

Mapeamento sistemático de metodologias de desenvolvimento centrado no usuário para jogos sérios

Helder Cognaco de Oliveira, Marcelo da Silva Hounsell, Avanilde Kemczinski

Programa de Pós-Graduação em Computação Aplicada (PPGCA)
Departamento de Ciência da Computação (DCC)
Universidade do Estado de Santa Catarina (UDESC)
Joinville, SC – Brasil

heldercdo@gmail.com, {marcelo.hounsell, avanilde.kemzcinski}@udesc.br

Abstract. *The development of Serious Games (SG) is an effective way to engage and educate users. SGs need to comply with the requirements of two types of users: content specialists and players. This article aims to identify which SG development methodologies support the participation of players into design and development of game elements. A systematic literature mapping was conducted to identify such methodologies and classify them regarding their life-cycle, how and when users participate. Seventeen methodologies were found, with players' involvement only at evaluation and requirements analysis activities. No methodology was found that support player's participation throughout the whole development process. Therefore, a participatory design methodology is required and seen as beneficial for games that target drug addicted users.*

Resumo. *O desenvolvimento de Jogos Sérios (JS) é uma forma eficaz de engajar e educar usuários. JSs precisam atender os requisitos de dois tipos de usuários: conteudistas e jogadores. Este artigo busca identificar quais metodologias de desenvolvimento de JS suportam a participação de jogadores no design e desenvolvimento de elementos de jogo. Um mapeamento sistemático de literatura foi conduzido para identificar tais metodologias e classificá-las quanto ao seu ciclo de vida, como e onde usuários participam. Dezesete metodologias foram encontradas, com envolvimento de jogadores apenas em etapas de avaliação e análise de requisitos. Nenhuma metodologia foi encontrada que oferece suporte à participação do jogador durante todo o processo de desenvolvimento. Portanto, uma metodologia de design participativo é necessária e vista como benéfica para jogos voltados a usuários viciados em drogas.*

1. Introdução

Em agosto de 2013 o governo do Estado de Santa Catarina lançou o projeto Rede Estadual de Atenção a Dependentes Químicos, denominado de Projeto Reviver. O projeto visa a acreditação das comunidades terapêuticas (CTs) que atuam no estado, além de desenvolver atividades de pesquisa e inovação junto à Universidade do Estado de Santa Catarina (UDESC) dentre outras [De Lima 2013].

Dentre as atividades de inovação, o projeto Reviver busca investigar o uso de jogos digitais no contexto do acolhimento e reinserção social dos dependentes químicos.

Dentre os jogos digitais destacam-se os Jogos Sérios (JS) que são aqueles cujo propósito principal não é o entretenimento mas, a educação, saúde, treinamento e etc. Nos JS existem pelo menos dois tipos de usuários a considerar: o especialista no conteúdo, aqui chamados de conteudista, que muito frequentemente é quem contrata o novo JS e; o usuário final, aqui chamado de jogador, normalmente um aprendiz. No processo de criação de JSs, pode-se ter o dependente químico envolvido em pelo menos duas formas: (a) passiva, sendo o beneficiário do resultado final do processo – o jogo – ou; (b) ativa, participando de etapas de criação do jogo. Esta metodologia pode até evoluir e se tornar uma atividade terapêutica dentro das CTs, engajando os jogadores no processo de criação deste material. Assim, espera-se que os participantes fiquem mais conscientes dos riscos de recaída que enfrentarão ao serem reinseridos na sociedade.

JS trazem benefícios pedagógicos não somente através do seu uso como produto final, pois participando do processo de design e desenvolvimento de JS, alunos são motivados e estimulados em seu pensamento criativo e aprendem mais sobre o tema abordado pelo jogo [Sevilla et al. 2012, Navarrete 2013]. Além disso, um dos estágios do modelo de engenharia de usabilidade proposto por [Nielsen 1993], indica a utilização de design participativo como forma de melhor adequar características de um sistema ao usuário final.

Sendo assim, este artigo pretende levantar através de um mapeamento sistemático da literatura quais metodologias centradas no usuário para criação de JS foram publicadas nos últimos 10 anos, e dentre estas quais dão suporte à participação do jogador em alguma etapa do ciclo de vida do software, com o objetivo de identificar qual delas pode melhor ser aprimorada para se adequar ao projeto Reviver.

Os conceitos e a prática do mapeamento sistemático estão descrito na seção 2, enquanto os resultados encontrados no mapeamento são demonstrados na seção 3. A discussão acerca das metodologias encontradas dá-se na seção 4 enquanto as conclusões finais deste artigo encontram-se na seção 5.

2. Mapeamento Sistemático

O processo de mapeamento sistemático da literatura (MSL) tem como objetivo fazer uma pesquisa em largura na literatura, e não em profundidade ([Kitchenham e Charters 2007], e [Petersen et al. 2008]). Por se basear em um protocolo de pesquisa, este mapeamento pode ser reproduzido por outros pesquisadores. Os itens que constituem este protocolo são: objetivos de busca, palavras-chave, filtros, mecanismos de busca acadêmica (MBA), critérios de exclusão/inclusão, e dados a serem levantados.

A definição dos objetivos de busca formaliza o que pretende-se encontrar na literatura através do mapeamento. Os objetivos serão consequentemente refletidos nos critérios, filtros e dados que serão levantados pela pesquisa. O Objetivo Geral deste trabalho é “identificar metodologias de desenvolvimento centrado no usuário para JS”. Os Objetivos Específicos são (i) catalogar em quais etapas ocorre a participação de usuário; (ii) definir qual tipo de usuário tem participado do design de JS e; (iii) identificar técnicas utilizadas quanto à participação do usuário.

A criação das palavras-chave se baseou na definição do objeto (metodologia), da ação (desenvolvimento) e da área de aplicação (jogos sérios). Com a utilização destas definições em inglês, e do uso de vários sinônimos para cada um dos três termos, a chave de busca inicialmente foi:

(Desenvolvimento OR Development OR Design OR "Game Design" OR Criação OR Making) AND (Metodologia OR Methodology OR Técnica OR Technique OR Processo OR Process OR Método OR Method) AND ("Jogos Sérios" OR "Jogos Educacionais" OR "Serious Games" OR "Educational Games")

Após a execução de teste piloto do MSL com esta frase de busca, verificou-se que o MBA IEEE Xplorer não dava suporte a frases com mais de 15 termos. Isto, somado à percepção empírica de quais termos não traziam resultados muito relevantes (termos em português, por exemplo), resultou na reformulação da chave de busca em sua forma final onde os termos abaixo em negrito foram procurados apenas nos resumos dos artigos, conforme disponibilidade do MBA:

(Design OR Develop*) AND (Method* OR Technique OR Process OR Framework) AND ("Serious Game" OR "Educational Game" OR "Serious Games" OR "Educational Games")*

Também foram definidos filtros auxiliares para artigos publicados em periódicos ou eventos entre os anos 2004 e 2014, em inglês, e da área de Ciência da Computação. Como os possíveis filtros em chaves de busca diferem de um MBA para outro, estes filtros não foram considerados no momento da criação das chaves de busca. Os MBAs escolhidos para esta pesquisa foram Web of Science, IEEE Xplore, ACM Digital Library e ScienceDirect. Estes MBAs foram selecionados porque: (a) em uma recente análise quantitativa de 40 MBAs, estes mecanismos estiveram entre os 10 mais bem avaliados, sendo que o Web of Science foi classificado como o melhor [Buchinger; Cavalcanti e Hounsell 2014]; (b) uma recente revisão sistemática de literatura também aponta que existe uma grande quantidade de artigos relacionados a jogos publicados no ScienceDirect [Connolly et al. 2012] e; (c) os MBAs IEEE Xplore e ACM Digital Library são de reconhecida relevância na área de Ciência da Computação.

A frase de busca foi adaptada em cada um dos mecanismos e a lista de artigos retornados foi armazenada. Para cada um dos artigos, foram aplicados os seguintes Critérios de Exclusão (CE): *CE1 - Não é um artigo sobre jogo digital; CE2 - Não é um artigo sobre JS; CE3 - Não propõe/altera uma metodologia de desenvolvimento de JS.*

A aplicação destes critérios deu-se inicialmente nos resumos dos artigos. Quando não foi possível concluir sobre todos os critérios desta forma, analisou-se de forma superficial os capítulos de introdução e conclusão. Os artigos onde não foram identificados nenhum dos CE, foram então analisados para os seguintes Critérios de Inclusão (CI): *CI1 - Acesso ao artigo na íntegra; CI2 – Artigo é do tipo Full Paper; CI3 – A metodologia mencionada engloba um ciclo de vida completo (análise de requisitos, desenvolvimento e avaliação/manutenção).*

Os artigos que atendiam todos os CIs foram então analisados de forma mais profunda. A leitura dos artigos buscou levantar as informações relevantes sobre as metodologias encontradas, no que tange à participação de usuários em uma ou mais etapas destas metodologias. As informações levantadas foram: (i) Qual o ciclo de vida de JS proposto; (ii) Em quais etapas ocorrem a participação de usuário; (iii) Qual o tipo de usuário participante; (iv) Qual a técnica de coleta de dados utilizada para a participação, e; (v) Qual ferramenta ou material foi utilizado na técnica de participação.

As metodologias foram classificadas quanto ao seu tipo de ciclo de vida em: (i) sequencial, que tem como característica uma evolução linear, onde uma etapa deve ser

totalmente executada para que a etapa seguinte seja iniciada. Um exemplo de metodologia sequencial aplicada no desenvolvimento de software é o modelo cascata. Ao final deste ciclo, tem-se um produto de software pronto, sendo necessária apenas a manutenção do mesmo [Preece; Rogers e Sharp 2002], e; (ii) iterativa, cuja característica principal é o comportamento incremental de seus modelos. Uma determinada sequência de etapas é executada repetidamente até o software chegar a sua forma final, sendo que cada iteração gera uma nova versão do software. A principal vantagem desse modelo interativo está na possibilidade de se adaptar a mudanças que surgem entre estas iterações, pois no modelo sequencial não há qualquer tipo de análise de alterações para um projeto já em execução, exceto após a construção do sistema [Sommerville 2011]. Com isso, é mais comum que metodologias de desenvolvimento iterativas tenham repetição de contato com *stakeholders*, visando identificar e adaptar-se a alterações propostas por eles.

Durante o levantamento dos dados foi encontrada uma exceção para esta classificação, pois o trabalho de [Rüppel e Schatz 2011] propõe uma metodologia híbrida, com atividades iniciais sendo executadas de forma concorrente, e o resto das atividades sendo executadas de forma iterativa.

Para verificar onde ocorre a participação e qual o tipo do usuário participante, foram estipuladas cinco etapas genéricas de ciclo de vida, que se baseiam no modelo cascata clássico [Sommerville 2011], e abrangem todas as metodologias avaliadas. Cada etapa de cada metodologia encontrada foi relacionada para uma destas etapas genéricas, que são: Análise de Requisitos; Design; Implementação; Testes e; Avaliação.

A etapa de análise de requisitos constitui a definição de quais requisitos o produto deve atender, incluindo conceitualização dos jogos, definição do que deve ser ensinado, público alvo e etc. Durante o design define-se a arquitetura de hardware e/ou software que irá compor o produto. Definições de abordagens pedagógicas, narrativas, níveis e outros elementos do *Game Design* também são realizadas neste momento. A etapa de implementação é responsável por transformar as definições de design em uma versão do produto, através de codificação e geração de protótipos. Os testes verificam se o que foi implementado está de acordo com os requisitos do jogo, uma tarefa interna de responsabilidade da equipe de desenvolvimento. Finalmente, há a etapa de avaliação, onde os *stakeholders* verificam se o produto, seja em sua versão final ou apenas um protótipo (nas metodologias iterativas), atende ou não às suas necessidades.

3. Resultados

Com a execução do protocolo de pesquisa para o mapeamento sistemático de literatura, foram encontrados nos MBAs 649 resultados de busca, que foram reduzidos a 17 artigos com a aplicação dos CEs e CIs, conforme explicitado na Tabela 1.

A pesquisa realizada no mecanismo Web of Science foi a mais relevante pois dos artigos retornados, 3,96% atendiam os critérios de seleção do protocolo de pesquisa. Resultados de ACM Digital Library, IEEE Xplore e ScienceDirect foram relevantes em 2,61%, 2,18% e 2% dos casos, respectivamente. Uma planilha digital em formato *xlsx*

contendo informações do protocolo de pesquisa (critérios, filtros, chaves de busca) e também os resultados encontrados, está disponível *online*¹.

Tabela 1. Resultados encontrados em cada MBA

MBA	Resultados	Após CE	Após CI
Web of Science	102	15	4
IEEE Xplore	230	40	5
ScienceDirect	50	6	1
ACM Digital Library	268	90	7
TOTAL	649	151	17

Dos 17 artigos encontrados, cada um apresentava em seu conteúdo uma metodologia, que foi classificada quanto ao seu tipo de ciclo de vida. Além desta classificação, foi identificado se existe alguma forma de participação de usuários no processo, seja o usuário conteudista ou o jogador, conforme demonstrado na Figura 1.

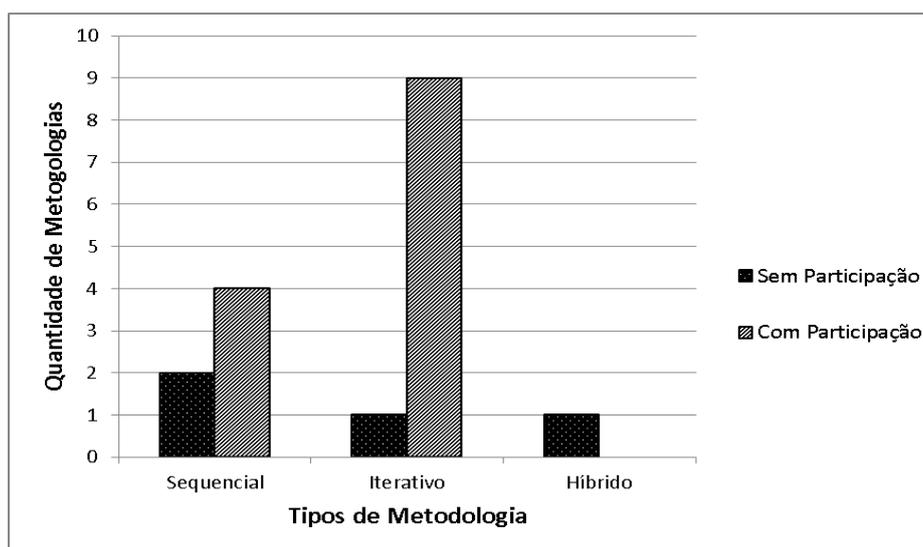


Figura 1. Relação entre quantidade e tipos de metodologias

Analisando a Figura 1 é possível perceber que 13 de todas as 17 metodologias (aproximadamente 76%) dão suporte à participação do usuário. Destas metodologias, 9 têm ciclo de vida iterativo e 4 têm ciclo sequencial.

A participação dos usuários foi dividida em três tipos: só conteudista; do conteudista e do jogador, e; só jogador. Na Figura 2 é apresentada a quantidade de metodologias que envolvem o usuário em cada uma das etapas listadas, além do tipo de participação que ocorre.

Apesar de existir a participação do conteudista em todas as etapas, foi mais frequente encontrar este tipo de participação durante a fase de análise de requisitos. Uma abordagem balanceada, utilizando tanto conteudista quanto jogador, também

¹ Disponível em: <https://www.dropbox.com/s/306fgj5vn411e38/MapeamentoMetJS.xlsx>

ocorre em pequeno número na fase de análise, porém é comum na fase de avaliação. Atividades onde somente o jogador participa ocorrem apenas na fase de avaliação.

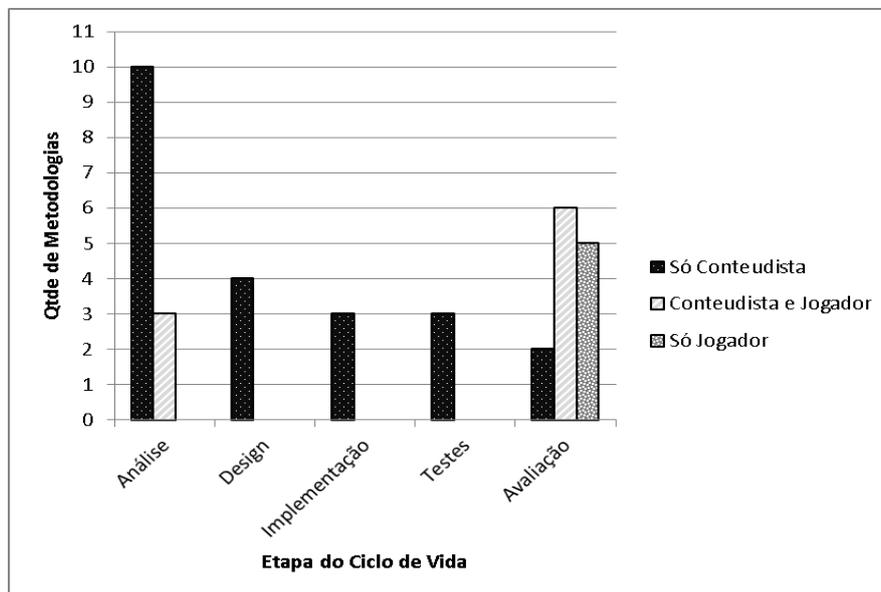


Figura 2. Relação entre quantidade de metodologias e suporte a usuários em cada uma das etapas

Quanto à técnica necessária para a participação do usuário, poucas metodologias encontradas fazem alguma recomendação além do esperado, ou seja, recomendam apenas entrevistas, questionários e observação direta. Dentre as poucas exceções que ocorrem, dois trabalhos aconselham a utilização da técnica *Think-Aloud*², porém sua utilização acontece em etapas diferentes do ciclo de vida do jogo. Na metodologia proposta por [Eagle e Barnes 2012], o *Think-Aloud* é utilizado na etapa de análise de requisitos, em que, por exemplo, um analista pode observar um usuário tentando aprender algum conceito sem a ajuda de um JS. Desta forma o analista pode entender a linha de raciocínio que é seguida pelo usuário em um caso de aprendizagem convencional. Já na metodologia de [Lazarou 2011], o *Think-Aloud* é utilizado em etapas de avaliação de usabilidade, onde se observa o jogador interagindo com uma versão do JS desenvolvido.

4. Discussão

Dentre os desafios na criação de JS, destaca-se a dificuldade em manter um equilíbrio entre aprendizagem, diversão e motivação [Franzwa; Tang e Johnson, 2013]. Observando que todas as etapas do ciclo de vida de um JS possuem ao menos duas metodologias recentemente propostas na literatura que dão suporte à participação de usuário conteudista, então entende-se que a atenção quanto ao aspecto pedagógico de JS já está fomentada e sendo trabalhada por diversos autores.

² *Think-Aloud* é uma técnica que consiste em observar um usuário realizando alguma atividade enquanto ele relata em voz alta os seus pensamentos em cada ação [Preece; Rogers e Sharp 2002].

Metodologias com a participação de jogadores estão concentradas nas últimas etapas do ciclo de vida de JS, com algumas exceções ocorrendo na etapa de análise de requisitos. Atividades de observação do jogador são explícitas na etapa de análise por alguns autores [Eagle e Barnes 2012, Rankin et al. 2008], porém apenas [Lazarou 2011], que criou uma metodologia chamada *Eight Phase Methodology* (EPM), aborda o jogador cara-a-cara nesta etapa, através de entrevistas. A EPM, portanto, pode ser vista como um ponto de partida para a reformulação de uma ou mais metodologias para alcançar o equilíbrio visado por [Franzwa; Tang e Johnson, 2013].

A EPM estabelece 8 etapas para o desenvolvimento de JS, sendo que tanto conteudistas quanto jogadores participam de algumas etapas. Inicialmente os usuários são consultados através de entrevistas em um levantamento de requisitos. Quando são encontradas contradições nos requisitos levantados, os usuários novamente são consultados para sugerir possíveis soluções. Após o desenvolvimento do jogo, a EPM especifica que devem ser executados testes de usabilidade com os usuários utilizando a técnica *Think-Aloud*, buscando melhorias antes do jogo ser efetivamente implementado em sala de aula. Após esta implementação, são colhidos dados enquanto os alunos executam o jogo. Estes dados podem ser obtidos em sala de aula através de observação, gravação em vídeo e outros. Por fim, coletam-se dados através de entrevistas após a execução do jogo, a fim de verificar se os conteudistas e os jogadores estão satisfeitos com o produto e se ele atende às suas expectativas e necessidades.

Nota-se que a participação de jogadores na etapa de implementação e design não é suportada por nenhuma das metodologias encontradas neste mapeamento sistemático de literatura. Isto desconsidera a oportunidade do jogador se envolver diretamente com atividades como produção gráfica, design de personagens, sonorização, narrativa e programação. Etapas do ciclo de vida de JS como análise de requisitos, design e prototipação, devidamente orientadas por profissionais técnicos e da área de saúde (p. ex.: psicólogos, terapeutas ocupacionais) podem ser desenvolvidas de maneira criativa e laboriosa e, portanto, apropriadas para uso em ambiente de CTs.

Assim, é possível criar um JS que não apenas atenda aos fatores pedagógicos, mas também às regras, mecânicas e aspectos visuais que estão de acordo com as expectativas dos seus futuros jogadores. Com isso, presume-se que a participação dos dependentes químicos do projeto Reviver na construção de jogos para prevenção à recaída aumentará a motivação dos mesmos em jogá-los.

4.1. Design Centrado no Usuário e Jogos Sérios

Estudos em Interação Humano-Computador apontam a importância de envolver usuários durante o design de softwares interativos desde as primeiras etapas de criação, através de design centrado no usuário. Existem diversas aplicações deste tipo de design que permitem este envolvimento, por exemplo [Preece; Rogers e Sharp 2002]: o design contextual é uma abordagem etnográfica onde um designer deve inserir-se no ambiente do usuário, e observar como ele executa suas tarefas; o design baseado em cenários busca estabelecer uma narrativa sobre alguma atividade do usuário; já no design participativo, o usuário pode interagir diretamente com a equipe de desenvolvimento para criar diversos componentes do software, desde requisitos até elementos de interface em um protótipo.

Tendo em vista o objetivo do Projeto Reviver em não apenas fazer com que dependentes químicos utilizem JS, mas que eles participem efetivamente da construção dos jogos, julga-se que design participativo é a ferramenta de Interação Humano-Computador que melhor se encaixa nesta proposta, embora as outras técnicas de design possam também contribuir neste processo.

4.2. Design Participativo

Design participativo (DP) é um método utilizado em desenvolvimento organizacional para aumentar a probabilidade de um sistema ser aceito para uso pelos próprios usuários, se estes forem envolvidos nas etapas formativas da construção do sistema [Helander; Landauer e Prabhu 1997]. Além disso, o DP pode ser utilizado para especificar com maior eficácia os requisitos funcionais dos usuários [Chin Jr 2004].

O DP consiste em transformar o usuário em um membro ativo da equipe de design, possibilitando que perspectivas e necessidades do software sejam propostas, desenvolvidas e/ou avaliadas por quem irá efetivamente usá-lo. As técnicas de DP definem não apenas como o usuário irá participar, mas também quando irá participar, ou seja, em qual momento do ciclo de vida do software isto deve ocorrer [Helander; Landauer e Prabhu 1997].

Técnicas de DP, como o PICTIVE³, têm sido utilizadas na construção de JS, como no trabalho de [Ribeiro e Veloso 2012], onde os autores também apontam uma tendência em utilização de design participativo na criação de ferramentas multimídia para crianças. O trabalho de [Matsunaga; Bispo e Borges. 2013] vai além e incorpora a utilização da técnica de DP BrainDraw⁴ em uma metodologia informal de construção de JS. Voltado à criação de jogos de entretenimento em sala de aula, [Moser 2013] se apóia não somente em DP, mas também em design centrado no usuário para um modelo de Interação Criança-Computador, o que demonstra a possibilidade de conciliar diferentes técnicas orientadas ao usuário em uma mesma metodologia.

As 61 diferentes técnicas de design participativo catalogadas em [Helander; Landauer e Prabhu 1997] podem ser aplicadas nos mais diferentes cenários, possibilitando que os DQs ajudem a criar enredos, personagens, cenários, pontuações entre outros, gerando um sentimento de propriedade sobre o jogo final. A aplicação de design participativo com os DQs também pode trazer benefícios sociais e cognitivos devido às interações e atividades desenvolvidas nas dinâmicas, podendo tornar-se parte efetiva do processo terapêutico exercido pelas CTs.

5. Conclusões

Com o intuito de criar jogos sérios (JS) com a participação de dependentes químicos no projeto Reviver, este trabalho analisou 649 artigos científicos obtidos de um mapeamento sistemático de literatura, nos 4 mecanismos de busca acadêmica mais relevantes para a área da computação, em busca das metodologias de JS criadas nos últimos 10 anos e identificou qual delas suportam a participação de usuários (jogadores

³ Criação de protótipos de interface do usuário utilizando materiais de escritório [Muller 1991].

⁴ Sessões de Brainstorming através de desenhos organizado em forma de rodadas [Helander; Landauer e Prabhu 1997].

ou conteudistas) em alguma etapa do ciclo de vida de desenvolvimento do jogo. Neste levantamento não se encontrou nenhuma metodologia que contemplasse o jogador em todas as etapas do ciclo de vida do JS.

Observou-se que quase 76% das metodologias encontradas dão suporte à participação de usuário. As metodologias que dão algum suporte à participação do jogador geralmente se restringem a permitir essa interação apenas nas fases de avaliação do jogo, com alguns poucos casos envolvendo o jogador na etapa de análise de requisitos. Com base em relatos positivos do uso de design participativo (DP) na criação de JS, engajando os jogadores a utilizar o produto final e educando-os desde o início do processo, conclui-se que se faz necessária uma metodologia participativa para desenvolvimento de JS para dependentes químicos mas, devido ao contexto específico, esta participação deve ser encarada como recurso técnico e, principalmente, como recurso terapêutico de reinserção social.

O desenvolvimento de uma metodologia participativa para JS no contexto da dependência química é uma inovação que reflete direta e positivamente tanto para os dependentes envolvidos no processo quanto para os futuros usuários do jogo resultante, ampliando o impacto social do produto JS. Como trabalho futuro, serão levantadas quais sub-etapas do ciclo de vida de JS podem ser adaptadas para suportar a participação de jogadores e quais técnicas de DP podem ser aplicadas nestes pontos.

Agradecimento

O autor Marcelo da Silva Hounsell, gostaria de agradecer ao projeto REVIVER, convênio 2013 TR 00370-5 FAPESCxFAPEU.

Referências

- Buchinger, D., Cavalcanti, G. A. d. S., e Hounsell, M. d. S. (2014). Mecanismos de busca acadêmica: uma análise quantitativa. *Revista Brasileira de Computação Aplicada*, 6(1):108–120.
- Chin Jr, G. (2004). A case study in the participatory design of a collaborative science-based learning environment. PhD thesis, Virginia Polytechnic Institute and State University.
- Connolly, T. M., Boyle, E. A., MacArthur, E., Hainey, T., e Boyle, J. M. (2012). A systematic literature review of empirical evidence on computer games and serious games. *Computers & Education*, 59(2):661–686.
- De Lima, C. M. (2013). Governo do estado lança projeto de apoio e atenção a dependentes químicos. Disponível em: <<http://www.sc.gov.br/index.php/mais-sobre-desenvolvimento-social/2151-governo-do-estado-lanca-projeto-de-apoio-e-atencao-a-dependentes-quimicos/>>. Acessado em: 10 jun. 2014.
- Eagle, M. J. e Barnes, T. (2012). A learning objective focused methodology for the design and evaluation of game-based tutors. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*, pages 99–104. ACM.
- Franzwa, C., Tang, Y., e Johnson, A. (2013). Serious game design: Motivating students through a balance of fun and learning. In *Games and Virtual Worlds for Serious Applications (5th VS-GAMES)*, pages 1–7. IEEE.

- Helander, M. G., Landauer, T. K., e Prabhu, P. V. (1997). Handbook of human-computer interaction. Elsevier.
- Kitchenham, B. e Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE-2007-01.
- Lazarou, D. (2011). Using cultural-historical activity theory to design and evaluate an educational game in science education. *Journal of Computer Assisted Learning*, 27(5):424–439.
- Matsunaga, R. M., Bispo, D. M., e Borges, M. A. F. (2013). Processo de construção de jogos para aprendizado. *Jornada de Atualização em Informática na Educação*.
- Moser, C. (2013). Child-centered game development (CCGD): developing games with children at school. *Personal and ubiquitous computing*, 17(8):1647–1661.
- Muller, M. J. (1991). Pictive—an exploration in participatory design. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pgs 225–231. ACM.
- Navarrete, C. C. (2013). Creative thinking in digital game design and development: A case study. *Computers & Education*, 69:320–331.
- Nielsen, J. (1993). *Usability Engineering*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Petersen, K., Feldt, R., Mujtaba, S., e Mattsson, M. (2008). Systematic mapping studies in software engineering. In *12th International Conference on Evaluation and Assessment in Software Engineering*, volume 17, page 1.
- Preece, J., Rogers, Y., e Sharp, H. (2002). *Interaction design: beyond human-computer interaction*. John Wiley & Sons.
- Rankin, Y. A., McNeal, M., Shute, M. W., e Gooch, B. (2008). User centered game design: evaluating massive multiplayer online role playing games for second language acquisition. In *Proceedings of the 2008 ACM SIGGRAPH symposium on Video Games*, pages 43–49. ACM.
- Ribeiro, M. e Veloso, A. (2012). Explogador: Serious games e design colaborativo. *VIDEOJOGOS 2012*, page 63.
- Ruppel, U. e Schatz, K. (2011). Designing a bim-based serious game for fire safety evacuation simulations. *Advanced Engineering Informatics*, 25(4):600–611.
- Sevilla, C., Santini, S., Haya, P., Rodriguez, P., e Sacha, G. (2012). Interdisciplinary design of videogames: A highly motivating method of learning. In *Computers in Education (SIIE), 2012 International Symposium on*, pages 1–6. IEEE.
- Sommerville, I. (2011). *Software engineering*. Addison-Wesley, 9th edition.