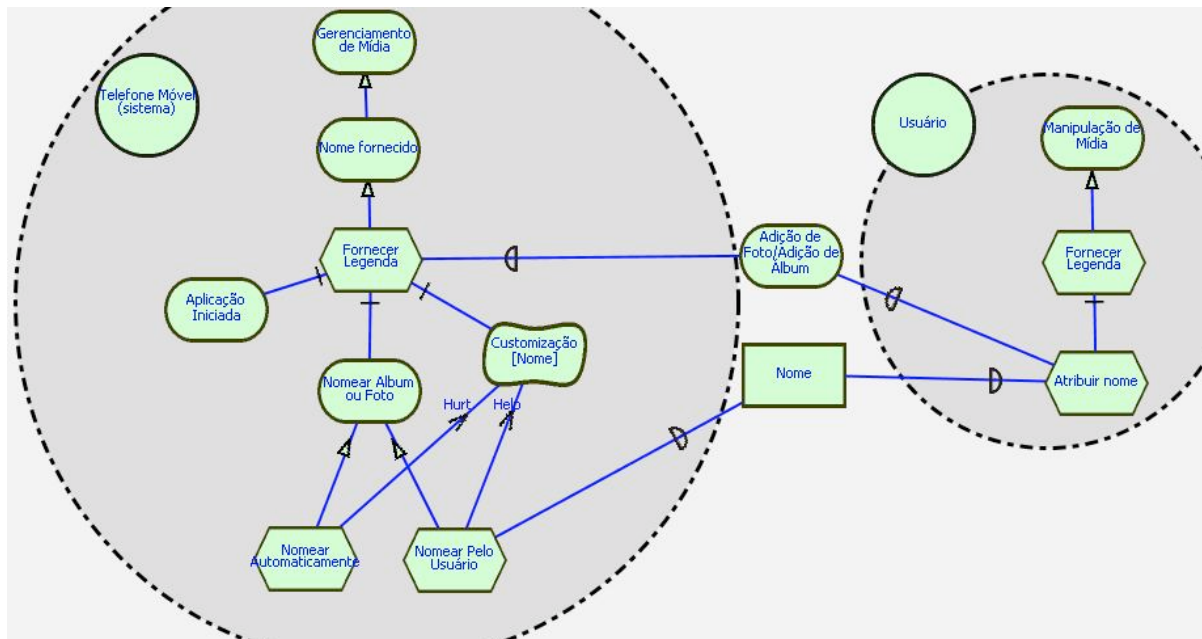


Figura 5.12 – Modelo SR do caso de uso Fornecer Legenda

Passo 1.3 – Construção do modelo i* intencional:

- **Diretriz 1 – Análise dos elementos intencionais:**

- Qual o estado final alcançado através da execução da tarefa “Fornecer Legenda”?

A resposta pode ser: “Nome fornecido a foto ou álbum”.

- Qual ator precisa atingir este objetivo (*dependor*)?

Resposta: o Telefone Móvel (sistema)

- De que ator depende a obtenção do objetivo (*dependee*)?

Resposta: depende do usuário

Este objetivo (**Nome fornecido**) será inserido entre a tarefa e o objetivo principal do ator *dependor*, ou seja, entre a tarefa **Fornecer Legenda** e o objetivo **Gerenciamento de Mídia** do ator **Telefone Móvel (sistema)**. Assim, **Fornecer Legenda** se tornará o meio para obter o objetivo **Nome fornecido**, que por sua vez será um meio para alcançar o objetivo principal, **Gerenciamento de Mídia**.

- **Diretriz 2 – Análise de atributos de qualidade:** observamos que caso a foto ou o álbum seja nomeado pelo usuário, o nome será customizado. Assim, a tarefa **Nomear Automaticamente** contribui negativamente para satisfazer o *softgoal* **Customi-**

zação [Nome] e a tarefa **Nomear pelo Usuário** contribui positivamente para a satisfação do mesmo *softgoal*.

5.2. Identificação de elementos com possibilidade de serem *features*

A segunda atividade do processo **G2FM** é a identificação dos possíveis elementos que poderão ser *features*. De acordo com a Figura 5.13, observamos que os artefatos consumidos nesta atividade são o modelo SR e a aplicação de algumas heurísticas.

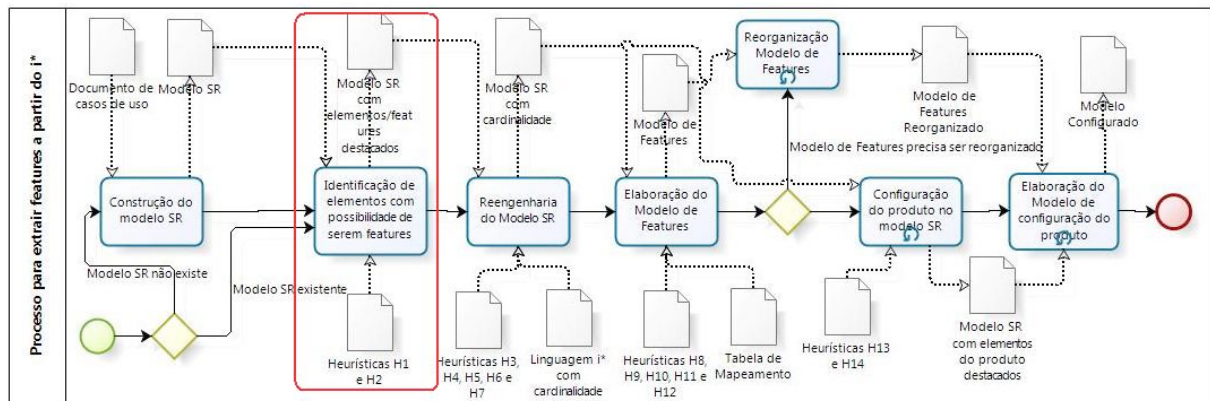


Figura 5.13 – Segunda atividade do processo G2FM

Por questões de legibilidade, aplicaremos as heurísticas aos modelos SR dos casos de uso separadamente, ao invés de unificar os modelos em um só. O resumo das heurísticas utilizadas nesta atividade encontra-se na Tabela 5.1.

Tabela 5.1 – Resumo das heurísticas H1 e H2

Heurística	Definição
H1	As <i>features</i> podem ser extraídas a partir dos elementos do tipo tarefa e recurso.
H2	As <i>features</i> podem ser extraídas a partir dos elementos internos do ator que representa o sistema e das dependências (particularmente do <i>dependum</i>) ligadas diretamente a esse ator.

Aplicando as heurísticas H1 e H2 aos modelos SR dos casos de uso **Apagar Foto** (Figura 5.4), **Visualizar Conteúdo** (Figura 5.6), **Adicionar Álbum** (Figura 5.8), **Apagar Álbum** (Figura 5.10) e **Fornecer Legenda** (Figura 5.12), obtemos as figuras 5.14, 5.15, 5.16, 5.17 e 5.18, respectivamente. Essas figuras obtidas representam a saída desta atividade do processo, ou seja, modelos SR com destaque nos elementos possíveis de serem *features*.

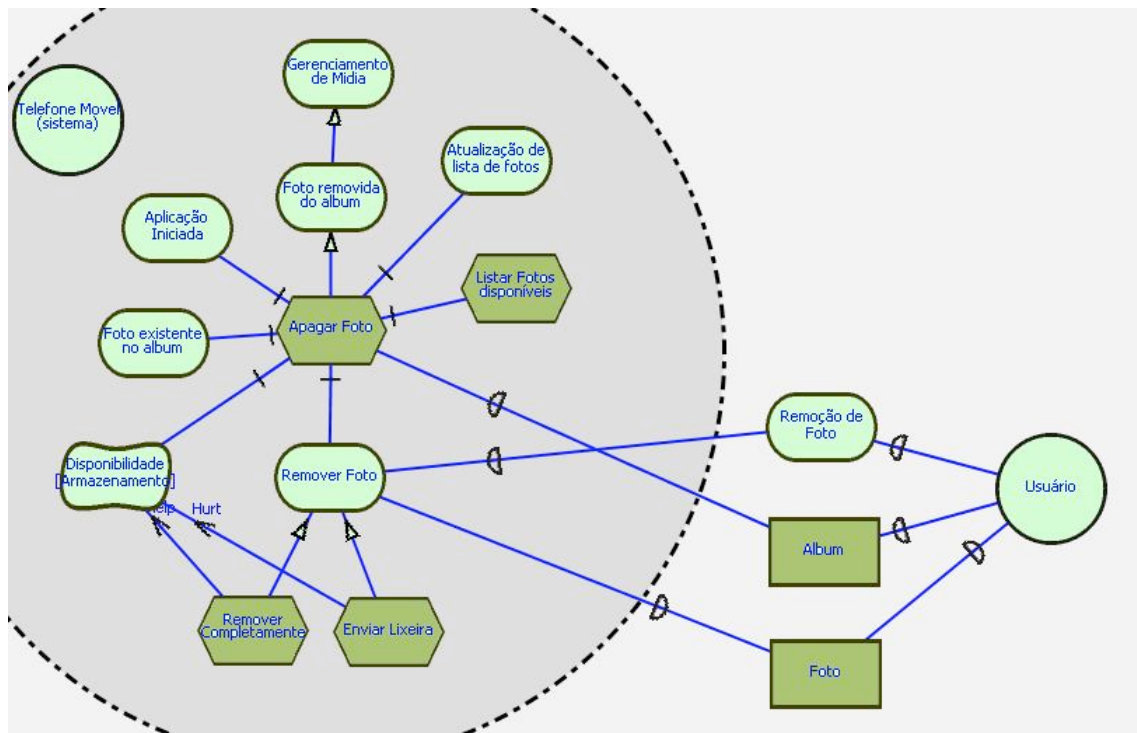


Figura 5.14 – Modelo SR do caso de uso Apagar Foto com possíveis elementos de serem *features*

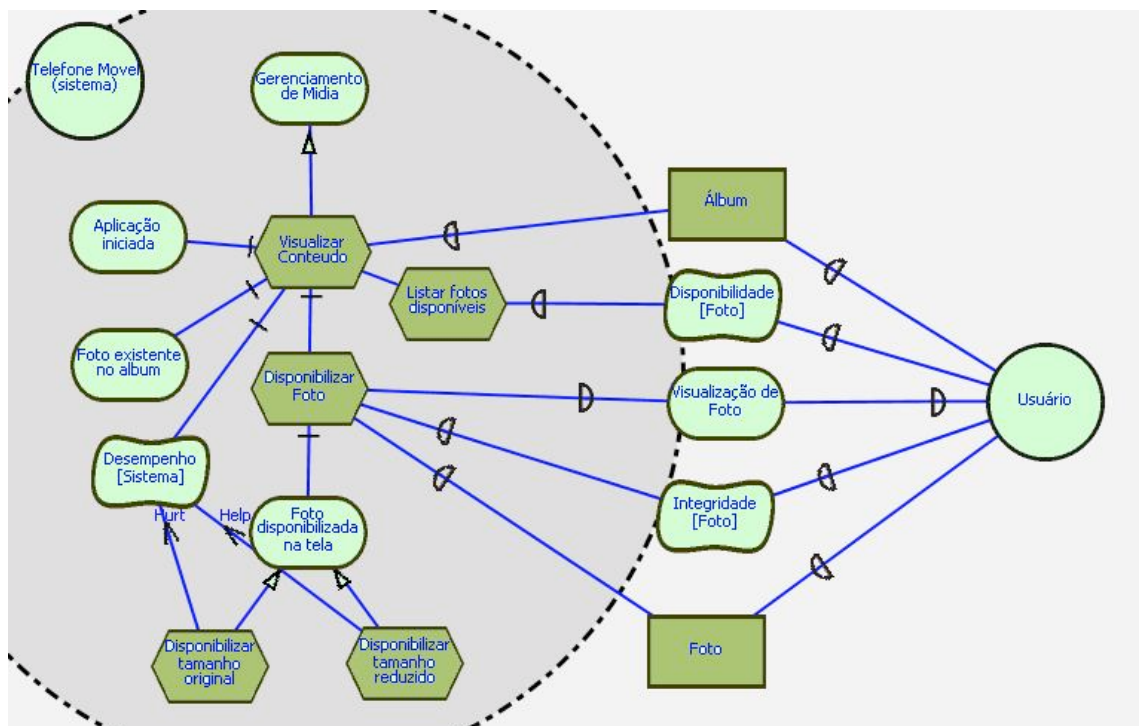


Figura 5.15 – Modelo SR do caso de uso Visualizar Conteúdo com possíveis elementos de serem *features*

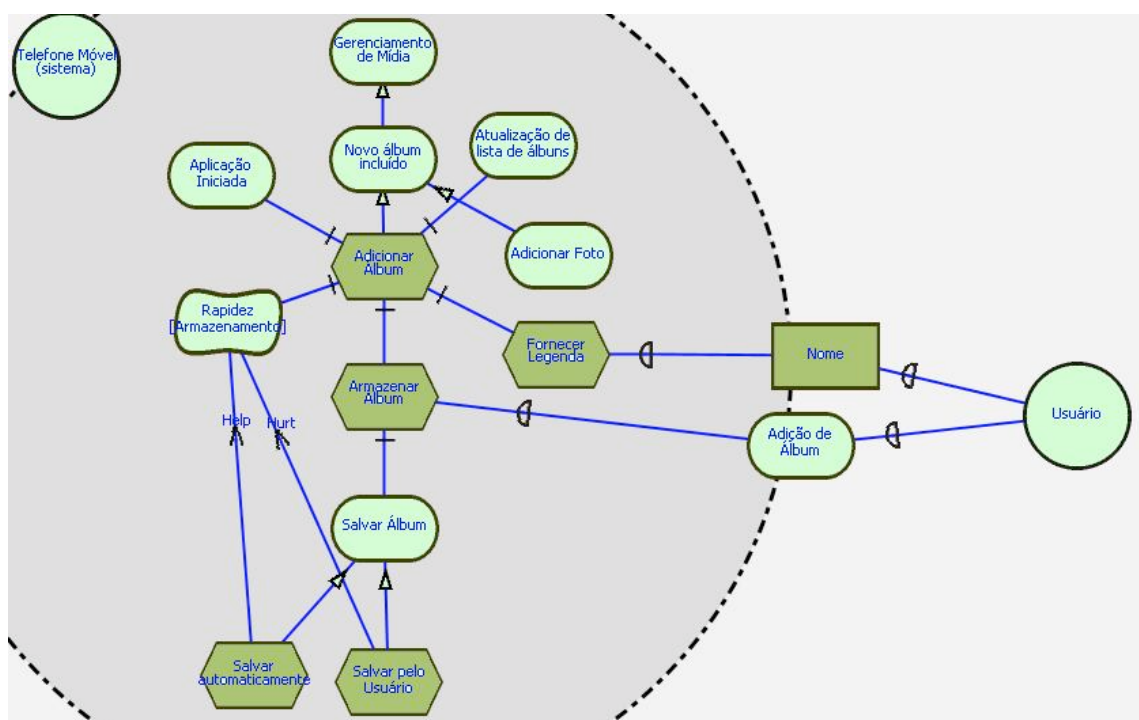


Figura 5.16 – Modelo SR do caso de uso Adicionar Álbum com possíveis elementos de serem *features*

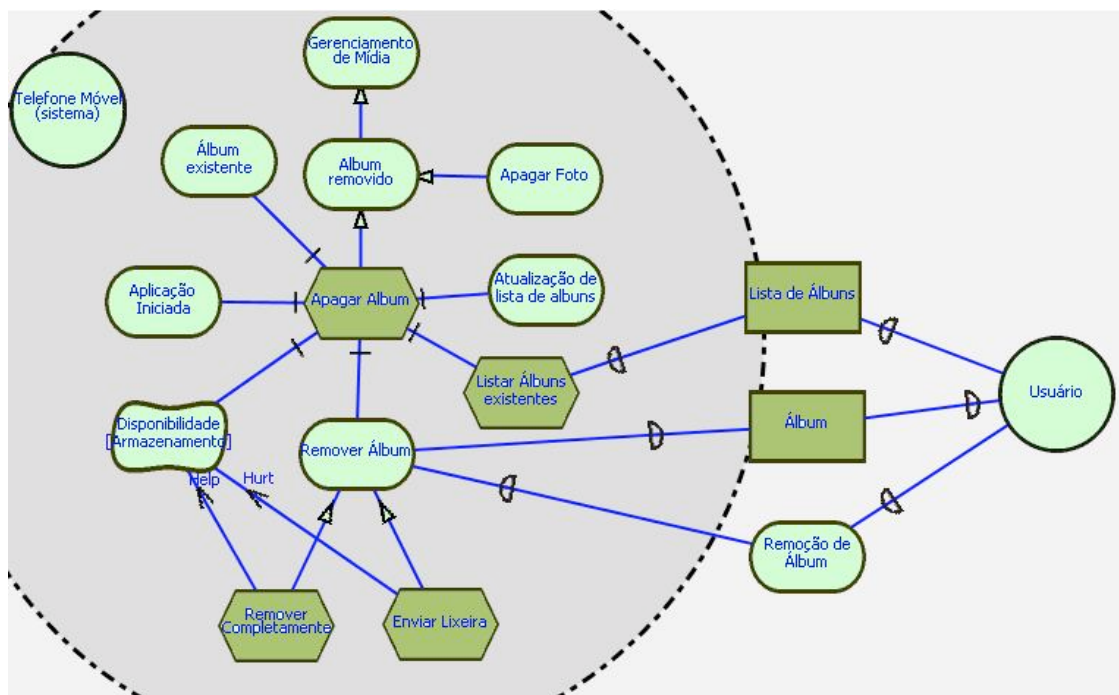


Figura 5.17 – Modelo SR do caso de uso Apagar Album com possíveis elementos de serem *features*

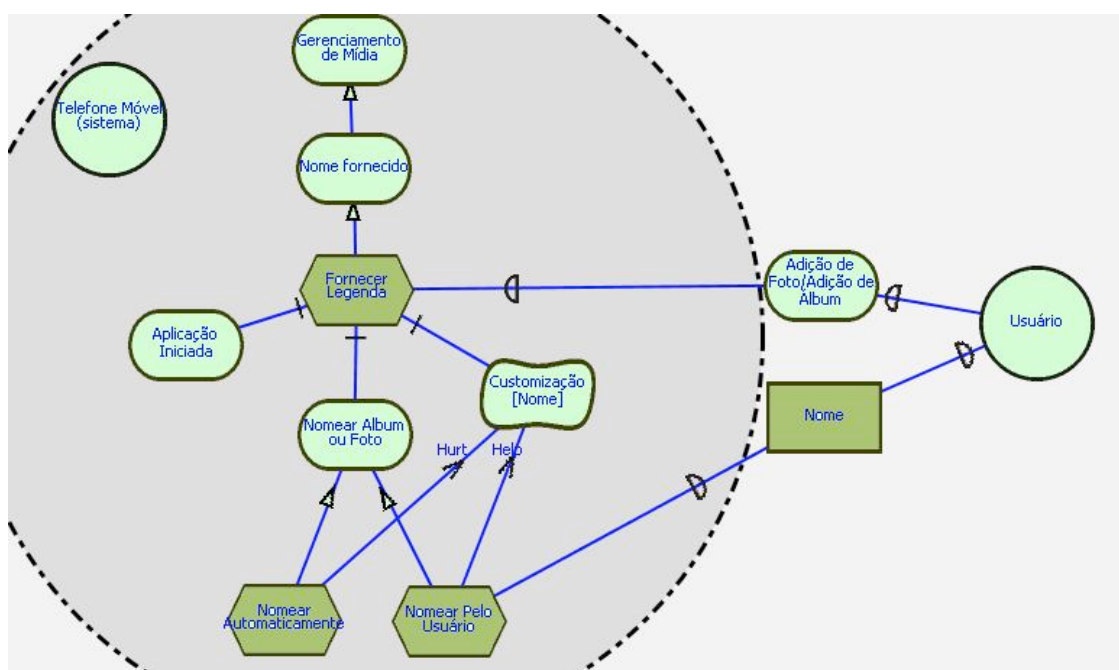


Figura 5.18 – Modelo SR do caso de uso Fornecer Legenda com possíveis elementos de serem *features*

5.3. Reengenharia do Modelo SR

Esta atividade tem como principal objetivo a adição de novas informações aos modelos SR gerados na atividade anterior, de acordo com a aplicação de algumas heurísticas e da linguagem *i*-c* (linguagem *i** com cardinalidade). O artefato produzido nesta atividade é o modelo SR com cardinalidade (vide Figura 5.19).

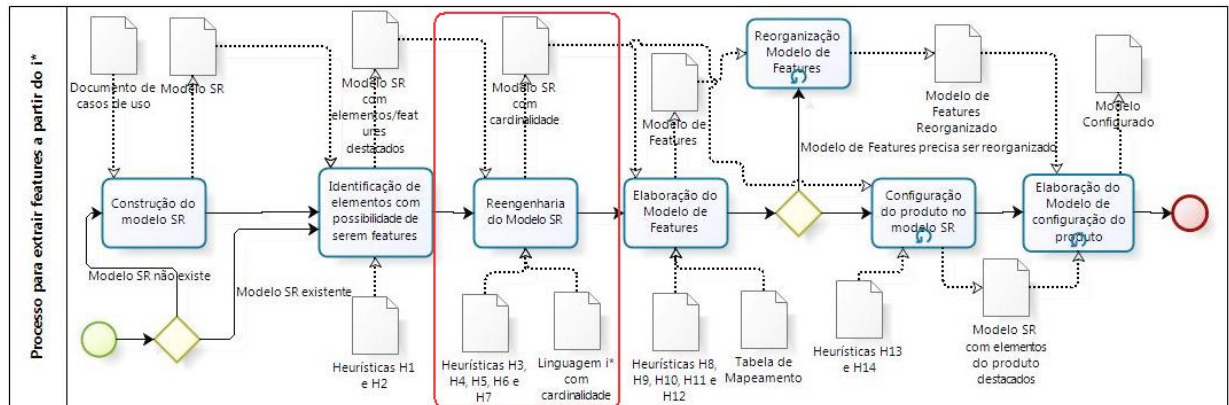


Figura 5.19 – Terceira atividade do processo G2FM

A Tabela 5.2 contém o resumo das heurísticas utilizadas nesta atividade.

Aplicando as heurísticas H3, H4, H5, H6 e H7 no modelo SR do caso de uso **Apagar Foto** (Figura 5.14), obtemos o modelo SR com cardinalidade da Figura 5.20.

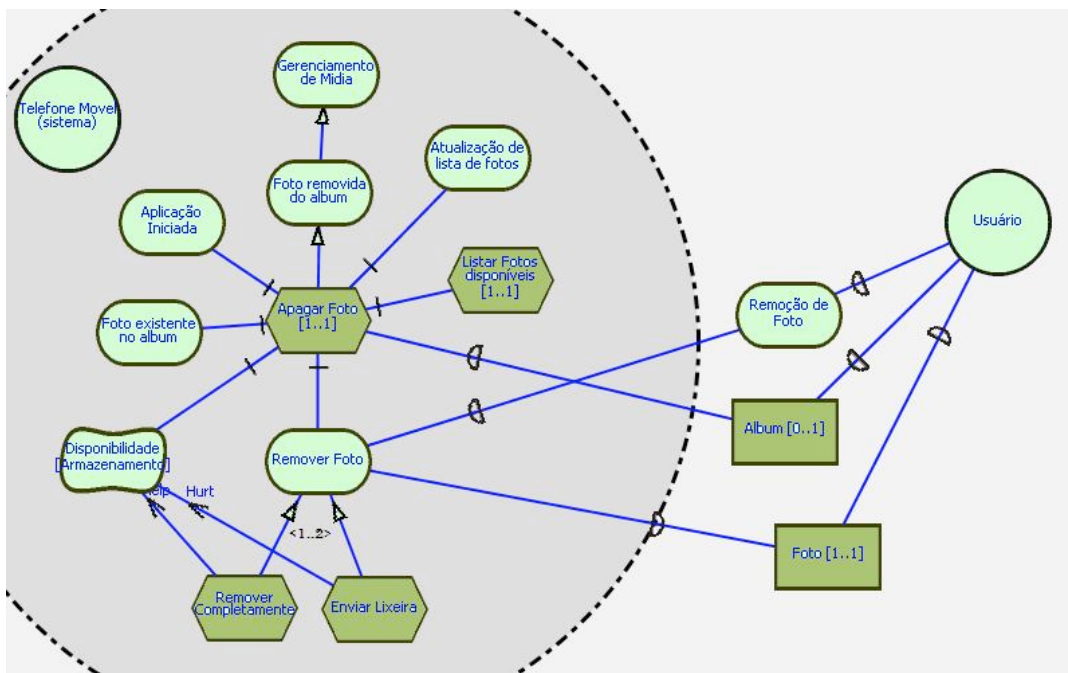


Figura 5.20 – Modelo SR com cardinalidade do caso de uso Apagar Foto

Tabela 5.2 – Resumo das heurísticas H3, H4, H5, H6 e H7

Heurística	Definição
H3	Todos os elementos do modelo SR que foram destacados como possíveis <i>features</i> serão reestruturados para conter cardinalidade.
H4	Se a quantidade de sub-elementos de uma ligação meio-fim que estão destacados como possíveis <i>features</i> , for maior do que 1, então esses sub-elementos serão agrupados e a cardinalidade pertencerá ao relacionamento. Se a quantidade for exatamente 1, então a cardinalidade pertencerá ao sub-elemento e não ao relacionamento.
H5	Os sub-elementos de uma ligação de decomposição de tarefas que estão destacados como possíveis <i>features</i> , serão modelados como <i>features</i> obrigatórias, significando que a cardinalidade pertencerá aos sub-elementos e terá a cardinalidade [1..1].
H6	As dependências (<i>dependum</i>), que estão destacadas como possíveis <i>features</i> , terão a cardinalidade no próprio elemento.
H7	Os elementos destacados como possíveis <i>features</i> , que não sejam sub-elementos de nenhum outro elemento que também seja feature, terão a cardinalidade pertencente a eles próprios e as <i>features</i> obtidas através deles estarão ligadas diretamente a <i>feature</i> raiz no modelo de <i>features</i> .

Aplicando as mesmas heurísticas no modelo SR do caso de uso **Visualizar Conteúdo** (Figura 5.15), obtemos o modelo SR com cardinalidade da Figura 5.21.

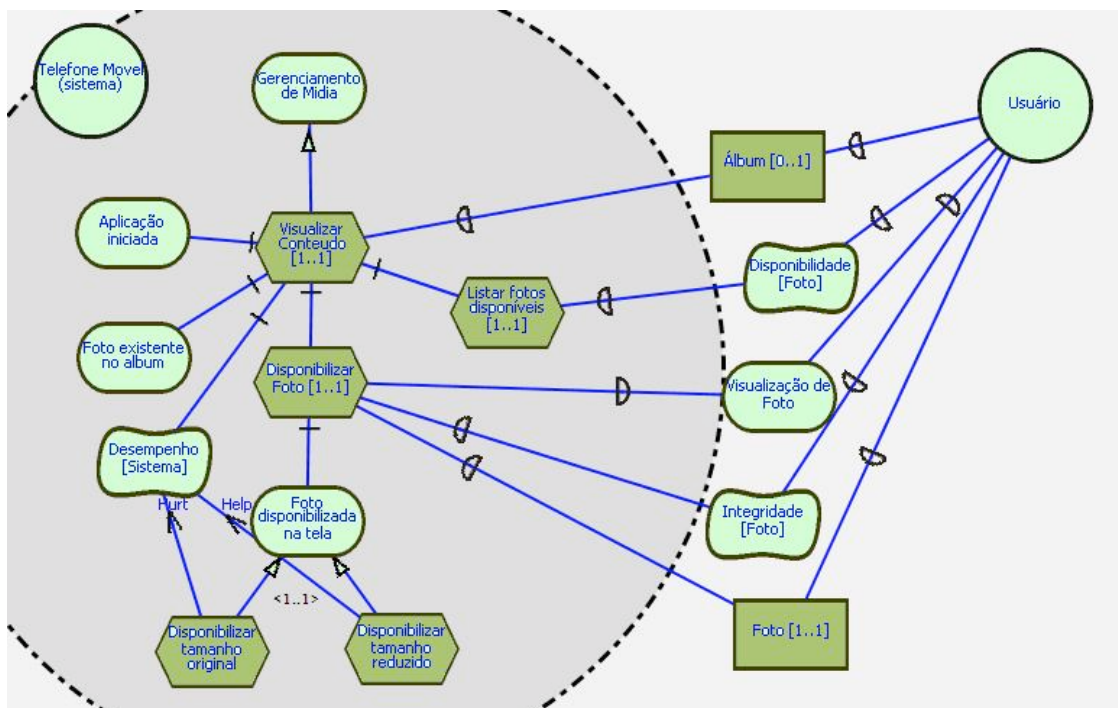


Figura 5.21 – Modelo SR com cardinalidade do caso de uso Visualizar Conteúdo

Repete-se a aplicação dessas heurísticas aos modelos SR dos casos de uso **Adicionar Álbum** (Figura 5.16), **Apagar Álbum** (Figura 5.17) e **Fornecer Legenda** (Figura 5.18), e obtém-se as figuras 5.22, 5.23 e 5.24, respectivamente.

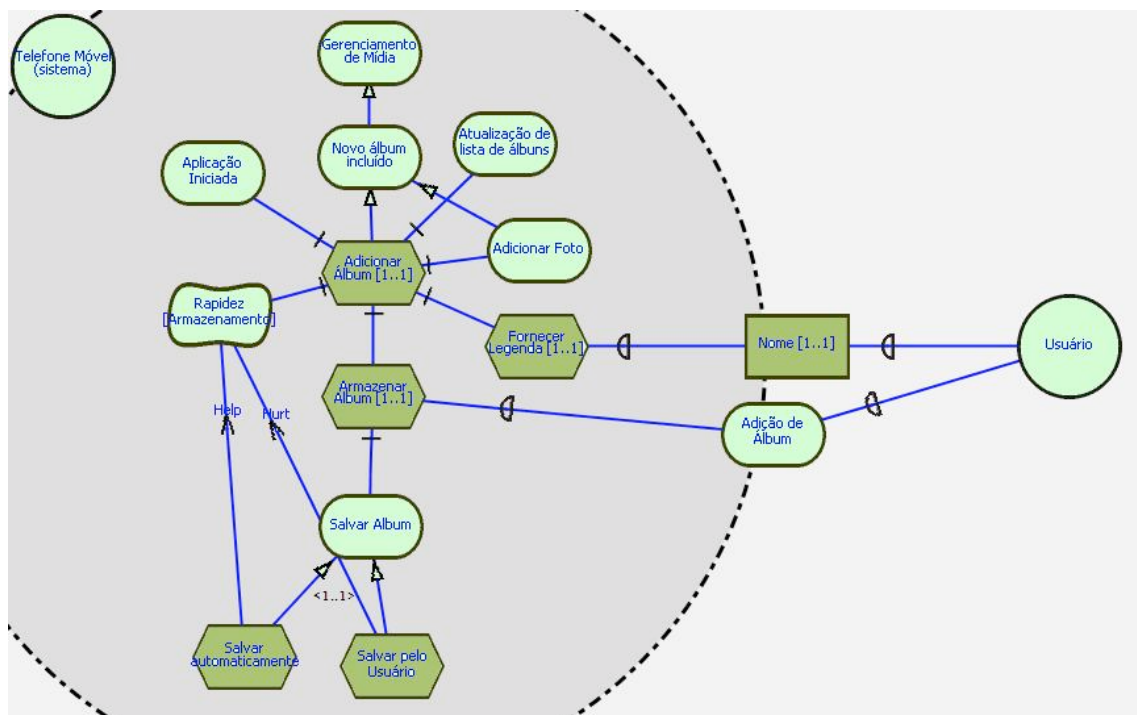


Figura 5.22 – Modelo SR com cardinalidade do caso de uso Adicionar Álbum

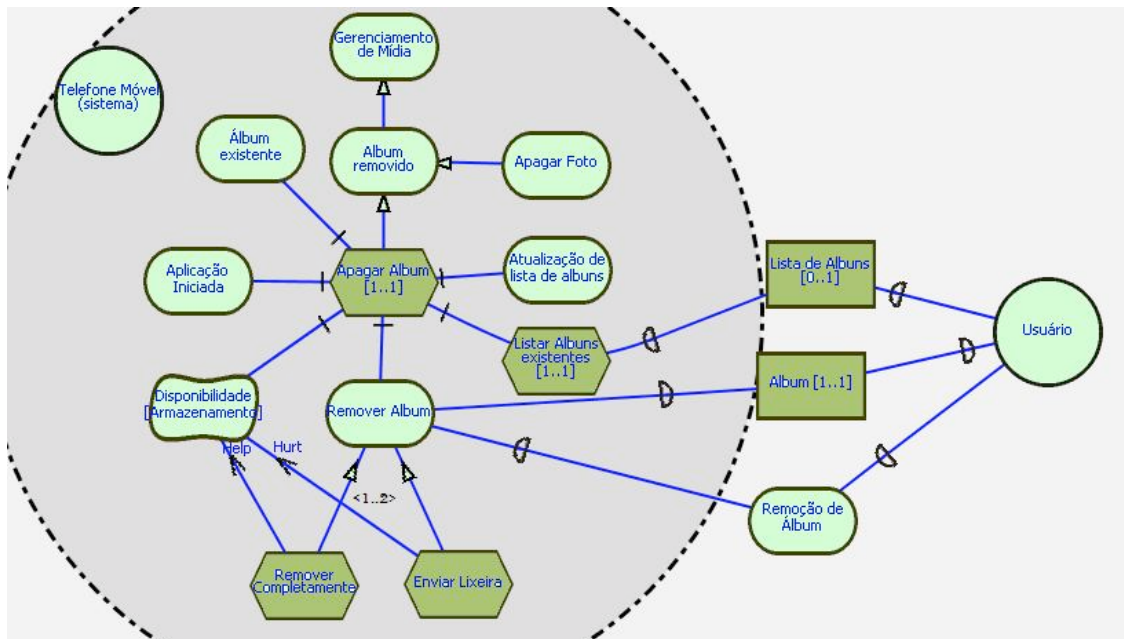


Figura 5.23 – Modelo SR com cardinalidade do caso de uso Apagar Álbum

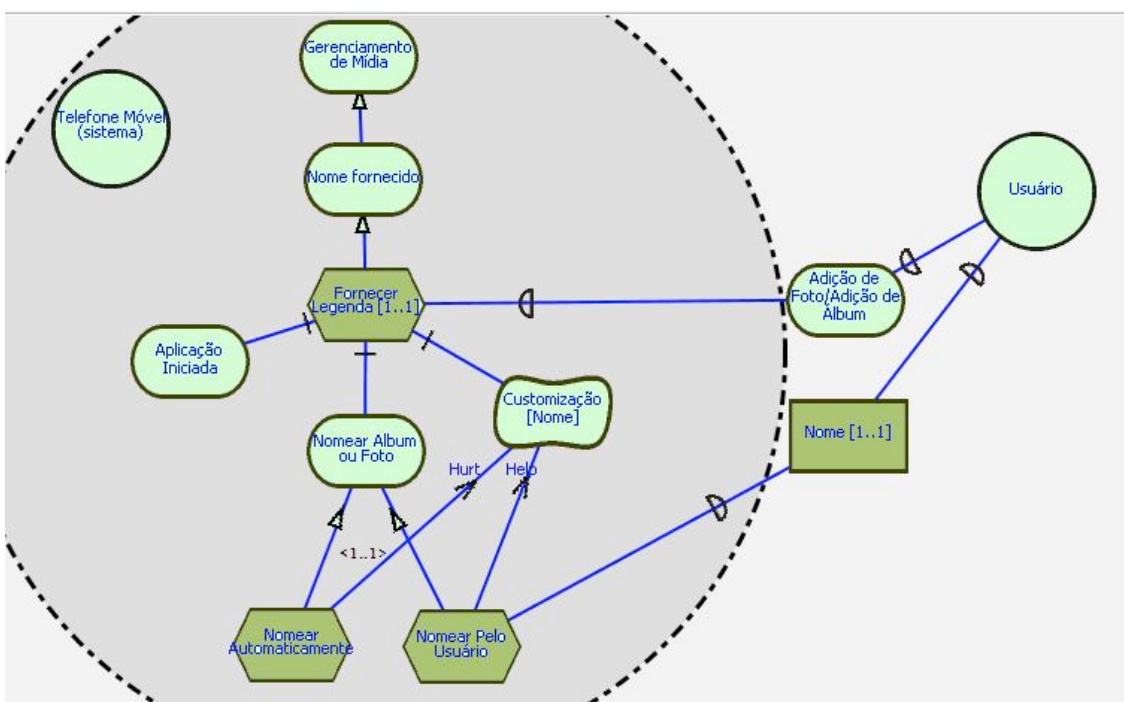


Figura 5.24 – Modelo SR com cardinalidade do caso de uso Fornecer Legenda

O modelo SR com cardinalidade do caso de uso **Adicionar Foto**, foi modelado no capítulo anterior e apresentado na Figura 4.13.