

## Jogos para Ensino de Levantamento de Requisitos de *Software*: uma Revisão Sistemática de Literatura

Luis Henrique Carvalho Rosa<sup>1</sup>, Luísa Perin Lucca<sup>1</sup>, Eduardo Luis Lemos<sup>1</sup>,  
Giliane Bernardi<sup>1</sup>, Roseclea Duarte Medina<sup>1</sup>

<sup>1</sup>Universidade Federal de Santa Maria (UFSM)

{lhcr, llucca, giliane}@inf.ufsm.br, {elemos04, roseclea.medina}@gmail.com

**Resumo.** Grande parte dos fracassos no desenvolvimento de software ocorrem por erros na etapa de levantamento de requisitos. A origem deste problema pode ser explicada pela forma que são abordados esses conteúdos pelo professor, usualmente apenas por meio de aulas expositivas. Para modificar esse cenário, a inserção de jogos educacionais em sala de aula surge como solução. Com base neste contexto, o presente estudo realizou uma Revisão Sistemática de Literatura de trabalhos que apresentassem o desenvolvimento de jogos digitais para o ensino de levantamento de requisitos de software. Durante o processo, foram encontrados seis jogos que abordaram o tema e trabalhavam com técnicas de levantamento de requisitos de software.

**Palavras-Chave:** Elicitação de requisitos, E-learning, Jogos sérios.

## Games for Teaching Requirements Elicitation: A Systematic Literature Review

**Abstract.** *Most failures in software development occur because of errors in the requirements elicitation stage. The origin of this problem can be explained by the way in which these contents are addressed by the teacher, usually only through lectures. To modify this scenario, insertion of educational games in the classroom appears as a solution. Based on the context, the present study carried out a Systematic Literature Review of works that present the development of digital games for the teaching of software requirements elicitation. During the process, were found six games that approached the subject and worked with techniques of software requirements elicitation.*

**Keywords:** *Requirements Elicitation, E-learning, Serious Games.*

### 1. INTRODUÇÃO

O desenvolvimento de um *software*, genericamente, apresenta três fases: definição, desenvolvimento e manutenção. A Engenharia de Requisitos localiza-se na fase de definição, sendo responsável por entender o funcionamento do *software*, como será a experiência do usuário final e como o sistema irá influenciar nos negócios do cliente. O entendimento correto dos requisitos consiste na etapa mais crítica do desenvolvimento de um *software*, possuindo influência direta na qualidade final do produto (Thiry et al., 2010; Vargas et al., 2011; Unkelos-Shpigel e Hadar, 2015; Petri e Chiavegatti, 2015).

Conforme apontado pelos autores, o levantamento de requisitos viabiliza a definição das restrições e características que o sistema deverá apresentar. No entanto, a sua incorreta realização ainda se apresenta como grande responsável pelo fracasso de projetos, por isso a grande relevância de uma correta execução do mesmo (Thiry et al., 2010; Erfurth e Kirchner, 2010; Araújo et al., 2016). Tal problema pode estar

relacionado com a dificuldade destes profissionais em realizar corretamente esta etapa do desenvolvimento, o que demonstra uma falta dos conhecimentos necessários para essa tarefa.

Considerando o âmbito educacional, o processo de ensino e aprendizagem na área pode ser responsável por esta falta de habilidade, pois, usualmente, adota um método “tradicional” de ensino, trabalhando essas competências meramente por meio de aulas expositivas e exercícios mecânicos, não refletindo a realidade do processo (Hoffmann, 2012; Daneva, 2015; Araújo et al., 2016). Em busca de abordagens que preparem o estudante para situações mais próximas às encontradas no desenvolvimento de *software*, instituições de ensino têm adotado diferentes ferramentas para o auxílio às aulas expositivas, como a utilização de jogos educacionais. Utilizados para treinar, aprender ou simular atividades reais, esses possibilitam que o estudante adote posturas mais práticas frente aos conteúdos teóricos vistos em sala de aula (Monsalve et al., 2010; Figueiredo et al., 2010).

Considerando a área de Engenharia de *Software* e suas subáreas, é crescente a utilização de jogos, como forma de mitigar as deficiências dos profissionais da área (Campos et al., 2011; Araújo et al., 2016). Os autores complementam que, por meio das simulações e desafios, o estudante sente-se motivado a realizar a tarefa e desenvolver as competências.

Tendo conhecimento da relevância do levantamento de requisitos para o sucesso de um projeto e a deficiência que os profissionais da área apresentam ao saírem da universidade, realizou-se esta pesquisa com o propósito de analisar iniciativas que utilizem jogos digitais para auxiliar o ensino. Por meio de uma Revisão Sistemática de Literatura buscou-se especificamente jogos que trabalhassem a temática de levantamento de requisitos. Este trabalho apresenta na seção 2 a descrição do método de Revisão Sistemática de Literatura aplicada, que descreve como a pesquisa foi conduzida. A seção 3 traz a análise dos resultados encontrados. Por fim, na seção 4 são apresentadas as considerações finais.

## 2. REVISÃO SISTEMÁTICA DE LITERATURA

O método de revisão usado nesta pesquisa foi o denominado *Systematic Literature Review* (SLR), apresentado por Kitchenham (2004). A SLR é uma técnica para identificar, avaliar e interpretar um conjunto amplo de fontes de estudos anteriores disponíveis que possuam alguma relevância para uma questão de pesquisa específica, área temática ou fenômeno de interesse. Com uma SLR, o pesquisador pode estruturar e organizar as informações obtidas pela pesquisa. Além disso, possibilita a descrição dos trabalhos utilizados para investigação, bem como as bases de dados consideradas. Para a realização de uma SLR, três etapas devem ser seguidas:

- 1) Planejamento: etapa de identificação das questões de pesquisa e a elaboração do protocolo de revisão;
- 2) Condução: etapa de seleção dos estudos, de acordo com o protocolo elaborado;
- 3) Relatório: etapa de validação e análise dos resultados. Para averiguar os aspectos que devem ser considerados na seleção de trabalho que realizem a utilização de jogos para o ensino de levantamento de requisitos, foi utilizado como base o

relatório técnico de Kitchenham (2004), que apresenta procedimentos para a realização de uma SLR na área de Engenharia de Software.

Com relação aos objetivos e definição da questão de pesquisa, o presente estudo buscou e analisou artigos científicos que apresentassem jogos para o ensino de levantamento de requisitos de *software*. Com esse estudo visamos elucidar a questão principal deste trabalho [QP]: [QP] “*Que jogos têm sido desenvolvidos para trabalhar o ensino de levantamento de requisitos de software?*”. Para melhor entender o estado da arte, foi formulada a questão específica: [QE] “*Quais técnicas de levantamento de requisitos foram trabalhadas nos jogos?*”

A pesquisa foi realizada com base em trabalhos publicados entre 2010 e 2016, utilizando a *string* de busca (“*Games*”) AND (“*Requirements Elicitation*” OR “*Requirements Engineering*”), inserida em português e inglês, para abranger uma maior quantidade de trabalhos. Em um primeiro momento, as buscas foram realizadas na base do Google Acadêmico. Contudo, devido ao retorno de muitos artigos não relacionados ao tema, percebeu-se a necessidade de realizar buscas em bases com maior relevância específicas da área, envolvendo tecnologias, Engenharia de *Software* e Educação. Dessa forma, foram selecionados para este trabalho os seguintes repositórios: IEEE, ACM Digital, RENOTE, RBIE, FEES, SBIE e WEI. Para iniciar a busca nos repositórios selecionados foram adotados os critérios de avaliação apresentados na Tabela 1:

Tabela 1. Critérios de inclusão e exclusão

Critérios de Inclusão	Critérios de exclusão
<ul style="list-style-type: none"><li>• Trabalhos que discorrem sobre jogos desenvolvidos para o ensino de levantamento de requisitos;</li><li>• Trabalhos em português ou inglês;</li><li>• Publicados entre 2010 e 2016.</li></ul>	<ul style="list-style-type: none"><li>• Artigos duplicados;</li><li>• Artigos resumidos, resumos e apresentações;</li><li>• Jogos que não sejam digitais;</li><li>• Artigos que não descrevam o jogo.</li></ul>

Esses critérios foram aplicados durante as etapas de eliminação por títulos, por resumo e palavras-chave; eliminação por leitura diagonal e eliminação por leitura completa. O Gráfico 1 mostra uma relação de arquivos selecionados por repositório.

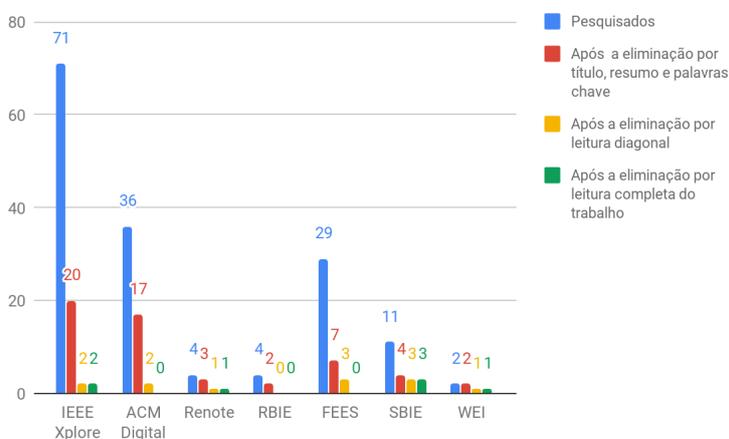


Gráfico 1. Artigos após cada fase de eliminação

No total, a pesquisa inicial resultou em 157 artigos encontrados, sendo que a barra azul-claro apresenta o número total de artigos retornados em cada repositório. A vermelha representa o número de artigos selecionados após a aplicação de eliminação por título, resumo e palavras-chave, restando 55 trabalhos. A barra amarela apresenta o número de artigos selecionados resultantes da aplicação de eliminação pela leitura diagonal do texto, com 11 artigos restantes. Por fim, a última barra, a verde, traz a relação dos artigos que passaram por todos critérios de seleção e serviram de base para esse trabalho, totalizando 7 artigos.

Quanto aos artigos eliminados, isso ocorreu, em sua maioria, por não apresentarem detalhadamente o desenvolvimento, não trabalharem com jogos digitais, serem duplicados ou não terem como foco a etapa de levantamento de requisitos de software. A próxima seção apresenta a análise da leitura detalhada dos 7 (sete) artigos resultantes da RSL.

### 3. ANÁLISE DE RESULTADOS

As principais discussões apresentadas nesta seção são fundamentadas nos resultados obtidos a partir da análise dos artigos selecionados após a etapa de extração da SRL, e fornecem respostas para as questões definidas no protocolo.

#### **QP: Que jogos têm sido desenvolvidos para trabalhar o ensino de levantamento de requisitos de *software*?**

Durante a pesquisa foram encontrados sete trabalhos que versavam sobre o ensino de levantamento de requisitos por meio de jogos digitais. Destes, dois discutiam o jogo UbiRE, proposto por Campos et al. (2011) e Santos et al. (2013), desenvolvido para trabalhar os conceitos de levantamento de requisitos para sistemas ubíquos. Optou-se por deixar os dois artigos pois um deles destacava o desenvolvimento do jogo e uma avaliação inicial de usabilidade do mesmo, enquanto o outro apresentava os resultados de uma avaliação realizada sob o ponto de vista da efetividade de uso.

O jogo UbiRE foi desenvolvido com a ferramenta GameMaker, sendo ambientado em uma casa inteligente. Seu objetivo central é montar um sistema inteligente que conecte todos os dispositivos da casa, atendendo o desejo do morador. Para progredir, o jogador deve analisar textos contendo instruções do morador, com o intuito de desvendar os requisitos necessários para realizar corretamente as conexões. O jogo dispõe de quatro fases, sendo cada fase referente a um cômodo da casa, estando estas dispostas conforme a dificuldade para realizar as conexões. Consoante o nível de dificuldade, a forma de apresentação dos requisitos se torna mais confusa, tornando complexa a realização das conexões. Ao final de cada fase é possível realizar uma simulação para verificar a funcionalidade de cada conexão.

Neste trabalho foram realizados dois processos avaliatórios, um de usabilidade e outro de efetividade. A avaliação de usabilidade buscava identificar os pontos fortes e fracos, bem como possíveis melhorias. Na avaliação de usabilidade destaca-se que as heurísticas utilizadas como indicadores foram pontuadas, em sua maioria, como razoáveis. Segundo os autores, isto deve-se ao fato de que o jogo ainda estava em desenvolvimento na época do teste. Na avaliação de efetividade foram aplicados um pré e pós-teste sobre os temas envolvidos, um treinamento sobre os temas, questionário de *feedback* e o formulário de caracterização do participante. O treinamento aplicado buscou igualar o conhecimento dos indivíduos. Por meio do formulário de

caracterização, foi possível dividi-los em dois grupos: o experimental (Grupo A), que usou o UbiRE, e o de controle (Grupo B), que usou outro jogo que não possuía a mesma temática, tendo efeito placebo. O Grupo de Controle teve uma melhora na média do pré para o pós-teste de 17,6% enquanto o Grupo Experimental teve um aumento de 20,8%. Os autores ainda relatam que nenhum participante do Grupo de Controle obteve nota maior que um participante do Grupo Experimental.

Outro trabalho encontrado durante a pesquisa foi o EAReq-Game, de Petri e Chiavegatti (2015), que trabalha conceitos de levantamento e priorização de requisitos. O jogo foi desenvolvido com a ferramenta RPG Maker VX e trata-se de um *Role Playing Game* (RPG) *single-player*, ambientado em um colégio. Atuando como o engenheiro de requisitos Ralph, o jogador deve coletar, organizar e priorizar os requisitos, com a finalidade de desenvolver um sistema acadêmico. Cada requisito possui três possíveis maneiras de classificá-los (obrigatório, necessário e fora do escopo), sendo cada alternativa equivalente a uma quantidade de pontos conforme a correteza das respostas. Um *feedback* é apresentado ao fim da fase para o jogador analisar seus acertos e erros, sendo que, para que possa avançar à próxima fase, deve alcançar uma quantidade mínima de pontos. Caso isso não ocorra e não haja mais requisitos a serem coletados o jogo é encerrado.

O jogo foi avaliado em termos de motivação, experiência do usuário e aprendizagem, contando com a participação de 14 estudantes. Para essa avaliação foi utilizado o modelo MEEGA de avaliações (Savi, 2011). O questionário aplicado aos estudantes continha questões de autoavaliação do estudante, onde os mesmos deveriam comparar seu conhecimento anterior e posterior à aplicação do jogo. Considerando a motivação, os índices apontam 70% de aprovação na maioria dos itens. Em termos da avaliação da experiência do usuário, os índices obtiveram aprovação em todos os itens, com exceção na Interação Social, que, segundo os autores, é justificada pelo fato do jogo ser *single-player*. Na avaliação da aprendizagem também foram apresentados índices satisfatórios. Ainda por meio das questões autoavaliativas foi ressaltado crescimento na aprendizagem sobre os conceitos. Os autores concluíram que o jogo cumpriu seus objetivos com sucesso, visto que a autoavaliação dos estudantes atingiu 78% de aprovação.

No trabalho de Vega et al. (2010) é apresentado o TREG, um jogo desenvolvido no ambiente virtual 3D Second Life. O cenário construído foi pensado com o propósito de manter o jogador imerso, e por esse motivo foi utilizado um ambiente familiar à maioria das pessoas, uma cozinha, onde, no papel de um chefe de cozinha, o jogador deve encontrar os ingredientes da receita para “Fazer um *workshop*”. O jogo associa os ingredientes da receita às boas práticas propostas por Gottesdiener (2002) no planejamento de *workshops* para o levantamento de requisitos de *software*. O cenário conta com a sala principal na qual há uma área de recepção, com um NPC (*Non Player Characters*) que é o guia e *teleporter* do jogador, conduzindo-o para a sala onde se inicia o jogo. Na sala chamada de cozinha há outro NPC, que auxilia o jogador, e objetos para interação. Na última sala é exibida a simulação do *workshop*, seguindo as escolhas feitas pelo jogador. Por meio destas simulações é possível perceber como o não seguimento das boas práticas impactaria no resultado final do *workshop*.

O trabalho teve como método de avaliação o teste de usabilidade, que avaliou a experiência do usuário referente a interface, mecânica de jogo e jogabilidade, por meio

de entrevistas, observações e questionários. Para a avaliação deste jogo contou-se com a participação de 9 usuários, sendo que o baixo número de participantes é embasado por Nilsen (2010), que propõe uma médias de cinco participantes para este tipo de avaliação. Desses 9 usuários, 3 eram novatos em mundos virtuais e 6 tinham experiência com este tipo de ambiente, sendo 3 destes usuários do Second Life.

Na interface foi avaliado o Painel de simulação, o NPC (*Miss Workshop*) e o HUD (*head-up-display*), que consiste em um *display* que guarda a pontuação do jogador. O painel de simulação foi o que mais recebeu comentários dentre os itens, justificado pelos autores pela sua maior utilização. O NPC foi muito pouco utilizado, tendo 6 usuários não interagido com ele após as instruções iniciais. Já com relação ao HUD, 5 dos participantes sequer perceberam sua presença.

Para avaliação da mecânica do jogo foram considerados dois critérios, a facilidade de uso e o seu entendimento. O jogo foi facilmente compreendido, entretanto os usuários tiveram dificuldades para encontrar a resposta correta para os desafios. Já para avaliar a jogabilidade foi utilizado o questionário IPQ (Igroup.org project consortium, 2010), que avalia a presença de um participante em um ambiente virtual. Nessa avaliação foram obtidos os índices mais elevados, o que mostra uma grande sensação de presença dentro do mundo virtual.

A próxima proposta é a de Rusu et al. (2011) que, por meio do jogo Earth Defense, simula uma invasão espacial para trabalhar levantamento de requisitos de *software*. O jogador é colocado no papel de um engenheiro de *software* responsável por construir um sistema de defesa com intuito de evitar a invasão alienígena e, por meio de entrevistas com cientistas e militares, descobrir os requisitos necessários para a construção deste sistema. Para isso, o jogador deve escolher quem será entrevistado, podendo escolher entre perguntas pré-selecionadas para conseguir obter as informações necessárias. Entretanto, os entrevistados podem dar informações confusas ou mentirosas, e caberá ao jogador escolher corretamente quem entrevistar, quais perguntas realizar e qualificar a relevância das informações. Para avaliação do jogo foram selecionados participantes de diversos níveis de escolaridade. Do total de 30 estudantes, 7 eram do ensino fundamental, 11 do ensino médio e 12 eram universitários. Ao terminarem o jogo, estes avaliaram o mesmo por meio de um questionário, classificando-o como divertido e interessante. Os autores afirmam, com base nas respostas dos questionários, que seu jogo pode auxiliar o ensino de conceito básicos. Eles ainda destacam que o propósito desta dinâmica de jogo é que o estudante pratique a técnica e perceba que suas decisões nesta etapa podem impactar diretamente no sucesso ou fracasso do desenvolvimento.

Já no trabalho de Thiry et. al. (2010) é apresentado o jogo Ilha de Requisitos, que objetiva auxiliar na compreensão do processo de engenharia de requisitos. Na história do jogo, o avião de “Jack Reqs”, um analista de requisitos, cai em uma ilha isolada, habitada por nativos canibais denominados “Nerds”. Jack os encontra e acaba se tornando temporariamente chefe dos nativos. O objetivo do jogo é ajudar Jack a encontrar uma saída da ilha, juntamente com os nativos, antes que um vulcão entre em erupção. Para auxiliar o jogador são apresentados *feedbacks* para cada ação, com dicas sobre os conteúdos.

Neste jogo, o estudante deve resolver desafios distribuídos pelo mapa e identificar os requisitos para a construção do barco que será utilizado para fugir da ilha.

Para a sua avaliação foi utilizado o *Framework* criado por Kochanski (2009), especificamente para estudos empíricos sobre os efeitos de aprendizagem pelo uso de jogos educacionais em Engenharia de *Software*.

Na avaliação foram comparados dados coletados por um pré e pós-teste dentro de dois grupos, de Experimentação (entraram no jogo), composto por 16 estudantes, e o Grupo de Controle (realizaram uma tarefa não relacionada a requisitos) composto por 15 estudantes. O Grupo de Controle obteve média de 4,46 no pré-teste caindo para 3,8 no pós-teste. Já no Grupo de Experimentação, a média do pré-teste foi de 3,31, subindo para 4,56 no pós-teste, concluindo que o jogo pode ter influenciado no aprendizado dos estudantes do Grupo de Experimentação. Na avaliação qualitativa os estudantes informaram que gostaram de jogar, mas o nível de dificuldade era muito alto, faltando mais informações de ajuda sobre o tema.

Por fim, no trabalho de Vargas et al. (2011) os autores apresentam o jogo denominado Elicit@ção, que se trata de um jogo *single-player* que simula o ambiente encontrado por um profissional da área, onde o estudante atua no papel de um analista de requisitos. O jogador deve interagir com os funcionários de uma empresa e, por intermédio desses diálogos, identificar os requisitos corretos para o desenvolvimento do sistema solicitado por este cenário. Os funcionários que interagem com o jogador são agentes de *software* que apresentam uma base emocional, baseada no modelo OCC (Ortony, Clore e Collins) (Ortony et al. 1988), para dar maior veracidade às interações, simulando assim um ambiente que o estudante encontrará no mercado de trabalho. Para isso, foi implementado um *Framework* Dialógico com intuito de representar os diversos perfis que os funcionários poderão apresentar.

Para a modelagem do ambiente foi utilizado o conceito de Instituições Eletrônicas (IE) que são sistemas computacionais formados por entidades autônomas que interagem conforme regras de comunicação e comportamento pré-definido. O seu desenvolvimento se deu por meio do ambiente *Electronic Institutions Development Environment* (EIDE) (EIDE 2017). Das ferramentas disponíveis no EIDE foram utilizadas o *Islander* para construção do cenário e o *aBuilder* para a construção dos agentes. Na modelagem das emoções dos agentes foram elencadas as emoções Satisfação/Frustração e Gratidão/Raiva. Esses estados emotivos são determinados a partir das ações do jogador. Para modelar o humor foram escolhidos dois tipos de humor: alegria e tristeza que são mais aleatórios e não mutáveis durante a interação entre o estudante e o agente.

Algumas constatações podem ser destacadas com relação aos seis jogos apresentados. Inicialmente, é possível classificar os mesmos na categoria de simuladores, pois todos buscam inserir o jogador em um contexto simulado de engenharia de requisitos, com exceção do jogo Ilha dos Requisitos que, por mais que busque esta imersão, se classifica como aventura e estratégia. Quanto aos desafios propostos, quatro são baseados em *Quests* (perguntas e respostas) enquanto dois buscam a resolução por meio de interações com possíveis usuários, representados, nestes casos, por agentes de *software*. Ainda, todos os jogos são *single player*, não colocando o estudante em situações de discussão e tomada de decisão conjunta, o que é uma característica de situações reais, evidenciadas em equipes de desenvolvimento de *software*. Por fim, cinco dos seis jogos foram avaliados junto a estudantes, sendo que o resultado, em todos os casos, foi considerado satisfatório, evidenciando o potencial do

uso de jogos como estratégia promissora no processo de ensino e aprendizagem de Engenharia de *Software*.

Com a análise dos trabalhos também foi possível verificar que técnicas de levantamento de requisitos de *softwares* foram trabalhadas por meio dos jogos apresentados, tornando possível responder à questão específica desta pesquisa.

**QE: Quais técnicas de levantamento de requisitos de *software* foram trabalhadas nos jogos?**

Os seis trabalhos apresentados se utilizaram basicamente de três métodos para o levantamento de requisitos de *software* (Sajid et al. 2010): (1) Conversação, onde foram abordadas as técnicas de entrevista e workshop; (2) Observação, na qual foi desenvolvida a técnica de observação das atividades realizadas pelos usuários; e, por fim, (3) Investigação Analítica, onde a técnica utilizada foi a análise de documentação.

Considerando a interação direta com usuários, a técnica de entrevista obteve o maior número de ocorrências, sendo trabalhada em dois dos seis jogos apresentados. Nela, o analista interage com um *stakeholder*, adotando diretrizes de maneira a deixá-lo confortável para que o mesmo forneça as informações necessárias. Essa técnica é trabalhada em Vargas et al. (2010) e Rusu et al. (2011), sendo que os autores argumentam que a sua aplicação requer do analista qualificações sociais difíceis de desenvolverem sem uma atividade prática que simule situações reais.

Ainda abordando a estratégia de diálogo, o trabalho de Veja et al. (2010) ensina boas práticas para aplicação e condução da técnica de *workshop*. Nessa técnica são realizadas reuniões muito bem estruturadas entre os analistas e uma seleção de *stakeholders* que representem a organização e contexto no qual o sistema será usado. Os autores ressaltam que a colaboração é um fator próprio do levantamento de requisitos de *software*, pois envolve clientes, desenvolvedores e usuários.

Já no jogo de Petri e Chiavegatti (2015) é trabalhada a técnica de observação. Nessa técnica o analista deve coletar informações observando a rotina de trabalho do local. Segundo os autores, para execução da técnica, é necessário que o analista tenha treinamento por meio de atividades práticas.

No trabalho de Campos et al. (2011) e Santos et al. (2013), os autores não desenvolveram o jogo especificamente para trabalhar uma técnica, contudo é possível relacionar o trabalho ao método analítico por meio da técnica de análise de documentação. Essa técnica consiste no estudo de diversos documentos que possam influenciar no desenvolvimento do sistema (Rezende, 2005). É possível observar sua aplicação no referido jogo durante a análise da lista de pedidos do morador, que pode ser vista como um documento que contém os requisitos necessários para fazer as conexões que atendam essas necessidades do proprietário.

Finalizando, no trabalho de Thiry et al. (2010) não é apresentada uma técnica específica, sendo que o jogo discute o processo de engenharia de requisitos como um todo e, durante seus desafios, a etapa de levantamento de requisitos é presente, visto que o estudante deve descobrir os requisitos necessários para escapar da ilha.

#### 4. CONSIDERAÇÕES FINAIS

Por meio da Revisão Sistemática de Literatura averiguou-se a importância de simulações de situações reais para a formação de um profissional da área de engenharia de requisitos. Os autores concluem que os jogos possibilitam ilustrar a teoria e que sua aplicação no ensino contribui para melhorar a motivação e concentração, bem como melhorar as habilidades necessárias para um profissional da área, pois, por meio deles, é possível simular ambientes reais.

Os artigos encontrados nesta pesquisa mostraram ganhos qualitativos dos estudantes após serem expostos a situações semelhantes às encontradas na indústria de *software*. Pela avaliação dos trabalhos, pode-se perceber que essas iniciativas têm potencial para formar estudantes mais preparados, o que poderá resultar em *softwares* mais eficientes e com menor taxa de fracasso nos projetos. A análise ainda mostrou que há uma diversificação das técnicas trabalhadas, contudo, que ainda é pequeno o número de jogos desenvolvidos para o ensino de levantamento de requisitos, o que propicia uma exploração e desenvolvimento de novas aplicações envolvendo a área, objetivo que desejava-se investigar por meio da SRL realizada.

#### REFERÊNCIAS

- Araújo, D. N.; Cruz, M. L. P. de M.; Pimentel, J. H.; Duque, M.; Alencar, F. “Jogos Educativos no Ensino da Engenharia de Requisitos”. In: **IX Fórum de Educação em Engenharia De Software (FEES)**, Maringá, 2016.
- Campos, B.; Lima, T.; Santos, R., Werner, C.; Limoeiro, F. “Experiência de Projeto e Desenvolvimento de Jogo para Ensino de Engenharia de Requisitos para Sistemas Ubíquos”. In: **Anais do Simpósio Brasileiro de Informática na Educação**, Vol. 1, No. 1, 2011.
- Daneva, M. “Play-testing and requirements engineering: implications for research and teaching”. In: **Proceedings of the Second International Workshop on Requirements Engineering and Testing**. IEEE Press, pp. 9-12, 2015.
- Erfurth, I.; Kirchner, K. “Requirements Elicitation with Adapted CUTA Cards: First Experiences with Business Process Analysis”. In: **Engineering of Complex Computer Systems (ICECCS), 2010 15th IEEE International Conference on**. IEEE. pp. 215-223, 2010.
- Figueiredo, K.; Ferreira, J.; Murta, L.; Clua, E. “Um jogo de estratégia de gerência de configuração”. In: **III Fórum de Educação em Engenharia de Software (FEES), Simpósio Brasileiro de Engenharia de Software (SBES)**, Salvador, 2010.
- Gottesdiener, E. “**Requirements by Collaboration: Workshops for Defining Needs**”, Addison Wesley, 2002.
- Hoffmann, A. “A trainer's guideline to teaching soft skills using improvisation theater: a workshop format exemplified on a requirements engineering game”. In: **Proceedings of the 16th European Conference on Pattern Languages of Programs**. ACM. (p. 4), 2012.
- Igroup.org project consortium**. Igroup Presence Questionnaire. Disponível em: <<http://www.igroup.org/projects/ipq/>>. Acesso em: 11 dez. 2017.

Kitchenham, B. "Procedures for performing systematic reviews," **Keele, UK, Keele Univ.**, vol. 33, no. 2004, pp. 1–26, 2004.

Kochanski, Djone "Um framework para apoiar a construção de experimentos na avaliação empírica de jogos educacionais". Dissertação apresentada à Universidade do Vale do Itajaí como requisito para a obtenção do título de Mestre em computação. São José, Brasil, 2009.

Monsalve, E. S.; Werneck, V. M. B.; Leite, J. C. S. P. "Simules-w: Um jogo para o ensino de engenharia de software". In: **III Fórum em Educação de Engenharia de Software (FEES), Simpósio Brasileiro de Engenharia de Software (SBES)**, Salvador, 2010.

Nilsen, J. "Why You Only Need to Test with 5 Users". Disponível em: <<http://www.useit.com/alertbox/20000319.html>>. acessado em: 11 dez. 2017.

Ortony, A.; Clore; G. L.; Collins, A. **The cognitive structure of emotions**. 1988.

Petri, G.; Chiavegatti, N. C. "Um role playing game para o ensino de elicitación e análise de requisitos". **renote**, 13(1), 2015.

Rezende, D. A. "Engenharia de Software e Sistemas de Informação". Brasport, 3ª edição, 2005.

Rusu, A.; Russell, R.; Cocco, R. "Simulating the software engineering interview process using a decision-based serious computer game". In: **Computer Games (CGAMES), 16th International Conference on**. IEEE. pp. 235-239, 2011.

Sajid, A.; Nayyar, A.; Mohsin, A. "Modern Trends Towards Requirement Elicitation." **Proceedings of the 2010 national software engineering conference**. ACM, 2010.

Santos, R.; Lima, T.; Campos, B.; Werner, C. "Experiência na Avaliação de Efetividade de um Jogo para Ensino de Conceitos de Requisitos em Sistemas Ubíquos". In: **Anais do Simpósio Brasileiro de Informática na Educação**. 2013.

Savi, R. "Avaliação de jogos voltados para a disseminação do conhecimento". Florianópolis, SC: UFSC, 2011. Originalmente apresentado como Tese de Doutorado, Universidade Federal de Santa Catarina, 2011.

Thiry, M.; Zoucas, A.; Gonçalves, R. Q. "Promovendo a Aprendizagem de Engenharia de Requisitos de Software através de um Jogo Educativo". In: **Anais do Simpósio Brasileiro de Informática na Educação**. Vol. 1, No. 1, 2010.

Unkelos-Shpigel, N.; Hadar, I. "Inviting everyone to play: Gamifying collaborative requirements engineering". In: **2015 IEEE Fifth International Workshop on Empirical Requirements Engineering (EmpiRE)**. IEEE. pp. 13-16, 2015.

Vargas, D.; Moro, T. B.; Dambrosio, G. M.; Cassal, M. L.; Bernardi, G.; Cordenonsi, A. Z. "Desenvolvimento de um Jogo de Empresa baseado em Agentes de Software e Instituições Eletrônicas para simulação de Elicitación de Requisitos de Software". In: **XVIII Workshop de Educação em Informática (WEI)**. Belo Horizonte–Minas Gerais. 2010.

Vega, K.; Pereira, A.; Robichez, G.; Fuks, H. "Testes de Usabilidade em TREG: avaliando um jogo de treinamento no Second Life". **Social Computing in the Global Workplace: Connecting People and Connecting Work**. 2010.