Integrando Engenharia de Software e Interação Humano-Computador em um Processo de Desenvolvimento de Software para/com Crianças

Jéssica C. B. Pereira, Amanda Meincke Melo

Universidade Federal do Pampa (Unipampa) – Campus Alegrete
Av. Tiarajú, 810 – Ibirapuitã – 97.546-550 – Alegrete – RS – Brasil

jcbp54@gmail.com, amanda.melo@unipampa.edu.br

Abstract. *Children are potential users of digital technologies as much as adults. Though there are several experiences described in the literature of children's involvement in software development to their use. However, these experiences do not show any Software Engineering approach. This work aims at proposing a software development process for/with children, integrating Software Engineering and Human-Computer Interaction methods, techniques and practices, to support researchers and software developers. This process, described in BPMN, can be refined and adopted as a reference for the development of software with children in new projects.*

Resumo. *Crianças são potenciais usuárias de tecnologias digitais tanto quanto os adultos. Embora haja várias experiências relatadas na literatura do envolvimento da criança no desenvolvimento de software para seu uso, essas não apresentam uma abordagem de Engenharia de Software. Este trabalho tem como objetivo, portanto, propor um processo de desenvolvimento de software para/com crianças, integrando métodos, técnicas e práticas de Engenharia de Software e Interação Humano-Computador, para auxiliar engenheiros de software e pesquisadores. Esse processo, descrito em BPMN, pode vir a ser aprimorado e adotado como referência para o desenvolvimento de software com crianças.*

1. Introdução

Crianças são usuárias de tecnologias digitais e, assim como os adultos, podem contribuir efetivamente para o seu desenvolvimento [Druin 2002]. Embora essas tecnologias adotem soluções em software, processos e métodos da Engenharia de Software (ES), diferentemente da Interação Humano-Computador (IHC), não consideram explicitamente o público infantil como possível parte interessada em seu processo de desenvolvimento.

Segundo Barbosa e Silva (2010), ES e IHC possuem enfoques diferentes no desenvolvimento de software, havendo alternativas à integração de ambas as áreas. Uma abordagem sistemática e disciplinada ao desenvolvimento de software para/com crianças, envolvendo a integração entre ES e IHC, tem o potencial de contribuir ao desenvolvimento de um sistema de software de qualidade ao mesmo tempo em que considera explicitamente as experiências, as necessidades e os anseios dos usuários envolvidos.

Propõe-se, neste artigo, apresentar um processo de desenvolvimento de software

para/com crianças, em BPMN (do inglês, *Business Process Model Notation*), organizado a partir do uso de técnicas como observação participante, reuniões, entrevistas e análise documental no contexto do projeto de pesquisa Design Inclusivo para/com Crianças e Adolescentes de Tecnologias Digitais de Informação e Comunicação (DICA-TDIC).

O texto está estruturado como segue. A Seção 2 apresenta o resultado de uma revisão de literatura realizada nas bases de dois eventos científicos nacionais, o Simpósio Brasileiro de Engenharia de Software (SBES) e o Simpósio sobre Fatores Humanos em Sistemas Computacionais (IHC), além de uma conferência internacional dedicada ao tema do design da interação e crianças: *Interaction Design and Children* (IDC). A Seção 3 descreve a metodologia adotada no trabalho. A Seção 4 sumariza os resultados obtidos. Finalmente, a Seção 5, apresenta algumas considerações finais e trabalhos futuros.

2. Revisão de Literatura

A revisão de literatura foi realizada, no período [2012, 2016], a partir de pesquisas nos anais de dois eventos nacionais de referência, um na área de Engenharia de Software – o SBES – e outro na área de Interação Humano-Computador – o IHC –, e de um evento internacional de referência para o design da interação e crianças – o IDC. No SBES, foram encontrados cinco trabalhos que abordam a integração entre ES e IHC e um que apresenta um estudo sobre gestão de processo. No IHC, foram encontrados três trabalhos que apresentam formas de integração entre IHC e ES. Já no IDC, foram encontrados três trabalhos que relatam o desenvolvimento de tecnologias para/com crianças.

Os trabalhos publicados no SBES têm considerado a integração entre ES e IHC em diferentes perspectivas. Contudo, não consideram o domínio de desenvolvimento de software para/com crianças. Propõem técnicas de inspeção de usabilidade para aplicações *web* [Fernandes *et al.* 2012] e *mobile* [Rivero *et al.* 2012], a avaliação de técnicas de avaliação [Rivero *et al.* 2014][Valentim *et al.* 2014], a investigação do impacto da adoção do Scrum na satisfação do cliente [Cartaxo *et al.* 2013], além de propor uma técnica para a criação de *personas* para apoiar no desenvolvimento de aplicações *web* para dispositivos móveis [Ferreira *et al.* 2015].

Trabalhos publicados no evento IHC observam requisitos como acessibilidade e usabilidade no *design* e na avaliação de tecnologias computacionais interativas [Borges *et al.* 2012][Braz *et al.* 2014][Boscarioli 2012]. Sua ênfase, entretanto, não está no envolvimento de crianças como parceiras de *design*, onde elas possam contribuir diretamente no desenvolvimento de um produto, expressando suas ideias, gostos e interesses.

No evento IDC, embora sejam reportadas várias experiências do envolvimento de crianças no desenvolvimento de tecnologias digitais, não foram identificadas experiências que adotem uma abordagem de ES, portanto, sistemática e disciplinada ao desenvolvimento de software [Pressman 2006]. Entre os trabalhos identificados, têm-se um estudo sobre a motivação de crianças para se envolverem com o Design Participativo [Hansen & Iversen 2013]; a apresentação de uma tecnologia desenvolvida com o envolvimento de crianças após a etapa de concepção e cujo desenvolvimento, conduzido de forma interdisciplinar, adotou o Design Centrado no Usuário [Gray *et al.*

2015] e a proposta de princípios de design multisensoriais para o Desenho Universal para crianças [Puccini *et al.* 2013].

3. Metodologia

Além da revisão de literatura, para se chegar ao modelo de processo proposto, em notação BPMN, adotaram-se técnicas como observação participante, reuniões, entrevistas e análise documental.

Realizou-se observação participante de dois estudos exploratórios envolvendo a produção de tecnologias digitais com crianças [Lôbo 2016] e em reuniões semanais do projeto de pesquisa DICA-TDIC. A observação participante é definida por Appolinário (2004, p. 144) como "Processo de observação no qual o pesquisador participa ativamente como membro do grupo que ele está estudando, utilizando esta posição privilegiada para obter informações acerca desse grupo".

A análise documental envolveu a verificação das agendas de trabalho de dois estudos exploratórios de Engenharia de Requisitos (ER) [Lôbo 2016], além da análise da descrição de dois processos de desenvolvimento de software para/com crianças no escopo do projeto de pesquisa DICA-TDIC [Ribeiro 2016][Souza 2016]. De acordo com Marconi e Lakatos (2009, p. 29), essa técnica consiste na representação condensada da informação do conteúdo de um documento.

Em complemento às reuniões semanais da equipe do projeto de pesquisa DICA-TDIC, foram realizadas reuniões com os colegas do grupo que produziam tecnologias com crianças, visando a conhecer seus processos de trabalho [Ribeiro 2016][Souza 2016]. Essas reuniões, no segundo semestre de 2016, colaboraram à dissolução de dúvidas, tomadas de decisões e à validação de modelos.

Entrevistas foram organizadas visando a orientar as reuniões para conhecer os processos de software utilizados pelos colegas do grupo DICA-TDIC. Segundo Barbosa e Silva (2010, p. 145), entrevista é "uma das técnicas mais utilizadas de coleta de dados e levantamento de requisitos. Tratando-se de uma conversa guiada por um roteiro de perguntas ou tópicos." A entrevista utilizada foi a não estruturada, que é uma entrevista flexível, onde são realizadas perguntas abertas para se aprofundar nos pontos desejados.

A abordagem adotada para o mapeamento e a modelagem do processo é a proposta por Tolfo (2013) *apud* Segui (2015), organizada em três etapas:

1. **Elaborar um mapa mental:** neste mapa são descritas as principais atividades do processo, seus atores e suas entradas e saídas;

2. **Elaborar fluxo de processo:** as atividades descritas no mapa mental são organizadas em ordem de ocorrência no processo;

3. **Modelar estado atual do processo:** adota-se a notação *Business Process Model Notation* para representar o processo em análise.

Essas etapas geram um ciclo, como ilustrado na Figura 1. Ao término deste ciclo, os responsáveis pelo modelo refinam a modelagem, analisam e discutem o fluxo e o mapa do processo.

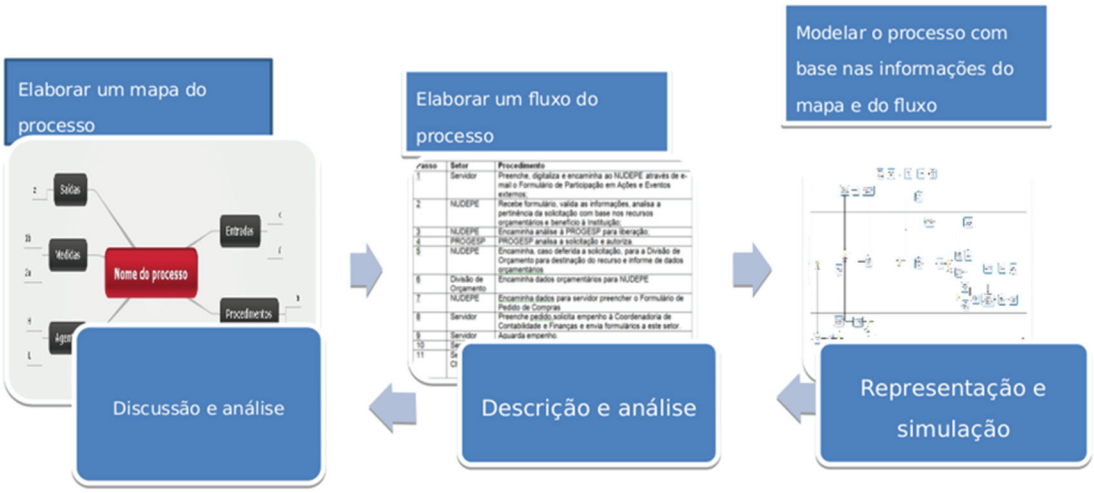

Figura 1. Abordagem para o mapeamento e a modelagem de um processo

Esse ciclo foi realizado para cada processo adotado no contexto do projeto de pesquisa DICA-TDIC, iniciando pelo processo de Engenharia de Requisitos aplicado em dois estudos exploratórios [Lôbo 2016]. Em seguida, foi realizado para apoiar a abstração de outros dois processos de desenvolvimento de *software* envolvendo crianças [Ribeiro 2016][Souza 2016].

4. Resultados

Como resultado, tem-se documentado, em notação BPMN, um processo de desenvolvimento de software que contempla a participação de crianças [Pereira 2016]. Seus níveis de abstração mais altos são apresentados nas Figuras 1, 2 e 3.

A Figura 2 evidencia as atividades realizadas pelas crianças – contidas nos subprocessos conduzidos pelos engenheiros de software. Na troca das mensagens são explicitadas as contribuições das crianças, o que inclui ideias, desenhos, histórias de usuários e sua priorização, *storyboards*, sugestões, identificação de problemas e impressões. Esses subprodutos, em níveis de abstração mais baixos, são modelados como artefatos [Pereira 2016]. Além disso, podem colaborar ativamente no desenvolvimento iterativo de protótipos e na codificação de software.

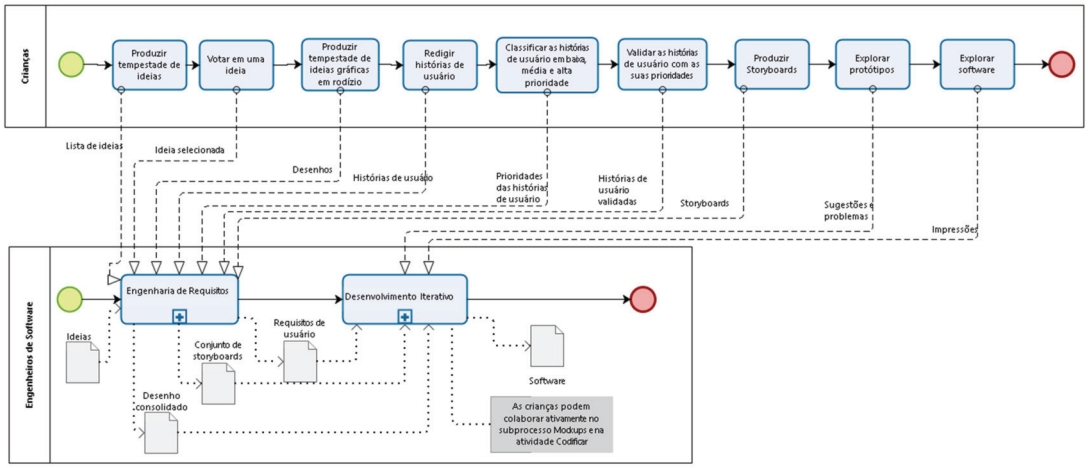

Figura 2. Modelo do processo Desenvolvimento de Software para/com crianças

Os subprocessos Engenharia de Requisitos e Desenvolvimento Iterativo são apresentados a seguir.

4.1 Engenharia de Requisitos

A Figura 3 representa o subprocesso de Engenharia de Requisitos em seu nível de abstração mais alto, com seu ciclo de atividades baseado em Sommerville (2007).

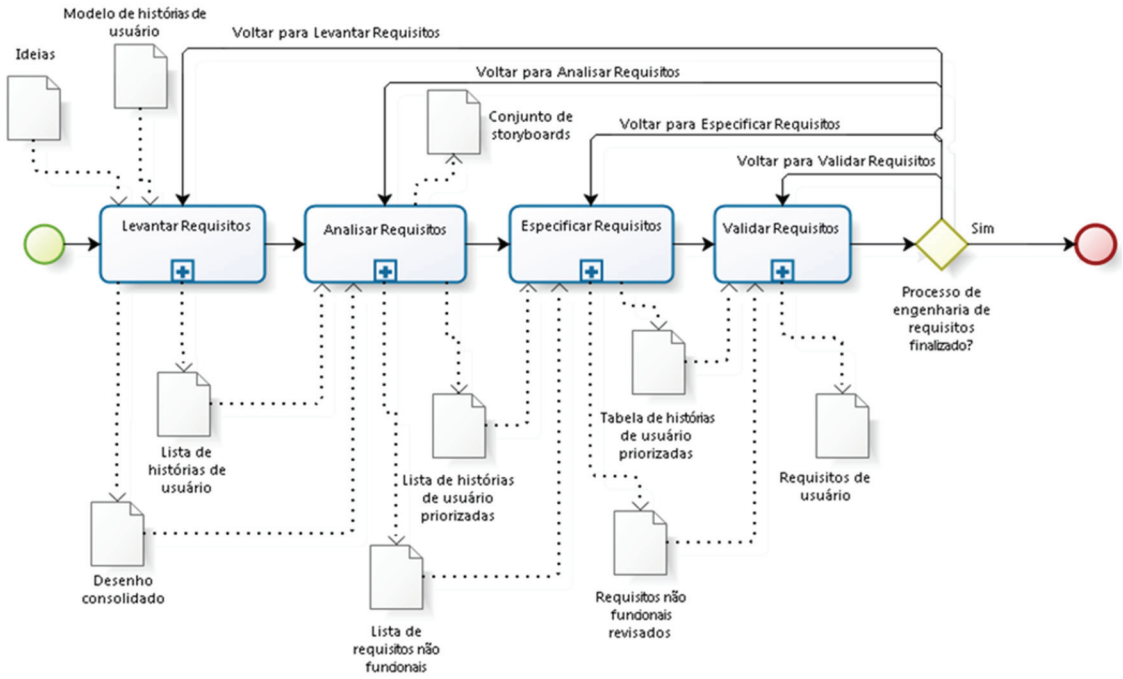

Figura 3. Modelo do subprocesso Engenharia de Requisitos

O subprocesso ER apresenta os subprocessos “Levantar Requisitos”, “Analisar Requisitos”, “Especificar Requisitos” e “Validar Requisitos”. Como entrada do subprocesso “Levantar Requisitos”, têm-se ideias para o desenvolvimento de uma tecnologia com as crianças e um modelo de histórias de usuário. Como saída, têm-se um desenho consolidado, que representa um esboço preliminar da interface de usuário, e uma lista de histórias de usuários, que representam os requisitos funcionais a serem desenvolvidos.

Esses artefatos são entradas para o subprocesso “Analisar Requisitos”. Como saídas do processo “Analisar Requisitos”, têm-se a lista de histórias de usuário priorizadas, a lista de requisitos não funcionais e um conjunto de *storyboards*, que descrevem algumas das interações do usuário com o sistema.

A lista de histórias de usuário priorizadas e a lista de requisitos não funcionais servem de entradas para o subprocesso “Especificar Requisitos”, que apresenta como saída uma tabela de histórias de usuário priorizadas e uma lista de requisitos não funcionais revisada.

A tabela de histórias de usuário priorizadas e a lista de requisitos não funcionais revisada são entradas do subprocesso “Validar Requisitos”, que gera como saída os requisitos de usuário. Antes de finalizar o processo, ocorre um evento de decisão, que determina se o subprocesso de Engenharia de Requisitos foi efetivamente finalizado ou

se ainda é necessário retornar a algum dos subprocessos de ER.

Durante o processo de Engenharia de Requisitos, as crianças participam ativamente dos subprocessos "Levantar Requisitos", "Analisar Requisitos" e "Validar Requisitos", gerando ideias, desenhos, histórias de usuário e *storyboards*. Assim, contribuem diretamente à identificação dos requisitos de usuário. Pereira (2016) detalha o subprocesso ER em mais dois níveis de abstração, explicitando a colaboração das crianças nas atividades evidenciadas na Figura 2, integradas aos subprocessos de ER.

4.2. Desenvolvimento Iterativo

A Figura 4 descreve as atividades do subprocesso Desenvolvimento Iterativo.

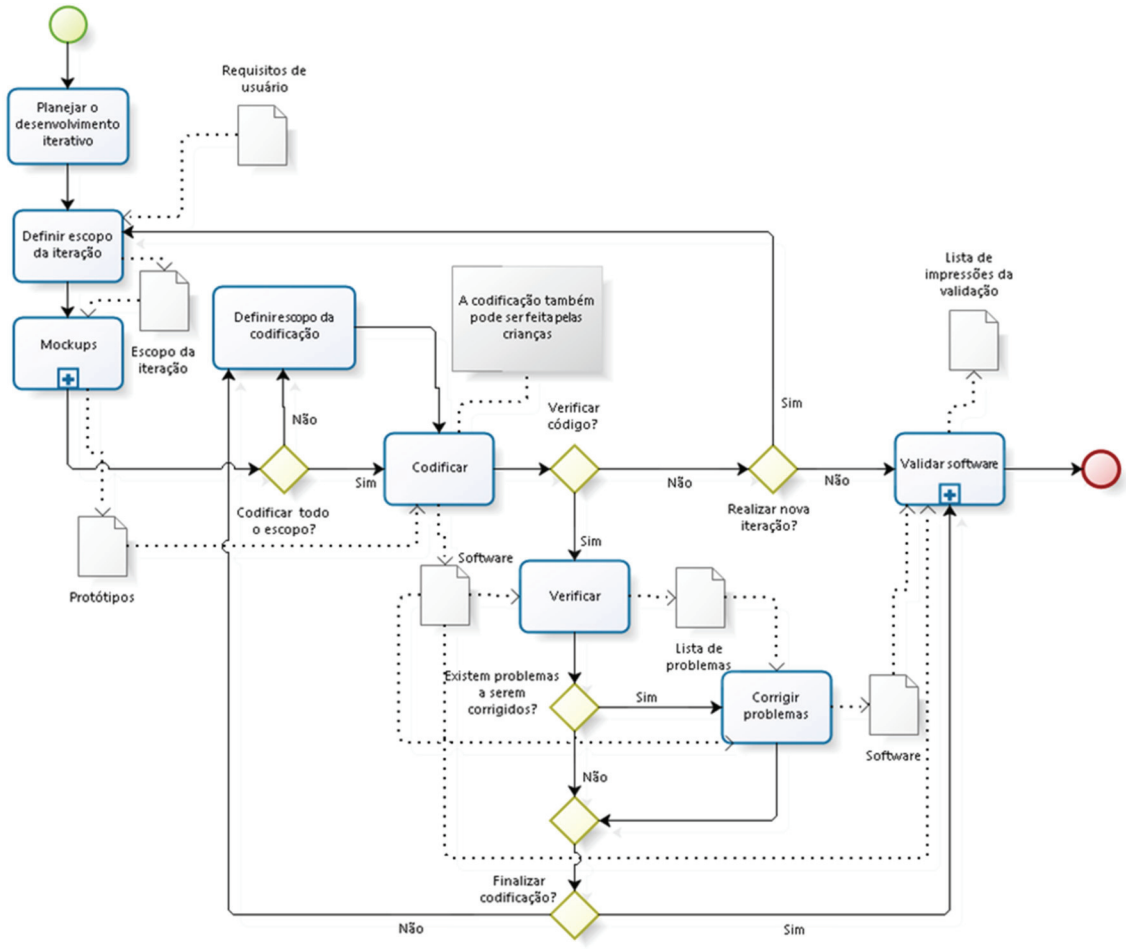

Figura 4. Modelo do subprocesso Desenvolvimento Iterativo

O subprocesso DI é iniciado com a atividade "Planejar o desenvolvimento iterativo", na qual ocorrem decisões como, por exemplo, se o desenvolvimento será iterativo e/ou incremental, métodos e técnicas de prototipação e de avaliação de interfaces de usuário a serem adotados, entre outras decisões relativas ao processo de desenvolvimento.

Na atividade "Definir escopo da iteração", que recebe os requisitos de usuário como entrada, define-se como parte do escopo todos os requisitos ou parte deles. Neste último caso, assume-se que o processo como um todo é iterativo. Sobre o escopo definido, no subprocesso "Mockups" [Muller *et al.* 1997], desenvolvem-se, com a

participação das crianças, protótipos da interface de usuário, em ciclos iterativos de prototipação e avaliação. Ao finalizar o subprocesso "Mockups", antes da atividade "Codificar", ocorre um evento de decisão para definir se todo o escopo será codificado como uma única atividade ou se a codificação será realizada de forma iterativa. A atividade "Codificar" gera software como saída. Essa atividade pode ser realizada pelas próprias crianças, por exemplo, com a adoção do ambiente Scratch [Ribeiro 2016]. O software produzido serve de insumo para as atividades "Verificar", "Corrigir problemas" e "Validar software". Uma vez que todo o software tenha sido codificado, prossegue-se ao subprocesso "Validar software". Os engenheiros de software, então, documentam as impressões da validação em uma lista que se torna saída do subprocesso.

Durante o processo DI, além de as crianças participarem do subprocesso "Validar software", podem atuar ativamente do subprocesso "Mockups" e da atividade "Codificar". Os subprocessos "Mockups" e "Validar software" são detalhados em Pereira (2016).

5. Conclusão

O modelo de processo proposto, em notação BPMN, provê apoio a engenheiros de software e pesquisadores para desenvolverem software para/com crianças, que contemplem a participação, os interesses, os gostos, as necessidades e as experiências desses usuários em uma perspectiva de ES. O modelo integra atividades de IHC a processos de Engenharia de Software, explicitando possibilidades de colaboração de crianças. Assim, dá oportunidade para que informem o que, como e porque querem.

Destaca-se como limitação do trabalho a ausência de documentação para o subprocesso Evolução de Software, que envolve acompanhar o software após sua entrega. Essa experiência, entretanto, não ocorreu no escopo do projeto de pesquisa DICA-TDIC.

Entre os trabalhos futuros, o modelo pode subsidiar o desenvolvimento de software com crianças em novos projetos. Além disso, pode ser revisado e aprimorado, por exemplo, contemplando o subprocesso Evolução.

6. Referências

Appolinário, F. (2004) Dicionário de Metodologia Científica, Atlas.

Barbosa, S. D. J., Silva, B. S. (2010), Interação Humano-Computador, Elsevier.

Borges, L. C. L. F., Filgueiras, L. V. L., Maciel, C., Pereira, V. C. (2012) Customizing a communication device for a child with cerebral palsy using participatory design practices: contributions towards the PD4CAT method. In: XI IHC, p. 57-66. SBC.

Boscarioli, C., Baqueta, J. J., Salles, C. G., Colling, J. P. (2012) Avaliação e design de interação de jogos voltados ao aprendizado de crianças surdas. In: XI IHC, p. 25-26. SBC.

Braz, P., Raposo, A., Souza, C. S. (2014) Uso de design probes no design de tecnologias para terapeutas de crianças com autismo. In: XIII IHC, p. 140-149. SBC.

Cartaxo, B., Araujo, A., Barreto, A. S. (2013) The Impact of Scrum on Customer Satisfaction: an Empirical Study. In: XXVII SBES, p. 129-136. SBC.

Druin, A. (2002) The role of children in the design of new technology. *Behaviour and information technology*, v. 21, n. 1, p. 1-25. Taylor & Francis.

Fernandes, P., Conte, T., Bonifácio, B. (2012) WE-QT: A Web Usability Inspection Technique to Support Novice Inspectors. In: XXVI SBES, p. 11-20. SBC.

Ferreira, B., Conte, T., Barbosa, S. D. J. (2015) Eliciting Requirements using Personas and Empathy Map to Enhance the User Experience. In: XXIX SBES, p. 80-89. SBC.

Gray, S., Robertson, J., Rajendran, G. (2015) BrainQuest: an active smart phone game to enhance executive function. In: XIV IDC, p. 59-68. ACM.

Hansen, E. I. K., Iversen, O. S. (2013) You are the real experts: studying teenagers' motivation in participatory design. In: XII IDC, p. 328-331. ACM.

Lôbo, K. L. S. (2016) Uma abordagem ética e responsável ao design para/com crianças: investigando a integração de práticas de interação humano-computador à engenharia de requisitos, Unipampa.

Marconi, M. A., Lakatos, E. M. (2009) Metodologia Científica, Atlas, 7ª. ed.

Muller, M. J., Haslwanter, J. H., Dayton, T. (1997) "Participatory Practices in the Software Lifecycle", Handbook of Human-Computer Interaction, Elsevier Science.

Pereira, C. J. B. (2016) Um Processo de Desenvolvimento de Software para/com Crianças: integração entre Engenharia de Software e Interação Humano-Computador, Unipampa.

Pressaman, R. S. (2006) Engenharia de Software, Mc Graw-Hill, 6ª. ed.

Puccini, A. M., Puccini, M., Chang, A. (2013) Acquiring educational access for neurodiverse learners through multisensory design principles. In: XII IDC, p. 455-458. ACM.

Ribeiro, S. S. (2016) Um método para o desenvolvimento de software por crianças com o ambiente Scratch, Unipampa.

Rivero, L., Conte, T. (2012) Using an Empirical Study to Evaluate the Feasibility of a New Usability Inspection Technique for Paper Based Prototypes of Web Applications. In: XXVI SBES, p. 81-90. SBC.

Rivero, L., Kawakami, G., Conte, T. (2014) Using a Controlled Experiment to Evaluate Usability Inspection Technologies for Improving the Quality of Mobile Web Applications Earlier in their Design. In: XXVIII SBES, p. 161-170. SBC.

Tolfo, C. (2013) Metodologia de mapeamento e modelagem de processos. p. 20-55 *apud* Segui, B. S. (2015) Desenvolvimento de sistema para solicitação de registro de marca de gado, Unipampa.

Sommerville, I. (2007) Engenharia de Software. Pearson Education, 8ª. ed.

Souza, A. T. (2016) Requisitos de segurança e privacidade no desenvolvimento de tecnologia web para/com crianças: integrando técnicas e práticas de Interação Humano-Computador e Engenharia de Software, Unipampa.

Valentim, N. M. C., Conte, T. (2014) Improving a usability inspection technique based on quantitative and qualitative analysis. In: XXVIII SBES, p. 171-180. SBC.