

## Utilizando Modelagem de Interação para Projetar uma Aplicação Móvel Multiusuário: Um Relato de Experiência

Adriana Lopes<sup>1</sup>, Natasha Valentim<sup>1</sup>, Bruna Moraes<sup>1</sup>, Renata Zilse<sup>2</sup>, Tayana Conte<sup>1</sup>

<sup>1</sup> Instituto de Computação – Universidade Federal do Amazonas (UFAM)  
Av. General Rodrigo Octávio, 6200, Coroado I – Manaus – AM – Brasil

<sup>2</sup> Samsung Research Brazil

{adriana, natashavalentim, bmf, tayana}@ufam.edu.br,  
renata.borges@samsung.com

**Abstract.** *Interaction models specify the structure and behavior of the system during user interaction. In this way, these models can improve the quality of use, since it helps software engineers to design the functionalities of the system. In addition, the models allow the identification of possible difficulties in the interaction. This paper reports the experience of a team project with interaction models during the development of a mobile application. The models were used as a basis for creating other artifacts, such as use cases. The results of this work show that team obtained a better understanding about mobile application, reducing rework in the construction of other artifacts.*

**Resumo.** *Modelos de interação especificam a estrutura e o comportamento do sistema durante a interação do usuário. Desta forma, estes modelos podem melhorar a qualidade de uso, uma vez que auxiliam engenheiros de software a projetarem as funcionalidades do sistema. Além disso, os modelos permitem a identificação de possíveis dificuldades de interação. Este artigo relata a experiência de uma equipe de projeto com modelos de interação durante o desenvolvimento de uma aplicação móvel. Os modelos foram usados como base para construir outros artefatos, como casos de uso. Os resultados deste trabalho mostram que houve melhor compreensão da equipe sobre a aplicação móvel, diminuindo o retrabalho na construção dos demais artefatos.*

### 1. Introdução

De acordo com Lira *et al.* (2015), a Engenharia de Software auxilia o desenvolvimento de sistemas e apoia as atividades para melhorar a qualidade do produto desenvolvido. Nesse sentido, existe uma preocupação da indústria de software em desenvolver sistemas que proporcionem aos usuários uma experiência interativa com qualidade [Souza e Barbosa, 2014]. Esta preocupação deve-se aos possíveis problemas na interação usuário-sistema, por exemplo, quando o usuário não compreende o que deve fazer ou não entende o resultado de uma funcionalidade executada.

Os problemas que afetam o uso dos sistemas podem influenciar em sua rejeição pelos usuários, sendo estes substituídos por outros sistemas [Rivero *et al.*, 2014]. Uma alternativa para reduzir os problemas na interação usuário-sistema é através do

emprego de modelagem de interação, durante a etapa de projeto do processo de desenvolvimento de software [Lopes *et al.*, 2015].

Falbo *et al.* (2008) afirmam que um dos principais meios de comunicação entre engenheiros de software é através de artefatos, sendo que cada um dos artefatos representa um modelo do sistema em desenvolvimento. Um desses artefatos é o modelo de interação, com o qual é possível descrever a estrutura e o comportamento do sistema durante a interação do usuário. O modelo de interação permite a detecção de eventuais problemas na interação usuário-sistema durante o projeto do sistema [Beaudouin-Lafon, 2000]. Por esta razão, o modelo de interação pode contribuir para a qualidade dos sistemas. Além disso, os engenheiros de software podem ter um artefato que auxilia na construção de outros artefatos no processo de desenvolvimento de sistemas, devido ao entendimento da interação usuário-sistema.

Neste contexto, uma das alternativas para apoiar o desenvolvimento de modelos de interação é através da MoLIC (*Modeling Language for Interaction as Conversation*) [Barbosa e Paula, 2003; Souza e Barbosa, 2014]. Com os diagramas MoLIC, a interação usuário-sistema é representada como uma metáfora da conversa que pode ocorrer entre o usuário e o sistema. Além disso, com a MoLIC pode-se representar todos os caminhos de interação, incluindo os caminhos alternativos para o usuário alcançar um mesmo objetivo [Paula *et al.*, 2003]. Assim, a MoLIC pode auxiliar no projeto de sistemas com foco na qualidade de uso, uma vez que os engenheiros de software podem projetar todas as alternativas de interação no sistema.

Este artigo apresenta um relato de experiência sobre o uso de modelos de interação com o apoio da MoLIC para projetar uma aplicação, a qual será nomeada como HCDP (*Home Care Development Project*). O artigo está estruturado da seguinte forma: a Seção 2 descreve os elementos de diagramas MoLIC. A Seção 3 apresenta algumas informações da aplicação HCDP e a ordem de construção dos artefatos utilizada na etapa de análise e projeto do processo de desenvolvimento. Seção 4 apresenta a análise realizada sobre o uso do modelo de interação durante a etapa de projeto da HCDP. Por fim, a Seção 5 apresenta as considerações finais deste relato de experiência e perspectivas futuras.

## **2. MoLIC (*Modeling Language for Interaction as Conversation*)**

A MoLIC, inicialmente proposta por Barbosa e Paula (2003) e continuada através de propostas de extensões [Sangiorgi e Barbosa, 2010; Souza e Barbosa, 2014], proporciona apoio ao desenvolvimento de modelos de interação. Com a MoLIC é possível projetar o sistema considerando a qualidade de uso, uma vez que o projetista compreende todos os caminhos de interação do usuário no sistema.

A Figura 1 apresenta um exemplo de diagrama MoLIC, o qual representa a modelagem de interação para um jogo educacional que auxilia no ensino dos tipos de defeitos que podem ser encontrados em um artefato [Lopes *et al.*, 2013]. Esses defeitos podem ser: Omissão (negligência de alguma informação necessária), Fato incorreto (utilização de informações que descrevem um fato que não é verdadeiro), Inconsistência (ocorrência de duas ou mais informações contraditórias entre si), Ambiguidade (definição vaga de uma informação, permitindo assim múltiplas interpretações) e Informação Estranha (informações desnecessárias). A descrição

destes defeitos em artefatos é baseada na taxonomia apresentada por Travassos *et al.* (1999). Conforme a modelagem de interação usuário-sistema representada no diagrama MoLIC, o jogador poderá selecionar um tipo de defeito de acordo com o exemplo adequado. Consequentemente, o jogador obterá pontuação caso este selecione o defeito corretamente, caso contrário, a pontuação não será considerada.

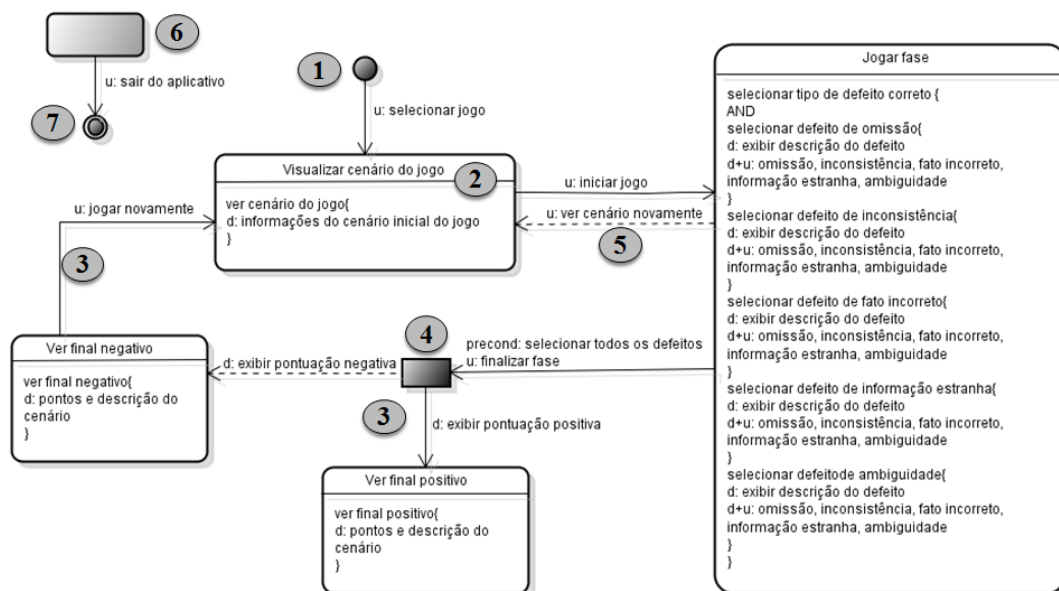


Figura 1. Exemplo de um diagrama MoLIC.

O diagrama MoLIC apresentado na Figura 1, possui os seguintes elementos:

**1. Ponto de Abertura** - indica o início da interação do usuário com a aplicação. Na Figura 1, o elemento é representado com um círculo preenchido com a cor preta.

**2. Cena** - mostra o momento em que o usuário decide como a interação deve prosseguir a partir das funcionalidades disponibilizadas no sistema. O elemento é representado como um retângulo de cantos arredondados e está dividido em dois compartimentos. No primeiro compartimento é detalhado o tópico da cena, que representa os diferentes objetivos do usuário durante a interação com a aplicação. Na Figura 1, por exemplo, é descrita a cena “Jogar fase”. No segundo compartimento as cenas são detalhadas através dos seguintes elementos:

**Signos** - representam a informação envolvida durante a interação, como campos e botões que estão inseridos na interface do sistema. Na Figura 1, temos os seguintes signos na cena “Jogar fase”: “omissão, inconsistência, fato incorreto, informação estranha e ambiguidade”.

**Falas** - especificam a interação com o signo. Caso o signo represente apenas informações, é utilizada a fala do designer (d), ou projetista (engenheiro de software que projetou a aplicação). Caso exista interação do usuário sobre a informação representada pelo engenheiro de software, é utilizada também a fala do usuário (d+u). No exemplo da Figura 1, para o diálogo “selecionar defeito de omissão”, ambos falam sobre o signo “d+u: omissão”. A definição de diálogo está descrita abaixo.

**Diálogos** - compõem a conversa (interação) sobre uma determinada atividade no sistema e são compostos por signos. Na Figura 1, por exemplo, temos o seguinte diálogo na cena “Jogar fase”: “selecionar defeito de omissão”.

**Estruturas de diálogos** - os diálogos podem ser compostos por outros diálogos, seguindo alguma estrutura, tais como: SEQ (representa os diálogos que devem ser utilizados em sequência), XOR (representa que apenas um diálogo deve ser utilizado), OR (representa a utilização de um ou mais diálogos) e AND (representa a utilização de todos os diálogos, porém não necessariamente em sequência). Na Figura 1, a estrutura AND representa a utilização dos diálogos “selecionar defeito de omissão”, “selecionar defeito de inconsistência”, “selecionar defeito de fato incorreto”, “selecionar defeito de ambiguidade” e “selecionar defeito de informação estranha”.

**3. Fala de Transição** - representa as mudanças de objetivos a partir da cena corrente, seja do usuário ou do sistema. Na fala de transição também são utilizados os conceitos de fala do usuário (u) e fala do designer/projetista (d). Na Figura 1, sobre fala do usuário, caso o jogador tenha um resultado negativo, o mesmo poderá jogar novamente. Em relação à fala do designer, caso o jogador obtenha uma pontuação mínima, o mesmo será direcionado pelo sistema para visualizar o resultado positivo.

**4. Processo do sistema** - representa o processamento interno de uma ação do usuário através de uma caixa preta. Isto ocorre somente quando é necessário o sistema interpretar a fala do usuário para fornecer o *feedback* adequado. Na Figura 1, por exemplo, é necessário o sistema interpretar os acertos e erros do jogador para que este seja direcionado para o *feedback* correto.

**5. Fala de recuperação da ruptura** – utilizada em uma situação de recuperação de ruptura que pode ocorrer durante a interação, através de uma seta tracejada. De maneira semelhante à fala de transição, também são utilizados os conceitos de fala do usuário (u) e fala do designer (d). Na Figura 1, é descrita a fala de recuperação de ruptura “u: ver cenário novamente”.

**6. Acesso ubíquo** - representa a oportunidade de o usuário usar outra funcionalidade, a partir de qualquer outra cena, para atingir um objetivo diferente do atual. Na Figura 1, o jogador pode a qualquer momento sair do jogo.

**7. Ponto de encerramento** - indica o fim da interação. Na Figura 1, o elemento é representado com um círculo negro preenchido em um círculo sem preenchimento.

A partir dos diagramas MoLIC, é possível construir outros artefatos durante o processo de desenvolvimento. Além disso, os engenheiros de software podem discutir sobre a interação usuário-sistema representada no diagrama MoLIC com o objetivo de colaborar com a melhoria das informações representadas, conforme sugerem Falbo *et al.* (2008) em relação aos modelos. A Figura 2 apresenta *mockups* construídos com base no diagrama MoLIC apresentado na Figura 1. A seguir são descritas as decisões para o mapeamento do diagrama MoLIC para os *mockups*, conforme apresenta a Figura 2.

- A cena “Visualizar cenário do jogo” foi mapeada para a tela “inicial do jogo” (1). A decisão dos nomes dos *mockups* é de responsabilidade do engenheiro de

software, pois o mesmo define a melhor forma de representar os objetivos do usuário.

- A fala de transição do usuário “u: iniciar jogo” foi representada através do botão “INICIAR JOGO” (2).
- A fala de recuperação de ruptura do usuário “u: ver cenário novamente” foi representada através do link “Ver cenário novamente” para que o usuário seja direcionado à tela “inicial do jogo” (3).
- Neste contexto, a fala de transição do engenheiro de software (representando na MoLIC como designer) “d: exibir pontuação positiva” após o processamento do sistema foi representada nos *mockups* como *feedback* do usuário, no caso do cenário final positivo do jogo (4).
- Já os signs “d: exibir descrição do defeito” e “d+u: omissão, inconsistência, fato incorreto, ambiguidade e informação estranha” foram mapeados, respectivamente, para representar informações e botões em que o usuário possa classificar os defeitos (5).

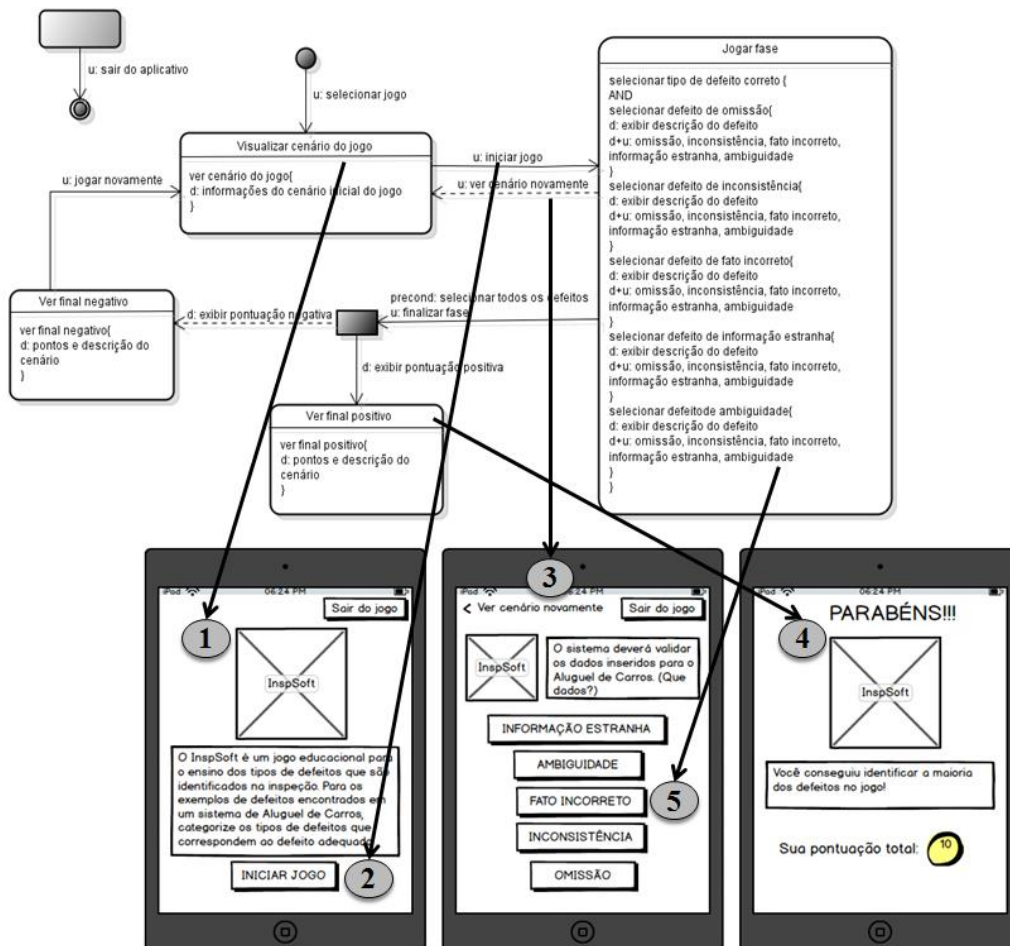


Figura 2. Mapeamento do diagrama MoLIC para os *mockups*.

A MoLIC foi escolhida para apoiar a modelagem de interação na aplicação HCDP. Assim, este artigo descreve a experiência de uma equipe de projeto com a MoLIC, demonstrando a maneira de utilização do modelo de interação entre os engenheiros de software e as lições aprendidas da equipe.

### 3. Aplicação HCDP e Práticas Adotadas para o Desenvolvimento dos Artefatos na Etapa de Análise e Projeto

A HCDP fez parte de um projeto de cooperação entre a Universidade Federal do Amazonas, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS) e a Samsung Research Brazil. A HCDP, uma aplicação móvel multiusuário, foi projetada e avaliada pela equipe de projeto. A motivação para o desenvolvimento desta aplicação é a dificuldade que familiares têm em gerenciar adequadamente as atividades da rotina de um idoso. Assim, foi desenvolvida a aplicação HCDP para apoiar um grupo de pessoas, sendo familiares e profissionais, que colaborativamente cuidam de um idoso.

Três equipes distribuídas trabalharam no desenvolvimento da aplicação HCDP. Uma das equipes é responsável pela HCDP (cliente). A equipe responsável definiu todas as funcionalidades a serem desenvolvidas. Outra equipe realizou as atividades relacionadas ao desenvolvimento da aplicação. Além disso, uma equipe realizou as atividades relacionadas à análise e projeto da aplicação, composta por 5 engenheiros de software e 1 gerente do projeto. Os membros desta possuíam experiência em análise e projeto de aplicações. A experiência relatada neste artigo refere-se ao trabalho da equipe de análise e projeto da aplicação HCDP.

Para a aplicação HCDP foram adotadas algumas práticas da metodologia Scrum [Scrum Alliance, 2016]. As etapas de análise e projeto da aplicação foram divididas em Sprints de duração fixa. O projeto teve duração de 6 meses, sendo dividido em 6 Sprints. A Tabela 1 apresenta um resumo das atividades realizadas durante a execução do projeto. A seguir, são descritas as atividades realizadas em cada Sprint para compreensão do contexto em que foi utilizada a modelagem de interação.

Na Sprint 1, a equipe realizou as atividades da etapa de análise da aplicação HCDP. Primeiramente foi aplicada a técnica Personas para identificar as necessidades de diferentes tipos de usuários [Castro *et al.*, 2008]. As personas foram criadas com base nos cenários definidos pela equipe responsável com o objetivo de representar diferentes usuários. As personas foram validadas com usuários potenciais da aplicação, pois estes mostraram interesse em obter uma aplicação com o mesmo propósito da HCDP. Após isto foram desenvolvidos cenários, que descreviam os detalhes da relação das personas com os dispositivos móveis. A partir das personas e cenários desenvolvidos foi realizada a identificação de requisitos do sistema. Em seguida foi desenvolvido o diagrama de caso de uso da HCDP. Após isto, a equipe responsável pelo HCDP e a equipe de análise e projeto definiram em conjunto a priorização de desenvolvimento dos casos de uso, ou seja, a ordem em que os casos de uso seriam desenvolvidos nas demais Sprints.

Na Sprint 2 começaram as atividades da etapa de projeto. Primeiramente a equipe selecionou um dos casos de uso, conforme a ordem de prioridade definida na Sprint 1. Após isto, foram desenvolvidos todos os artefatos solicitados pela equipe responsável que representassem o caso de uso da HCDP, que são: **Diagramas MoLIC**

- para projetar a interação usuário-sistema; **Especificação de Casos de uso** - para representar um conjunto de ações realizadas pelo sistema, incluindo os fluxos principais, alternativos e de exceção; **Mockups** - para representar a interface do sistema; **Diagramas de atividades** - para fornecer uma visualização do comportamento do sistema através da sequência de ações nas atividades.

• **Tabela 1: Atividades de Análise e Projeto Realizadas em cada Sprint.**

Sprint	Atividade
Sprint 1	Detalhamento de Persona
	Criação de Cenários
	Identificação de Requisitos
	Criação de Diagrama de Casos de Uso
	Priorização de Casos de Uso
	Reunião de Retrospectiva com a Equipe de Análise e Equipe Responsável sobre as Lições Aprendidas
Sprint 2	Modelagem de Interação para o Caso de Uso A
	Especificação de Caso de Uso A
	Desenvolvimento de Mockups para o Caso de Uso A
	Desenvolvimento de Diagramas de Atividades para o Caso de Uso A
	Reunião de Retrospectiva com a Equipe de Análise e Equipe Responsável sobre as Lições Aprendidas
Sprint 3	Modelagem de Interação para o Caso de Uso B (Sprint 3)
	Modelagem de Interação para o Caso de Uso C (Sprint 4)
	Modelagem de Interação para o Caso de Uso D (Sprint 5)
Sprint 4	Especificação de Caso de Uso B
	Especificação de Caso de Uso C
	Especificação de Caso de Uso D
Sprint 5	Desenvolvimento de Mockups para o Caso de Uso B (Sprint 3) Desenvolvimento de Mockups para o Caso de Uso C (Sprint 4) Desenvolvimento de Mockups para o Caso de Uso D (Sprint 5)
	Desenvolvimento de Diagramas de Atividades para o Caso de Uso B (Sprint 3)
	Desenvolvimento de Diagramas de Atividades para o Caso de Uso C (Sprint 4)
Sprint 6	Desenvolvimento de Diagramas de Atividades para o Caso de Uso D (Sprint 5)
	Reunião de Retrospectiva com a Equipe de Análise e Equipe Responsável sobre as Lições Aprendidas
	Melhorias nos Artefatos das Sprints Anteriores
Sprint 6	Reunião de Retrospectiva Geral com a Equipe de Análise e Equipe Responsável sobre as Lições Aprendidas

Na atividade de modelagem de interação, a equipe desenvolveu os diagramas MoLIC para representar todos os caminhos de interação entre o usuário e o sistema. Na especificação de casos de uso, a equipe descreveu os casos de uso considerando os diferentes fluxos de execução. Foram desenvolvidos também os *mockups* necessários para representar a interface. Após o desenvolvimento dos artefatos na Sprint 2, estes foram analisados por engenheiros de softwares diferentes, ou seja, o engenheiro de

software que construiu o artefato não foi o mesmo que analisou. Tal prática teve como objetivo identificar inconsistências nas informações representadas nos artefatos.

Em relação às Sprints posteriores, as mesmas atividades foram realizadas nas Sprints 3, 4 e 5, como mostra a Tabela 1. Já na Sprint 6, foram realizadas melhorias em alguns artefatos de Sprints anteriores.

Além disso, ao final de cada Sprint, foi realizada a atividade de reunião de retrospectiva da equipe, sendo discutidas as lições aprendidas. Nestas reuniões, foi utilizada a abordagem PABC-Pattern para codificar lições aprendidas pela equipe [Rabelo *et al.*, 2014]. Os principais elementos da PABC-Pattern são: **Título** - descrição do nome da lição de forma resumida; **Situação** - detalhamento da situação no projeto; **Causa da Situação** - motivo da ocorrência da situação no projeto; **Consequência da Situação** - descrição da consequência da situação no projeto; **Ação** - detalha a solução para a situação, ou seja, esclarece uma atividade que foi aplicada para resolver tal situação; **Benefício** - descreve os efeitos (positivos e/ou negativos) que foram causados pela ação ou pela situação no projeto.

Mais detalhes da abordagem PABC-Pattern podem ser vistos no trabalho de Rabelo *et al.* (2014). Ao encerrar a etapa de projeto da HCDP, houve um total de 44 lições codificadas com a abordagem PABC-Pattern.

#### 4. Análise do Uso dos Modelos de Interação no Projeto da Aplicação HCDP

O objetivo desta análise visa compreender as percepções dos engenheiros de software em relação ao entendimento dos diagramas MoLIC e o processo utilizado para desenvolver tais artefatos. A análise está relacionada com as Sprints 2 e 3. Foram selecionados os principais artefatos desenvolvidos em tais Sprints, como: requisitos e diagrama e especificação de caso de uso, cenário, diagramas MoLIC, *mockups*, diagramas de atividades, revisão dos artefatos e documentos de lições aprendidas.

##### 4.1. Análise da construção dos artefatos na Sprint 2

Na Sprint 2, a ordem de construção adotada para *mockups*, diagramas MoLIC, especificação de casos de uso e diagramas de atividades é apresentada na Figura 3.

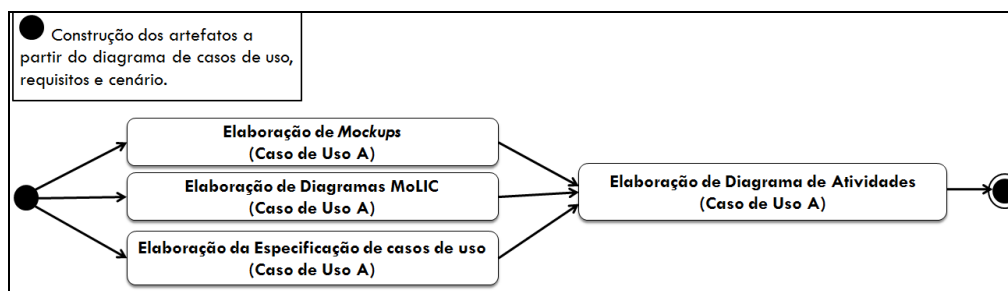


Figura 3. Atividades realizadas para a construção de artefatos na Sprint 2.

Nota-se que a construção de *mockups*, diagramas MoLIC e especificação de casos de uso foi realizada de forma paralela, apenas com base no diagrama de caso de uso, requisitos e no cenário. Em seguida, o diagrama de atividades foi construído, com base nos três artefatos desenvolvidos. Ao final da Sprint 2, após o desenvolvimento de



todos os artefatos, foi realizada a reunião de retrospectiva da equipe de projeto da HCDP. A seguir são discutidas algumas lições identificadas na Sprint 2.

Na Sprint 2, existem 15 documentos que relatam as lições aprendidas codificadas com auxílio da abordagem PABC-Pattern, com um total de 7 documentos relacionados com a construção dos artefatos. Foram selecionadas as três principais lições aprendidas que relatam problemas na construção dos artefatos. Uma destas lições aprendidas tem por **Título** Retrabalho ao manter o padrão para os artefatos, a qual foi gerada pela seguinte **Situação**: “*Ao desenvolver os artefatos separados após a reunião, existe um trabalho excessivo em padronizar os casos de usos, mockups, modelagem de interação e diagrama de atividade.*”. A equipe de projeto definiu a seguinte estratégia de **Ação**: “*Uma análise em conjunto onde será definida a ideia geral dos requisitos antes de criar os artefatos através de um documento formal*”. Assim, a equipe concluiu que pode-se reduzir o retrabalho em relação às inconsistências entre tais artefatos.

Nesse contexto, foi selecionada a lição aprendida que tem por **Título** Dependência entre atividades pode ocasionar atrasos no projeto, que ocorreu pela seguinte **Situação**: “*A melhoria do caso de uso dependia da entrega dos mockups. A entrega dos mockups chegou atrasada*”. Diante disto, foi observada a seguinte **Consequência da Situação**: “*Retrabalho. Atraso da atividade seguinte que deveria ser entregue a outro participante, a qual a outra deveria fazer a modelagem de interação com o caso de uso*”. Sobre as estratégias de **Ação**, para esta codificação, foi definido: “*Quando a atividade de uma pessoa depende da atividade de outra pessoa, deve-se haver um cuidado e esforço para que não se atrase*”. Assim, percebe-se que houve atraso para a melhoria da especificação de casos de uso, uma vez que tal artefato deveria representar as mesmas informações nos *mockups* e diagramas MoLIC.

Ainda neste contexto, foi selecionada a lição aprendida que tem por **Título** Falta de rastreabilidade entre os mockups e o caso de uso, a qual ocorreu pela seguinte **Situação**: “*Falta de rastreabilidade ou mapeamento para entender o que está no mockup e o que isto representa no caso de uso*”. A equipe de projeto definiu a seguinte estratégia de **Ação**: “*Ver quais são os mockups representados no caso de uso e integrar mais os dois artefatos*”. Portanto, houve falta de entendimento de alguns engenheiros de software em relação à compreensão dos requisitos para o desenvolvimento de diferentes artefatos. Diante disto, houve inconsistências ao representar as informações da HCDP entre os artefatos desenvolvidos apenas com base nos requisitos, diagrama de caso de uso e cenário.

Ressalta-se que na Sprint 2 os diagramas MoLIC não foram utilizados como base para a construção dos demais artefatos. Portanto, não houve um artefato que foca nas informações essenciais na interação usuário-sistema para a construção dos outros artefatos. Por esta razão, houve retrabalho da equipe em relação à consistência dos artefatos. Assim, a equipe realizou uma mudança na ordem de construção dos artefatos. Esta mudança é apresentada na próxima subseção.

#### **4.2. Análise da construção dos artefatos na Sprint 3**

Na Sprint 3, a ordem de construção em relação aos diagramas MoLIC, *mockups*, especificação de casos de uso e diagramas de atividades é apresentada na Figura 4.

Esta ordem de construção é diferente em relação à ordem construção adotada na Sprint 2. A construção de *mockups*, diagrama de atividades e especificação de casos de uso foi realizada com base nos diagramas MoLIC, requisitos, diagramas de caso de uso, além do cenário. Esta mudança ocorreu para usar os diagramas MoLIC como base para a construção de outros artefatos.

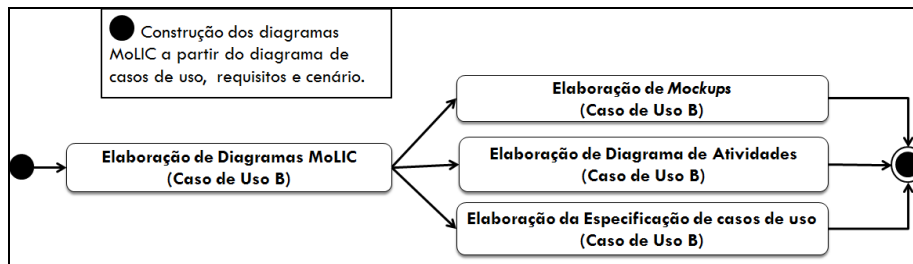


Figura 4. Atividades realizadas para a construção de artefatos na Sprint 3.

No Sprint 3, a equipe de projeto realizou a elaboração conjunta da modelagem de interação, ou seja, o engenheiro de software responsável pela modelagem da interação construía os diagramas MoLIC e os demais engenheiros acompanharam e discutiram sobre a modelagem realizada, projetando soluções alternativas de interação para o usuário. Com a construção de modelos de interação de maneira conjunta, foi possível identificar possíveis problemas na interação usuário-sistema, sendo desenvolvidas soluções para que tais problemas não ocorram. Após isto, os *mockups*, diagrama de atividades e especificação de casos foram construídos, com base nos diagramas MoLIC desenvolvidos. Ao final da Sprint 3, também foi realizada a reunião de retrospectiva da equipe de projeto da HCDP. A seguir são discutidas algumas lições identificadas na Sprint 3.

Em relação às lições aprendidas da equipe nesta Sprint, existem 12 documentos que relatam a codificação das lições aprendidas, com um total de 4 documentos relacionados com a construção dos artefatos. Foram selecionadas as 3 principais lições aprendidas sobre o uso de diagramas MoLIC como base para a construção dos demais artefatos pela equipe.

Uma lição aprendida selecionada sobre ao uso da MoLIC na Sprint 3 tem por **Título** Uso de diagramas MoLIC no início, a qual foi ocasionada pela seguinte **Situação**: “O uso dos diagramas MoLIC no início possibilitou a reflexão sobre o caso de uso. E também ver que é necessário o uso em conjunto com a interface”. Foi atribuída a esta lição aprendida codificada a seguinte **Consequência da Situação**: “Com base nos diagramas MoLIC, a equipe acabou obtendo tempo e ao final da sprint as modificações foram finalizadas de maneira simples”. No entanto, as modificações realizadas nos *mockups* refletem em modificações que devem ser realizadas no diagrama MoLIC, uma vez que tais modelos podem ser projetados em conjunto (Lopes *et al.*, 2015). Em relação às estratégias de **Ação**, para esta codificação, não houve um plano de ação a ser definido, pois tal experiência foi considerada como positiva.

Nesse contexto, foi selecionada a lição aprendida que tem por **Título** Excesso de informação no documento de interação, que ocorreu pela seguinte **Situação**: “O uso dos diagramas MoLIC no início possibilitou a reflexão sobre o caso de uso. E também entender que é necessário o uso em conjunto com a interface”. Diante disto, foi

observada a seguinte **Consequência da Situação**: “*O fato de a modelagem tornar-se grande talvez não seja o mais adequado do ponto de vista do responsável pela modelagem. Mas para equipe, o detalhamento ajudou no projeto*”. Sobre as estratégias de **Ação**, para esta lição aprendida foi definido: “*Redução no tamanho do documento que contém a modelagem de interação, mantendo apenas informações necessárias*”. Assim, percebe-se que o uso da modelagem de interação com o apoio da MoLIC foi útil para o entendimento da equipe sobre a interação usuário-sistema. Esta prática foi importante na etapa de projeto, pois alguns problemas do produto final (software) podem estar relacionados à falta de preocupação dos profissionais sobre as perspectivas dos usuários.

Além disso, outra lição aprendida selecionada tem por **Título** Reflexão sobre consumo de tempo em atividades do projeto, a qual ocorreu pela seguinte **Situação**: “*Benefícios em atividades que consumiam tempo de projeto e não eram bem compreendidas pelos participantes*”. A equipe de projeto definiu a seguinte estratégia de **Ação**: “*Participantes tiveram que refletir o seguinte: Quando não se tinha um padrão de interface, começar pela modelagem de interação fazia sentido. Após se ter um padrão de interface, analisar a partir dos mockups as alterações na modelagem de interação estava mais no contexto*”. Portanto, a equipe de projeto concluiu que apesar do uso da modelagem de interação inicial ser útil para compreender as funcionalidades do sistema, as alterações a partir dos *mockups* para a compreensão de todas as funcionalidades são úteis. Nota-se que neste projeto houve uma melhor compreensão das funcionalidades a partir dos diagramas MoLIC em conjunto com os *mockups*.

Com base nestas lições aprendidas observa-se como aspectos positivos: (a) o uso dos diagramas MoLIC no início para compreender as funcionalidades e (b) seu uso como base para a construção de outros artefatos. O excesso de informação no documento de interação foi considerado um aspecto negativo para o projetista responsável. No entanto, este mesmo aspecto foi considerado como positivo pelos demais projetistas devido à melhor compreensão de informações para construção dos outros artefatos.

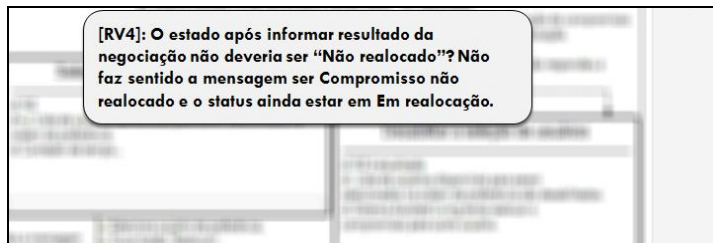
### 4.3. Revisão dos Diagramas MoLIC na Sprint 3

Além dos documentos com as lições aprendidas codificadas, analisou-se o documento contendo toda a revisão da modelagem de interação da Sprint 3. Foi escolhido apenas o documento com a revisão da modelagem de interação da Sprint 3 devido a este ter sido utilizado pelos engenheiros de software como base para a construção de outros artefatos. Este documento incluiu os diagramas MoLIC e outras informações necessárias para entender quais os problemas podem ter ocorrido após a definição da modelagem de interação no início da Sprint.

Como a HCDP possui informações confidenciais, as figuras com parte dos diagramas MoLIC foram embaçadas, sendo disponibilizadas apenas as revisões de engenheiros de software em relação aos diagramas. A revisão dos diagramas MoLIC foi realizada com base na taxonomia apresentada por Travassos *et al.* (1999).

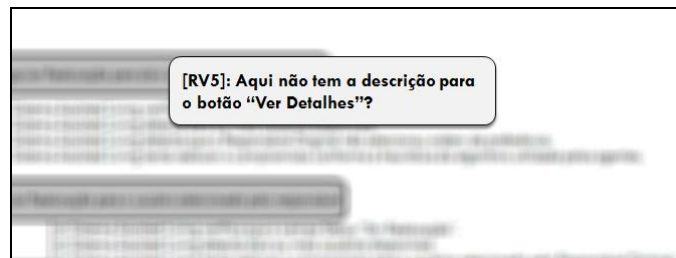
Na Figura 5, no documento de revisão dos diagramas MoLIC, observa-se a identificação de um defeito do tipo inconsistência de mensagem exibida para o usuário (informação da mensagem foi representada através do elemento signo da MoLIC).

Caso este defeito não fosse identificado, tal defeito poderia impactar a construção de *mockups* e especificação de casos de uso, pois a informação seria mapeada de maneira inconsistente com os requisitos do sistema, diagramas de caso de uso e o cenário.



**Figura 5. Revisão do engenheiro de software responsável pelas regras descritas em todos os artefatos – Defeito de inconsistência.**

Em relação à construção de *mockups*, foi identificado, por exemplo, defeito de omissão de informações como mostra a Figura 6, pois não foi representada a opção de exibição de detalhes para informação ao usuário (opção de detalhes de informação foi representada através do elemento signo da MoLIC). Esta opção seria mapeada no *mockup* como um botão da interface.



**Figura 6: Revisão do engenheiro de software responsável pelos *mockups* – Defeito de omissão.**

Estes defeitos apresentados na Figura 5 e na Figura 6 se enquadram nos defeitos encontrados no estudo de caso apresentado no trabalho de Lopes *et al.* (2015), com base na taxonomia apresentada por Travassos *et al.* (1999). Desta forma, é importante que haja a revisão dos modelos de interação, pois defeitos podem prejudicar o entendimento dos profissionais sobre artefatos construídos com base nos diagramas MoLIC. Tal prática foi realizada no documento da modelagem de interação e nos demais artefatos antes de serem entregues para a equipe responsável pela HCDP.

Embora a atividade de revisão tenha sido realizada nos diagramas MoLIC por outro engenheiro de software, nota-se que esta não foi realizada com uma técnica específica para a revisão de diagramas MoLIC. Possivelmente, o uso de uma técnica específica poderia ter auxiliado na identificação dos defeitos antes da construção de *mockups* e especificação de casos de uso. Assim, foi necessária a correção dos diagramas MoLIC, especificação de casos de uso e *mockups*, devido a tais defeitos.

#### 4.4. Discussão dos Resultados

Foi analisado o uso da modelagem de interação nas Sprints 2 e 3, devido à redução do retrabalho que a equipe obteve na Sprint 3 em comparação com a Sprint 2. Entende-se

que ao projetar a interação usuário-sistema é possível ter a compreensão sobre a interface. Assim, modelos de interação e *mockups* podem ser projetados de maneira conjunta [Lopes *et al.*, 2015]. Em relação às demais Sprints, foi utilizada a mesma ordem de construção dos artefatos da Sprint 3.

Através dos resultados desta análise foi possível obter indícios sobre os benefícios da utilização de diagramas MoLIC na etapa de projeto. Porém, caso os diagramas MoLIC possuam defeitos, isto prejudicará o entendimento dos profissionais envolvidos, ocorrendo retrabalho devido às correções realizadas, tanto nos diagramas MoLIC quanto nos artefatos desenvolvidos com base nos diagramas MoLIC.

Através da experiência desta equipe de projeto com a aplicação HCDP, sugere-se o uso de modelagem de interação no início da construção de outros artefatos. Esta prática pode ser utilizada para que haja compreensão dos engenheiros de software em relação à qualidade de uso na interação do usuário. Através do uso de modelos de interação é possível projetar e detectar eventuais problemas na interação usuário-sistema durante a etapa de projeto do sistema. Assim, os modelos de interação contribuíram para a qualidade da HCDP. Além da mitigação no retrabalho ao construir outros artefatos de maneira paralela.

#### **4.5. Ameaças à Validade dos Resultados deste Relato de Experiência**

Foram analisadas as seguintes ameaças que podem afetar a validade dos resultados deste relato de experiência [Wöhlin *et al.*, 2000], que são:

**Validade Interna** - *A expectativa do engenheiro de software responsável pelos diagramas MoLIC em relação aos benefícios da modelagem de interação:* Isto pode ter influenciado o projetista responsável pela construção dos diagramas MoLIC em obter resultados positivos na etapa de projeto. Porém, ressaltamos que nenhum dos projetistas da equipe possuíam relação com a autoria da MoLIC. Além disso, as lições aprendidas foram codificadas por um membro da equipe que não tinha participado da construção dos diagramas MoLIC. Portanto, os resultados das lições aprendidas abordam apenas a conclusão da equipe sobre cada Sprint.

**Validade Externa** - *Diagramas MoLIC como um artefato base para a construção dos demais artefatos de uma aplicação móvel:* Não é possível afirmar que os mesmos benefícios podem ser generalizados para diferentes tipos de aplicação, como Web ou desktop. É importante compreender o uso dos diagramas para diferentes tipos de aplicações.

**Validade de Conclusão** - *Limitação da amostra:* Este é o resultado de uma única equipe de projeto. Portanto, os resultados não são generalizáveis, mas são considerados válidos para o contexto do projeto.

### **5. Considerações Finais e Perspectivas Futuras**

Este artigo descreveu o relato de experiência de uma equipe de projeto com o uso de modelos de interação, através da MoLIC, para o desenvolvimento de uma aplicação móvel multiusuário. A modelagem de interação proporcionou compreensão para a equipe sobre alternativas de interação no sistema para os usuários. Isto deve-se a colaboração de todos os engenheiros de software da etapa de projeto no

desenvolvimento dos diagramas MoLIC. Posteriormente, os diagramas MoLIC foram utilizados como base para a construção de outros artefatos, como *mockups*, especificação de casos de uso e diagrama de atividades.

Portanto, sugere-se o uso de modelagem de interação no início da construção de outros artefatos para que haja compreensão dos engenheiros de software em relação à qualidade de uso do sistema. É importante ressaltar, que durante a Sprint 2 houve bastante retrabalho na correção de inconsistências dos artefatos, construídos a partir dos requisitos, diagrama de caso de uso e cenário. Tal retrabalho foi mitigado através da discussão do modelo de interação ao ser projetado pela equipe, que posteriormente foi utilizado como base para a construção dos demais artefatos na etapa de projeto das demais Sprints. Além disso, sugere-se a revisão dos diagramas MoLIC antes destes serem utilizados como base para a construção dos demais artefatos na etapa de projeto.

Assim, com os resultados deste relato de experiência, espera-se que a modelagem de interação seja amplamente adotada por engenheiros de software na melhoria da identificação e compreensão de todos os caminhos de interação do usuário com o sistema. Com tal compreensão, é possível desenvolver um sistema com foco na qualidade de uso e com pouco retrabalho na construção dos demais artefatos.

Como trabalhos futuros pretende-se analisar os artefatos da etapa de projeto que podem conter redundância na representação de informações, com o objetivo de entender os artefatos necessários para o desenvolvimento de software com qualidade. No entanto, ressalta-se que todos os artefatos elaborados na etapa de projeto da aplicação HC DP foram solicitados pela equipe responsável. Ainda como trabalhos futuros, pretende-se analisar a modelagem de interação com possíveis usuários finais para compreender todos os caminhos de interação usuário-sistema projetados.

### **Agradecimentos**

As autoras agradecem ao projeto de cooperação entre a Universidade Federal do Amazonas, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS) e a Samsung Research Brazil. Ao apoio financeiro concedido pelo “Large Scale Qualification PROgram on MOBILE Technologies” que é suportado pela Samsung Eletrônica da Amazônia Ltda, nos termos da Lei de Informática 8387/91; CAPES; e FAPEAM através de números de processos: 062.00600 / 2014; 062.00578 / 2014.

### **Referências**

- Barbosa, S. D. J. and Paula, M. G. (2003), “Designing and Evaluating Interaction as Conversation: a Modeling Language based on Semiotic Engineering”, In Interactive Systems. Design, Specification and Verification, 10th DSV-IS Workshop, p. 16–33.
- Beaudouin-Lafon, M. (2000), “Instrumental interaction: an interaction model for designing post-WIMP user interfaces”, In Proceedings of SIGCHI conference on Human factors in computing systems, p. 446 – 453.
- Castro, J. W., Acuña, S. T. and Juristo, N. (2008) “Enriching requirements analysis with the personas technique”, In Proceedings of the International Workshop on: Interplay between Usability Evaluation and Software Development (I-USED 2008), p. 13-18.

- Falbo, R. A., Machado, B. N. and Carvalho, V. A. (2008), “Uma Infra-estrutura para Apoiar a Elaboração Colaborativa de Artefatos de Software”, Em Anais do V Simpósio Brasileiro de Sistemas Colaborativos (SBSC 2008), p. 158-168.
- Lira, W. A. L., Alves, F. V. M., Santos Neto, P. A., Rabêlo, R. A. L., Britto and R. S. (2015), “Estimativa de Esforço em Projetos Ágeis de Software Utilizando Mapas de Kohonen”, Em XIV Simpósio Brasileiro de Qualidade de Software (SBQS 2015), p. 20-33.
- Lopes, A., Viana, D., Rabelo, J., Vieira, A. and Conte, T. (2013), “InspSoft 2.0: Um Jogo para o Ensino de Inspeção de Software”, Em XII Simpósio Brasileiro de Qualidade de Software (SBQS 2013).
- Lopes, A., Marques, A. B., Barbosa, S. D. J. and Conte, T. (2015), “Evaluating HCI Design with Interaction Modeling and Mockups: A Case Study”, In Proceedings of International Conference on Enterprise Information Systems, p. 79-87.
- Paula, M. G., Barbosa, S. D. J. and Lucena, C. J. P. (2003), “Relating Human-Computer Interaction and Software Engineering Concerns: Towards Extending UML Through an Interaction Modeling Language”, In Workshop proceedings: Closing the Gaps: Software Engineering and Human-Computer Interaction, p. 40-46.
- Rabelo, J., Viana, D., Santos, G. e Conte, T. (2014) “Usando PABC-Pattern para Codificar o Conhecimento: Um estudo Experimental”, Em Simpósio Brasileiro de Qualidade de Software - SBQS, 2014, Blumenau. Simpósio Brasileiro de Qualidade de Software, p. 1-15.
- Rivero, L., do Valle, R. e Conte, T. (2014) “Aplicando Design e Avaliação de Usabilidade para Melhorar a Qualidade de um Aplicativo Web Móvel”, Em Anais do Simpósio Brasileiro de Qualidade de Software (SBQS 2014), p. 260-267.
- Sangiorgi, U. B. and Barbosa, S. D. J. (2010) “Estendendo a linguagem MoLIC para o projeto conjunto de interação e interface”, In Proceedings of the IX Symposium on Human Factors in Computing Systems (IHC '10), p. 61-70.
- Souza, L. G. and Barbosa, S. D. J. (2014), “Estendendo a MoLIC para apoiar o design de sistemas colaborativos”, Em Anais do XIII Simpósio Brasileiro Sobre Fatores Humanos em Sistemas Computacionais (IHC'2014), p. 25-28.
- Scrum Alliance (2016). In: <http://www.scrumalliance.org/>, acessado em 15/06/2016.
- Travassos, G., Shull, F., Fredericks, M. and Basili, V. R. (1999), “Detecting defects in object-oriented designs: using reading techniques to increase software quality”, In Proceedings of the 14th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA '99), p. 47-56
- Wöhlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B. and Wesslén, A. (2000), “Experimentation in software engineering: an introduction”, Kluwer Academic Publishers, 1st Edition.