

Aplicação de Técnicas do SPIDe à Engenharia de Requisitos: uma revisão sistemática

Applying SPIDe's Techniques in Requirements Engineering: a systematic review

Jean C. S. Rosa¹, Fiama S. Silva², Gilton J. F. Silva², Ecivaldo Matos¹

¹Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)
Av. Adhemar de Barros, s/n – Salvador – BA – Brasil

²Departamento de Computação – Universidade Federal de Sergipe (UFS)
Av. Marechal Rondon, s/n – São Cristóvão – SE – Brasil

jean.rosa@ufba.br, {fssantos, gilton}@dcomp.ufs.br, ecivaldo@ufba.br

Resumo. Entender os desejos e necessidades dos (potenciais) usuários são desafios da Engenharia de Requisitos e do Design de Interação. Ambos possuem ferramentas para auxiliar o engenheiro de requisitos e o designer de interação identificar e especificar os desejos e necessidades, uma delas é o SPIDe (Semio-Participatory Interaction Design). Todavia, ainda não se sabe se o SPIDe possibilita a engenharia requisitos de software. Nesse sentido, este artigo apresenta resultados de uma revisão sistemática da literatura sobre o uso das técnicas que compõem o SPIDe aplicadas à Engenharia de Requisitos. Os resultados apontam que duas técnicas que compõem o SPIDe já foram utilizadas na Engenharia de Requisitos.

Abstract. Understand (potential) user's desires and needs are challenges for Requirements Engineering and Interaction Design. Both, Requirements Engineering and Interaction Design, have tools to help the requirements engineer and interaction designer to identify and specify desires and needs, like the SPIDe (Semio-Participatory Interaction Design). However, it is not yet known whether SPIDe enables engineering software requirements. In this sense, this paper presents results of a systematic review on the use of SPIDe techniques applied to Requirements Engineering. The results show that two techniques that make up the SPIDe have been used in Requirements Engineering.

1. Introdução

Software educacional é um artefato computacional que deve atuar como mediador em atividades de formação e desenvolvimento de pessoas em áreas distintas do conhecimento [Gomes e Wanderley 2003]. Todavia, é importante destacar que os softwares educacionais não garantem a aprendizagem [Matos 2013]. Para que isto ocorra, é necessário que educadores e estudantes apropriem-se dessa tecnologia [Rosa 2016].

A utilização desses artefatos computacionais tem favorecido a construção do conhecimento; todavia, é necessário atentar-se para a sua qualidade, isto para que os

seus benefícios sejam adequados [Junior e Paola 2014]. A qualidade dos softwares educacionais tem sido estudada por pesquisadores da área de Interação Humano-Computador (IHC) e de Informática na Educação, dado que a qualidade da interação pode influenciar os processos de ensino e de aprendizagem mediatizados pelos artefatos computacionais [Matos 2013; Rosa, Schwarzelmüller e Matos 2015].

A qualidade de softwares educacionais está relacionada à capacidade de ele atender os desejos e as necessidades dos seus usuários com relação ao ensino e a aprendizagem, sendo professores(as) e estudantes os usuários mais característicos desse tipo de softwares [Gomes e Padovani 2005].

Kujala *et al.* (2001) aborda que além dos desejos e da necessidade, o contexto de uso também deve ser considerado, de modo que a análise dos desejos, das necessidades e do contexto de uso devem produzir requisitos a serem considerados na construção de softwares educacionais. De acordo com Heiskari *et al.* (2009), a obtenção e análise de requisitos podem ser as partes mais importantes e mais difíceis na concepção de um software.

Capturar com precisão os requisitos do sistema é importante para o bom desenvolvimento de um produto de software e para o resultado geral de um projeto. Falhas na identificação, definição e validação dos requisitos podem fazer com que todo o ciclo de vida de um projeto seja comprometido [Sayão, Staa e Leite 2003]. A Engenharia de Requisitos (ER) pode ser conceituada como o amplo espectro de tarefas e técnicas que levam a um entendimento dos requisitos na Engenharia de Software (ES) [Pressman e Maxim 2016].

Por meio da ER é possível entender o que o usuário deseja, analisando suas necessidades, negociando uma solução para o problema, especificando e gerindo os requisitos durante o prosseguimento do projeto. Sem a ER há probabilidade de a solução proposta não satisfazer as necessidades dos usuários. Sendo assim, a ER também tem interesse na qualidade que a solução computacional proporcionará ao seu usuário [Pressman e Maxim 2016].

Outro processo que também busca aumentar a qualidade de soluções computacionais é o Design de Interação (DI). O DI está relacionado com a qualidade de uso de soluções computacionais. O objetivo do DI é dar formas às coisas digitais para o uso das pessoas [Lowgren 2014]. A ER e o DI possuem métodos, técnicas, modelos e ferramentas sistemáticas próprias para a construção de uma solução computacional. Uma das ferramentas para o DI é o SPIDe, proposto por Rosa e Matos (2016b) e expandido por Pita *et al.* (2017).

O SPIDe é um processo semioparticipativo de DI. Os estudos sobre o SPIDe [Rosa e Matos 2016; Pita *et al.* 2017] tem tratado especificamente sobre requisitos ligados diretamente a Interação Humano-Computador (IHC), deixando de lado outros tipos de requisitos relacionados à Engenharia de Software (ES).

Nesse sentido, este artigo apresenta uma revisão sistemática da literatura (RSL) sobre o uso das técnicas participativas que compõem o SPIDe aplicadas à Engenharia de Software. Com os resultados da RSL foi possível (i) identificar se as técnicas que compõem do SPIDe podem ser utilizadas junto à ER; (ii) em qual etapa da ER as

técnicas podem ser utilizadas; além de (iii) fundamentar a hipótese de que com o SPIDe é possível elicitare requisitos de interação (humano-computador) e de software de modo integrado.

Este artigo está dividido em seis seções. A seção seguinte trata sobre pesquisas que abordam sobre a integração entre IHC e ES. A Seção 3 trata sobre o processo semioparticipativo para o design de interação, o SPIDe. Por sua vez, a metodologia aplicada na pesquisa é apresentada na Seção 4. As seções 5 e 6 tratam sobre os resultados e as considerações finais, respectivamente.

2. Integração ES & IHC

O processo de desenvolvimento de software e o seu estudo está relacionado à Engenharia de Software (ES) [Sommerville 2007]. Por sua vez, a concepção, desenvolvimento, avaliação, o uso e os principais fenômenos ligados a eles é tema de estudo da disciplina denominada Interação Humano-Computador (IHC) [ACM 1992].

Apesar de parecerem semelhantes em seus objetivos, a ES e a IHC se desenvolveram teórica e metodologicamente disjuntas. De acordo com Barbosa e Silva (2010), a qualidade de uso foi preocupação da ES na década de 1970. Todavia, a ES direcionou seus interesses para a construção, implementação e manutenção de softwares. Contudo, Seffah, Desmerais e Metzker (2005) apontam que a ES e a IHC são áreas complementares, mas não bem integradas. Ferre e Moreno (2004) indicam que a essência interdisciplinar da IHC é o maior obstáculo para a sua integração com a ES e por isso ambas se aproximam da construção de software de diferentes perspectivas. Mesmo assim, pesquisas¹ sobre a integração entre a ES e a IHC vem sendo desenvolvidas ao longo dos anos.

Ferre e Moreno (2004), por exemplo, apresentam algumas técnicas de IHC que podem ser utilizadas durante a ES e melhorar tanto a qualidade do software, quanto a qualidade de interação. Por sua vez, Silva *et al.* (2004) apresentam padrões de ES e IHC que podem auxiliar equipes multidisciplinares de desenvolvimento de software. Memmel, Gundelsweiler e Reiterer (2007) apresentam um modelo ágil interdisciplinar para a integrar IHC à metodologia ágil *eXtreme Programming* (XP), o CRUISER. Segundo os pesquisadores, o CRUISER é uma tentativa de fazer uma ponte entre a ES e a IHC por meio das características comuns entre as áreas.

Apesar de várias pesquisas tratarem sobre a integração entre a ES e a IHC, Ogunyemi e Lamas (2014) apresentam várias lacunas nessas tentativas. A diferença entre os objetivos, o envolvimento do usuário, os fatores humanos, a terminologia diferente são exemplo de lacunas que ainda permeiam a integração entre a ES e a IHC e que devem ser estudadas para a construção de um software de qualidade de software e de interação.

A pesquisa apresentada neste artigo busca contribuir para preencher as lacunas sobre o envolvimento do usuário e fatores humanos no DI e na ES por meio da aplicação do SPIDe para a elicitare requisitos de interação (humano-computador) e

¹ [Ferre e Moreno 2004; Silva *et al.* 2004; Seffah, Desmerais e Metzker 2005; Memmel, Gundelsweiler e Reiterer 2007].

de software de modo integrado no contexto de concepção de softwares educacionais. Nesse sentido, o SPIDe é o tema da próxima seção.

3. SPIDe

O SPIDe² é um processo semi-participativo para o design de interação de artefatos computacionais [Rosa 2016]. A construção do SPIDe partiu do princípio que “*quem pode dizer quais critérios são mais importantes para sua satisfação é o próprio usuário*” [Matos 2013 p. 92]. Nesse sentido, sob a perspectiva de design de interação semi-participativo o SPIDe foi construído sob a base teórica da Engenharia Semiótica (EngSem) e dos princípios e técnicas do Design Participativo (DP) [Rosa e Matos, 2016b].

A EngSem é uma teoria de IHC de base semiótica que compreende o processo de interação humano-computador como um processo comunicativo entre o designer de interação e o usuário mediado pela interface computacional [De Souza 2005]. Por sua vez, o DP é uma abordagem que valoriza a participação de potenciais usuários durante toda a concepção de uma tecnologia, onde para além de ser consultado, ele pode auxiliar na construção de soluções atuando em colaboração com o desenvolvedor [Luck 2003]. O design de interação é um processo social que deve incluir não só o designer, mas também o usuário [Baranauskas 2013], por isso, além da visão teórico-conceitual da EngSem, o SPIDe inclui técnicas do DP [Rosa e Matos 2016b].

O SPIDe implementa o Design Centrado na Comunicação (DCC), uma prática da EngSem que considera o design de interação como um processo de manipulação de mensagens [Barbosa, Paula e Lucena 2004]. O DCC divide o processo de design de interação em três etapas: análise de contexto, engenharia de interface e avaliação. Cada uma dessas etapas é composta por técnicas do DP e apesar de serem apresentadas sequencialmente, conforme a Figura 1, caso seja necessário, é possível retornar a técnicas ou etapas anteriores [Rosa e Matos 2016b].

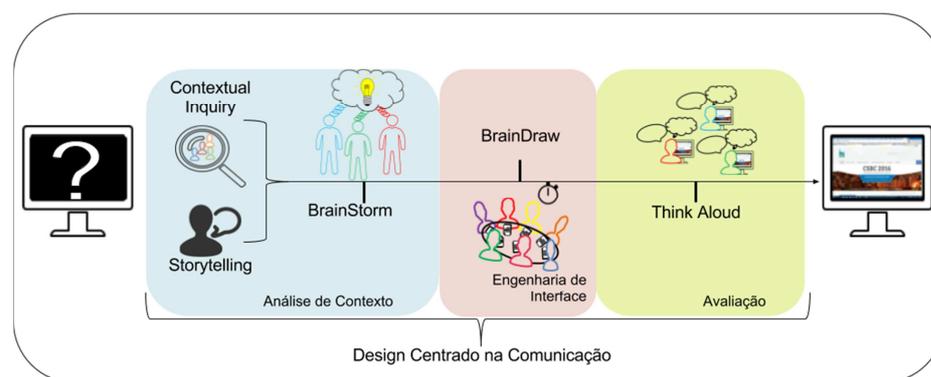


Figura 1. Composição do framework SPIDe (Pita et al., 2017)

A análise de contexto tem como objetivo conhecer o usuário, o seu contexto e como ele soluciona o problema. Essa etapa é composta por três técnicas do DP, o *contextual inquiry*, o *storytelling* e o *brainstorm* [Pita et al. 2017]. De acordo com Rosa e Matos (2016b), os resultados dessa etapa apresentam diretrizes sobre tecnologias

² [Rosa 2016; Rosa e Matos 2016a; Rosa e Matos 2016b; Pita et al. 2017].

disponíveis ao usuário, o modo como ele resolve o problema, como evitar problemas de interação e requisitos de interação.

A etapa de engenharia de interface tem por objetivo a criação de protótipos da tecnologia digital a ser (re)construída a partir da análise de contexto. O *braindraw* é a técnica do DP que compõe essa etapa e resulta em desenhos com a fusão de ideias dos seus participantes que posteriormente deve basear a construção de protótipos. Por fim, a etapa de avaliação tem o propósito de avaliar os protótipos construídos pela etapa de engenharia de interface. Essa etapa é composta pela técnica *think-aloud*. Durante a aplicação dessa técnica os usuários devem verbalizar os seus pensamentos ao interagir com o protótipo criado [Rosa e Matos 2016b].

A combinação da EngSem com o DP proposta pelo SPIDe transforma o paradigma inicial de interação humano-computador da EngSem, pois os usuários são codesigners de interação, ou seja, o usuário assume o papel de designer [Rosa e Matos 2016b].

A primeira versão do SPIDe³ foi composta por quatro técnicas do DP (*contextual inquiry*, *brainstorm*, *braindraw* e *think-aloud*) e foi aplicada no domínio educacional, com estudantes de ensino fundamental e professores para o redesign de interação da rede socioeducacional TecCiência⁴. A proposta de concepção do SPIDe é considerar o design por todos e para todos. Nessa proposta inicial Rosa e Matos (2016b) se dedicaram a aspectos culturais no design de interação. Para os pesquisadores, o SPIDe pode considerar esses aspectos devido a sua característica semioparticipativa.

A pesquisa desenvolvida por Pita *et al.* (2017) continuou o desenvolvimento do SPIDe. Aplicou-se a análise de contexto do SPIDe no domínio de mobilidade urbana com usuário com deficiência visual também a partir da perspectiva de design para todos e por todos. A pesquisa resultou na primeira modificação do SPIDe. Pita *et al.* (2017) sugeriram a adição da técnica *storytelling* devido usuários cegos ou deficientes visuais (público alvo dos pesquisadores) sentirem-se constrangidos pela aplicação do *contextual inquiry*. Desse modo, é possível escolher entre a aplicação do *contextual inquiry* ou do *storytelling* durante o (re)design de interação de uma tecnologia digital interativa sob a perspectiva de uso/design para/por todos.

3.1 Integração SPIDe & ER

Segundo Pressman e Maxim (2016), a ER é uma ação da Engenharia de Software, que estabelece uma base sólida para o projeto e construção de um software, fornecendo meios para entender as necessidades do usuário. A ER é composta por quatro fases: (i) elicitação, (ii) análise e negociação, (iii) documentação, e (iv) verificação e validação [Kontoya e Sommerville 1998].

Durante a fase de elicitação de requisitos busca-se obter o máximo de informações importantes e relevantes de um determinado produto para descobrir seus requisitos [Kontoya e Sommerville 1998]. Tendo em vista que o objetivo da fase de elicitação assemelha-se com o propósito das etapas de análise de contexto e engenharia

³ cf. [Rosa 2016; Rosa e Matos 2016b]

⁴ www.tecciencia.ufba.br

de interface do SPIDE, foi proposto que essas etapas do SPIDE possam contribuir com essa fase da ER.

A fase de verificação e validação dos requisitos deve ser realizada com o objetivo de verificar possíveis omissões, conflitos e ambiguidades nos requisitos descritos, validando-os [Kontoya e Sommerville 1998]. O objetivo dessa fase é análogo ao objetivo da etapa de avaliação do SPIDE, visto que ambos buscam identificar se o que foi criado por meio das fases/etapas anteriores está conforme as necessidades e desejos dos usuários.

A partir dessas relações foi possível criar a seguinte hipótese: com o SPIDE é possível elicitar requisitos de interação (humano-computador) e de software de modo integrado. Para embasar essa hipótese, foi realizada uma revisão sistemática literatura (RSL), com a metodologia e os resultados apresentados nas próximas seções.

4. Metodologia

De acordo com Rosa e Matos (2016b) e Pita *et al.* (2017), a aplicação do SPIDE pode gerar requisitos para o (re)design de interação considerando: aspectos culturais, a participação de crianças, adultos e deficientes visuais nos domínios educacional e mobilidade urbana. A pesquisa apresentada neste artigo tem como objetivo expandir o SPIDE investigando o uso das técnicas participativas que o compõem aplicadas à ER.

Para atingir esse objetivo foi realizada uma revisão sistemática literatura (RSL). A RSL deste trabalho explorou produções científicas que apresentam resultados da utilização das técnicas contidas no SPIDE contribuindo com a geração de requisitos ou auxiliando as etapas do processo de engenharia de requisitos. Para guiar a revisão sistemática foi utilizado o procedimento proposto por Kitchenham (2004).

Espera-se com a RSL, baseada em produções científicas anteriores, responder as seguintes questões de pesquisa:

- (1) As técnicas do SPIDE têm sido utilizadas em conjunto com as atividades do processo de Engenharia de Requisitos?
- (2) Em qual etapa do processo da engenharia de requisitos as técnicas do SPIDE têm sido utilizadas?
- (3) Quais tipos de requisitos são gerados por meio da utilização das técnicas do SPIDE quando aplicadas à Engenharia de Requisitos?

Para responder essas perguntas foram buscadas produções científicas em sete bibliotecas científicas digitais, apresentadas na Figura 2. A *string*⁵ de busca utilizada foi definida com base nas técnicas contidas no framework SPIDE associadas à Engenharia de Requisitos.

As execuções da busca nas bibliotecas digitais resultaram em uma amostra inicial de 370 produções contendo 22 em duplicidade, restando 348 produções, no qual foi necessário realizar a leitura dos títulos, resumos e palavras-chave com o objetivo de

⁵ (“engenharia de requisitos” OR “requirement engineering”) AND (“contextual inquiry” OR “storytelling” OR “braindraw” OR “braindrawing” OR “brainstorm” OR “brainstorming” OR “think-aloud” OR “think aloud”)

filtrar os trabalhos que tivessem relação com o tema em questão, considerando os critérios de inclusão e exclusão definidos no planejamento.

Após a realização dessa etapa de seleção, 326 trabalhos foram rejeitados e restaram 22 como aceitos. Os trabalhos coletados na etapa de seleção foram lidos de forma integral com o objetivo de coletar somente trabalhos relevantes para realizar a condução da pesquisa. Através da aplicação desta etapa de extração foram aceitos 10 trabalhos e 12 trabalhos foram excluídos com base nos critérios de inclusão e exclusão definidos. O diagrama da Figura 2 a demonstra de forma simplificada o processo de Revisão Sistemática, os resultados iniciais e finais. O gráfico na Figura 3 apresenta o resultado da aplicação dos critérios de exclusão após a retirada dos trabalhos duplicados.

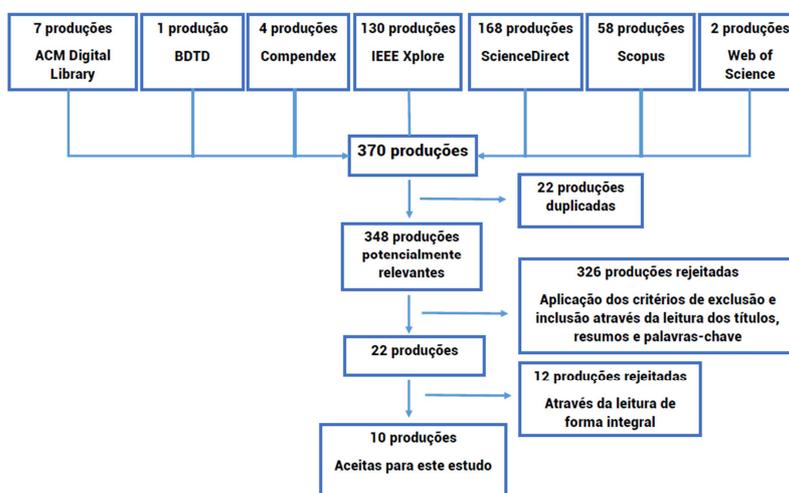


Figura 2. Detalhamento da Revisão Sistemática da Literatura

Foram excluídas produções com menos de três páginas (*short papers*), relatórios técnicos e livros (*white papers*), estudos secundários, produções que não apresentam resultados da utilização do *contextual inquiry*, *storytelling*, *braindraw*, *brainstorm* ou *think-aloud* relacionados ao processo de engenharia de requisitos de software e que não estejam escritos em português ou inglês.

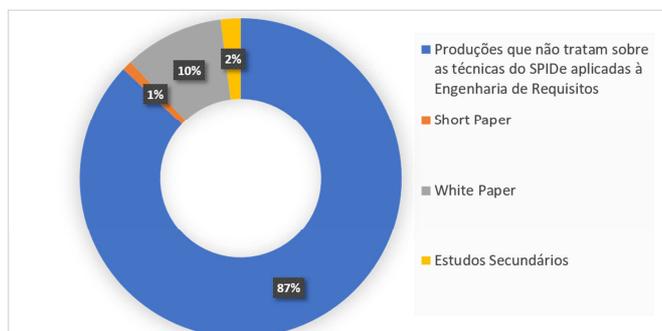


Figura 3. Motivação da exclusão dos artigos

Os 10 trabalhos resultantes da busca nas bibliotecas digitais e da aplicação dos critérios de inclusão e exclusão formam o *corpus* de produções científicas para responder as questões da revisão sistemática.

5. Resultados

A revisão sistemática que resultou em um *corpus* de 10 produções científicas, apresentados no Quadro 1 por ordem decrescente de publicação. As produções científicas resultantes foram publicadas entre os anos de 2009 e 2015. Em sua maioria, elas apresentam estudos sobre a utilização de técnicas específicas junto ao processo de ER de software. Três delas são brasileiras, duas vinculadas a Universidade Federal de Pernambuco [Batista e Silva 2015; Batista 2014] e uma vinculada à Universidade Federal do Rio de Janeiro [Laporti, Borges e Braganholo 2009].

Quadro 1. Produções científicas encontradas

Índice	Título	Ano
P1	<i>Selecting Creativity Techniques for Creative Requirements: An Evaluation of Four Techniques using Creativity Workshops</i>	2015
P2	<i>A creative process to elicit contexts for context sensitive systems</i>	2015
P3	<i>Assessment and Evaluation of Requirements Elicitation Techniques Using Analysis Determination Requirements Framework</i>	2014
P4	<i>Collaborative Method for Business Process Oriented Requirements Acquisition and Refining</i>	2014
P5	Um processo criativo de descoberta de contextos para sistemas sensíveis a contexto	2014
P6	<i>Requirements Engineering Meets Physiotherapy: An Experience with Motion-Based Games</i>	2013
P7	<i>Using Storytelling to Record Requirements: Elements for an Effective Requirements Elicitation Approach</i>	2011
P8	<i>Validation of the Effectiveness of an Optimized EPMcreate as an Aid for Creative Requirements Elicitation</i>	2010
P9	<i>Structured Digital Storytelling for Eliciting Software Requirements in the ICT4D Domain</i>	2010
P10	<i>Athena: A collaborative approach to requirements elicitation</i>	2009

Segundo Lai, Peng e Ni (2014), um dos principais problemas ocorridos durante a ER está relacionado diretamente com a falta de comunicação adequada entre os usuários e engenheiros de requisitos. Com isso, os autores propuseram a utilização de um método desenvolvido por eles, chamado *Business Process Oriented Collaborative Requirements* (BPCRAR). O método proposto é baseado na teoria da narrativa e na teoria argumentativa, utilizando o *group storytelling* para identificar as necessidades dos usuários, o *Narrative Network Model* para identificar os Casos de Uso, a *Business Process Model and Notation* (BPMN) para descrever os requisitos formalmente e o *dialogue game* para refinar os requisitos.

O *group storytelling* é uma variação da técnica *storytelling*, onde as construções das histórias narradas pelos participantes são compartilhadas com o grupo de envolvidos, permitindo sua complementação e o compartilhamento dos conhecimentos (Lai, Peng e Ni, 2014). Laporti, Borges e Braganholo (2009), Batista e Silva (2015) e Batista (2014) utilizaram em suas pesquisas o *group storytelling*.

Laporti, Borges e Braganholo (2009) fazem uso do *group storytelling*, todavia a partir da composição do método Athena. De acordo com os pesquisadores, este método possui uma abordagem colaborativa para a elicitación de requisitos baseada no *group storytelling*, onde os usuários contam histórias sobre as tecnologias que suportam as suas atividades. As histórias contadas são resumidas, depois convertidas em cenários e após isso transformadas em casos de uso. Para demonstrar a efetividade do método proposto, os autores realizaram uma avaliação experimental com dois grupos, onde o grupo que utilizou o método proposto conseguiu obter casos de uso mais detalhados e precisos.

De acordo com Batista e Silva (2015) e Batista (2014), os requisitos sofrem uma influência cada vez maior do contexto em que os sistemas serão utilizados. Na busca por sistemas que sejam adaptáveis às necessidades dos usuários e às mudanças no contexto operacional foi proposto a utilização de um *framework* durante a etapa de elicitación e especificação de requisitos com o intuito de apoiar os engenheiros de software no desenvolvimento de sistemas sensíveis ao contexto. O *framework* intitulado como BVCCon apresenta um processo guiado por técnicas criativas como o *group storytelling*, mapas mentais e a utilização da ferramenta 5W1H + condicional. O processo proposto foi avaliado por meio de estudo de casos, que indicou a viabilidade do processo, considerando o grau de satisfação da equipe de desenvolvimento.

Sinnig *et al.* (2010) citam a importância da ER para o desenvolvimento de projetos da tecnologia da informação e da comunicação para o desenvolvimento (ICT4D). A abordagem ICT4D tem sido utilizada para aplicar as Tecnologias da informação e comunicação (TIC) na correção das desigualdades socioeconômicas e promover o desenvolvimento dessas comunidades. Para apoiar a condução do projeto, os autores selecionaram a técnica *storytelling*. Com a sua investigação foi possível identificar que a utilização do *storytelling* se mostrou útil para realizar o levantamento de informações junto a pessoas que possuem pouca capacidade para ler e escrever, permitindo com que elas se expressem com mais facilidade. A utilização da técnica junto a abordagem ICT4D se mostra mais vantajosa do que outras técnicas de elicitación de requisitos comumente utilizadas, como entrevistas, grupos focais e estudos etnográficos.

No estudo realizado por Boulila, Hoffmann e Herrmann (2011) demonstrou-se que o *storytelling* é uma ferramenta poderosa para auxiliar na transferência e na captação do conhecimento, onde os usuários podem usar uma forma mais natural de transmitir suas necessidades e solicitações. O estudo dos pesquisadores comprovou, por meio de um estudo de caso, que o *storytelling* demonstrou ser mais eficaz que a técnica *brainstorm*.

Besrou, Bin e Dominic (2014) também buscaram avaliar algumas das técnicas mais utilizadas no processo de elicitación de requisitos. As técnicas escolhidas para a

análise foram: entrevista, *brainstorm* e *Joint Application Development* (JAD). As avaliações das técnicas foram feitas com o objetivo de identificar a estrutura e conteúdo sintático da técnica, semântica, funcionamento e nível de decomposição, capacidade de comunicação da técnica de acordo com a legibilidade e compreensão do documento produzido e a utilidade da técnica através de uma resposta direta do usuário. Ao fim do experimento foi obtido um resultado estatístico indicando que a técnica de entrevista obteve uma pontuação mais alta seguido por JAD e *brainstorm*.

Os autores Svensson e Taghavianfar (2015) e Sakhnini, Berry e Mich (2010) discutiram a importância da utilização de técnicas que estimulam a criatividade junto ao processo de ER, durante a etapa de elicitação dos requisitos. Svensson e Taghavianfar (2015) realizaram workshops de criatividade para avaliar as técnicas *hall of fame*, *constraint removal*, *brainstorm* e *idea box*. O resultado desses workshops mostrou que a aplicação do *brainstorm* gerou o maior número de requisitos criativos, seguido pelas técnicas de *constraint removal*, *hall of fame* e *idea box*.

Por sua vez, pesquisa conduzida por Sakhnini, Berry e Mich (2010) buscou investigar a utilização da EPMcreate e sua variante o POEPMcreate, baseadas no Modelo Elementar Pragmático (EPM), em comparação com a técnica de *brainstorm*. O objetivo foi identificar qual das técnicas se mostra mais eficaz durante a etapa de elicitação de requisitos. Ao fim do experimento foi possível constatar que POEPMcreate é mais eficaz para ajudar a gerar novos requisitos do que EPMcreate, que por sua vez é mais eficaz em ajudar a gerar novos requisitos do que o *brainstorm*.

Com o objetivo de elicitar e modelar requisitos de jogos baseados em movimento para pacientes em tratamento, os autores Pasquale *et al.* (2013) propõem a utilização da metodologia RE-FIT. A metodologia é dividida em 3 etapas, onde são aplicadas as técnicas *brainstorm*, pesquisas *ad-hoc* durante a etapa de elicitação de requisitos e a técnica *FLAGS goal model* para realizar a modelagem dos mesmos. Para identificar a validade da metodologia proposta foi realizado um estudo de caso em um hospital e a conclusão obtida é que a utilização das técnicas *brainstorm*, pesquisas *ad-hoc* e a observação direta proporcionou um melhor entendimento das necessidades dos usuários sem a necessidade de realizar muitas interações e pôde auxiliar os engenheiros de software a produzir modelos de requisitos que melhor se adaptem às necessidades da equipe médica e dos pacientes.

Respondendo a primeira questão norteadora da revisão sistemática, somente duas técnicas do SPIDE (a saber *brainstorm* e *storytelling*) têm sido utilizadas por pesquisadores em atividades do processo de ER - o Quadro 2 apresenta a relação entre as técnicas e as produções científicas. As duas técnicas foram utilizadas pelos pesquisadores durante a fase de elicitação de requisitos, o que responde a segunda questão norteadora da revisão sistemática. Todavia, não foi especificado quais tipos de requisitos foram coletados por meio de suas aplicações, deixando a terceira questão norteadora como lacuna.

Nas produções selecionadas não foram encontradas informações sobre o uso das técnicas *contextual inquiry*, *braindraw* e *think-aloud*, sendo assim, não foi possível identificar em quais etapas estas técnicas podem ser utilizadas junto ao processo de ER.

Quadro 2. As respectivas técnicas utilizadas em cada produção científica

Técnica	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	SPIDe
Storytelling		X		X	X		X		X	X	X
Contextual inquiry											X
Brainstorm	X		X			X		X			X
Braindraw											X
Think-aloud											X

6. Considerações Finais

Este artigo apresentou resultados de uma RSL com o objetivo de explorar produções científicas que apresentem resultados da utilização das técnicas contidas no SPIDe para a geração de requisitos ou auxiliando as etapas do processo de engenharia de requisitos.

Com os resultados da RSL, foi possível identificar que duas das cinco técnicas que compõem o SPIDe, o *storytelling* e o *brainstorm*, já foram utilizadas durante a ER e podem gerar requisitos de software. Todavia, não foi possível identificar quais os tipos de requisitos são gerados e, para além disso, não foram encontradas a aplicação do *contextual inquiry*, *braindraw* e *think-aloud* deixando lacunas para a continuação da pesquisa.

Como trabalhos futuros será realizada uma análise em profundidade das produções científicas encontradas na RSL, além da aplicação de um estudo de caso para identificar se é possível engenhar requisitos de softwares educacionais por meio dessas técnicas.

Referências

- ACM. (1992). *ACM SIGCHI Curricula for Human Computer Interaction*. Retrieved from <http://www2.parc.com/istl/groups/uir/publications/items/UIR-1992-11-ACM.pdf>
- Baranauskas, M. C. C. (2013). O modelo semioparticipativo de design. In: *Codesign de Redes Digitais: tecnologia a serviço da inclusão social*, Maria Cecília C Baranauskas, Maria Cecília Martins and José Armando Valente (eds.). Penso.
- Barbosa, S. D. J., Paula, M. G. and Lucena, C. J. P. (2004). Adopting a communication-centered design approach to support interdisciplinary design teams. “*Bridging the Gaps II: Bridging the Gaps Between Software Engineering and Human-Computer Interaction*” *WIL Workshop - 26th International Conference of Software Engineering*. IEEE, 102–107.
- Barbosa, S. D. J. and Silva, B. (2010). *Interação Humano Computador*. Elsevier, Rio de Janeiro.

- Batista, C. (2014). *Um Processo Criativo de Descoberta de Contextos para Sistemas Sensíveis a Contexto*. Dissertação (Mestrado em Ciência da Computação). Universidade Federal de Pernambuco.
- Batista, C. and Silva, C. (2015). A creative process to elicit contexts for context sensitive systems. In: *Proceedings of the annual conference on Brazilian Symposium on Information Systems*. Brazilian Computer Society, 43.
- Besrou, S., Rahim, L. and Dominic, P. (2014). Assessment and evaluation of requirements elicitation techniques using analysis determination requirements framework. In: *Computer and Information Sciences (ICCOINS), 2014 International Conference on*. IEEE, p. 1–6.
- Boulila, N., Hoffmann, A. and Herrmann, A. (2011). Using Storytelling to record requirements: Elements for an effective requirements elicitation approach. In: *Multimedia and Enjoyable Requirements Engineering-Beyond Mere Descriptions and with More Fun and Games (MERE), 2011 Fourth International Workshop on*. IEEE, 9–16.
- De Souza, C. S. (2005). *The Semiotic Engineering of Human-Computer Interaction*. MIT Press.
- Ferre, X. and Moreno, A. M. (2004) Improving software engineering practice with HCI aspects development process. *Software Engineering Research and Application* 3026: 349–363.
- Gomes, A. S. and Padovani, S. (2005). Usabilidade no ciclo de desenvolvimento de software educativo. In: *Anais do Simpósio Brasileiro de Informática na Educação*. SBC.
- Gomes, A. S. and Wanderley, E. G. (2003). Elicitando requisitos em projetos de software educativo. In: *Anais do IX Workshop de Informática na Escola*. SBC, p. 119-130.
- Heiskari, J., Kauppinen, M., Runonen, M., and Mannisto, T. (2009). Bridging the gap between usability and requirements engineering. In: *Requirements Engineering Conference, 2009. RE'09. 17th IEEE International*. IEEE, p. 303-308.
- Junior, O. O. B. and Paola, Y. (2014). Análise de Abordagens Objetivas para Avaliação de Softwares Educacionais. In: *Proceedings of the 13th Brazilian Symposium on Human Factors in Computing Systems*. ACM, 353–356.
- Kitchenham, B. (2004). *Procedures for performing systematic reviews*. Keele, UK, Keele University, 33(2004), 1-26.
- Kotonya, G. and Sommerville, I. (1998). *Requirements engineering: processes and techniques*. Wiley Publishing.
- Kujala, S., Kauppinen, M., and Rekola, S. (2001). Bridging the gap between user needs and user requirements. *Advances in Human-Computer Interaction I*, 45-50.
- Lai, H., Peng, R. and Ni, Y. (2014). A collaborative method for business process oriented requirements acquisition and refining. In: *Proceedings of the 2014 International Conference on Software and System Process*. ACM, 84–93.

- Laporti, L., Borges, M. and Braganholo, V. (2009). *Athena: A collaborative approach to requirements elicitation*. *Computers in Industry* 60, 6 (2009), 367–380.
- Lowgren, J. (2014). Interaction Design - brief intro. In: *The Encyclopedia of Human-Computer Interaction* (2nd ed.), Mads Soegaard and Rikke Friis Dam (eds.). The Interaction Design Foundation, Aarhus, Dinamarca.
- Luck, R. (2003). Dialogue in participatory design. *Design Studies* 24, p. 523–535.
- Matos, E. S. (2013) *Dialética da Interação Humano-Computador: tratamento didático do diálogo midiaticizado*. Tese (Doutorado em Educação). Universidade de São Paulo.
- Memmel, T., Gundelsweiler, F. and Reiterer, H. (2007). Agile Human-Centered Software Engineering. In: *Proceedings of the 21st British HCI Group Annual Conference on People and Computers*. British Computer Society, p. 167-175.
- Ogunyemi, O. and Lamas, D. (2014). Interplay between human-computer interaction and software engineering. In: *Proceedings of Iberian Conference on Information Systems and Technologies*. CISTI.
- Pasquale, L., Spoletini, P., Pometto, D., Blasi, F. and Redaelli, T. (2013). Requirements engineering meets physiotherapy: An experience with motion-based games. In: *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 315–330.
- Pita, G. L.; Zobot, D.; Rosa, J. C. S.; Matos, E. (2017). Adapting the SPIDe to include visually impaired users in interaction design. In: *Proceedings of the 16th Brazilian Symposium on Human Factors in Computer Systems. (IHC'17)*. SBC, p. 505-508.
- Pressman, R. and Maxim, B. (2016). *Engenharia de Software-8ª Edição*. McGraw Hill Brasil.
- Rosa, J. (2016). *Design de Interação Multicultural: Um Framework Semioparticipativo para o (re)design da Interação de Softwares Educacionais*. Dissertação de Mestrado. Instituto de Matemática e Estatística. Universidade Federal da Bahia (UFBA).
- Rosa, J. C. S. and Matos, E. (2016a). Multicultural Interaction Redesign Using a Semio-Participatory Approach. In: *Proceedings of 17th International Conference on Informatics and Semiotics in Organisations*, Maria Cecilia Calani Baranauskas, Kecheng Liu, Lily Sun, Vânia Paula de Almeida Neris, Rodrigo Bonacin and Keiichi Nakata (eds.). Springer International Publishing, p. 240–242.
- Rosa, J. e Matos, E. (2016b). Semio-participatory framework for interaction (re)design of education softwares. In: *Proceedings of the 15th Brazilian Symposium on Human Factors in Computer Systems. (IHC'16)*. p. 33.
- Sayão, M., Staa, A. V. and Leite, J. C. S. P. (2003). *Qualidade em Requisitos*. Relatório Técnico, 47(03).
- Sakhnini, V., Berry, D. and Mich, L. (2010). Validation of the effectiveness of an optimized EPMcreate as an aid for creative requirements elicitation. In: *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 91–105.

- Seffah, A., Desmerais, C. and Metzker, E. (2005). HCI, Usability and Software Engineering Integration: present and future. In: *Human-Centered Software Engineering - Integrating Usability in the Software Development Lifecycle*, Ahmed Seffah, Jan Gulliksen and Michel Desmarais (eds.). Springer, p. 37-57.
- Silva, A. C., Silva, J. C. A., Penteado, R. A. D. and Silva, S. R. P. (2004). Aplicabilidade de Padrões de Engenharia de Software e de IHC no Desenvolvimento de Sistemas Interativos. In: *Anais do IV Congresso Brasileiro de Computação*. SBC, p. 118–123.
- Sinnig, D., Pitula, K., Becker, R., Radhakrishnan, T. and Forbrig, P (2010). Structured digital storytelling for eliciting software requirements in the ICT4D domain. In: *Human-Computer Interaction*. Springer, 58–69.
- Sommerville, I. (2007). *Engenharia de Software*, 8ª edição. Pearson Addison-Wesley, 22, 103.
- Svensson, R. and Taghavianfar, M. (2015). Selecting creativity techniques for creative requirements: An evaluation of four techniques using creativity workshops. In: *Requirements Engineering Conference (RE), 2015 IEEE 23rd International*. IEEE, 66–75.