

ENgAGED: Um Processo de Desenvolvimento de Jogos para Ensinar Computação

Paulo E. Battistella¹, Christiane G. von Wangenheim¹

¹Grupo de Qualidade de Software (GQS) – Departamento de Informática e Estatística (INE) – Universidade Federal de Santa Catarina (UFSC) – Florianópolis – SC – Brasil

paulo@incod.ufsc.br, c.wangenheim@ufsc.br

***Abstract.** The use of educational games for teaching computing has been shown to be an effective instructional strategy that leads to active learning and entertaining. In this context, the article presents ENgAGED - a process for systematically developing educational games. The process is based on the integration of existing literature on instructional and game design. The process is directed to teachers or students who develop games for computer education graduation. The process has been evaluated has been confirmed by an expert panel. In general, experts consider the process unambiguous, consistent, complete, understandable, accurate, flexible, helpful and good usability.*

***Resumo.** O uso de jogos educacionais para o ensino de computação vem se mostrando uma estratégia instrucional eficaz, que conduz os alunos à uma aprendizagem ativa e divertida. Neste contexto, o artigo apresenta ENgAGED – um processo para desenvolver sistematicamente jogos educacionais. O processo baseia-se na integração da literatura existente com design instrucional e design de jogos. O processo é direcionado para professores ou alunos que desenvolvem jogos para o ensino de computação na graduação. O processo foi avaliado por um painel de especialistas. De modo geral, os especialistas consideraram o processo não ambíguo, consistente, completo, compreensível, correto, flexível, útil e com boa usabilidade.*

1. Introdução

Os cursos de computação em instituições de ensino superior são formados por uma complexa matriz curricular, estando organizados em diversas áreas de conhecimento, como Algoritmos e Complexidade, Engenharia de Software, Interação Humano-Computador, Linguagem de Programação e Sistemas Operacionais [ACM/IEEE-CS 2013]. Tipicamente, a estratégia instrucional predominante de ensino nestes cursos são aulas expositivas [ACM/IEEE-CS 2013]. A aula expositiva é uma estratégia adequada para apresentar conceitos abstratos e informações factuais para um grande grupo de alunos [Ruutmann e Hants 2011]. No entanto, outras estratégias (como simulação, laboratórios, análise de casos) trazem benefícios ao ensino, levando os alunos a níveis de aprendizagem mais elevados e possibilitando a aplicação do conhecimento em situações práticas [Choi e Hannafin 1995; Poulova e Klimova 2015].

Neste contexto, jogos são considerados poderosas estratégias instrucionais possibilitando diversos benefícios, aumentando a eficácia da aprendizagem, interesse,

motivação e persistência dos alunos [Kirriemur e McFarlane 2004; Freitas 2006; Mitchell e Savill-Smith 2004; Prensky 2007]. Estes jogos também promovem a aprendizagem ativa, permitindo alcançar um aprendizado mais profundo, em um tempo de ensino aceitável e sem sobrecarregar o professor [Wangenheim e Shull 2009]. Além disso, podem servir como meio de entretenimento para aprofundar e praticar conceitos, bem como ilustrar dinâmicas ou princípios abstratos [Hays e Singer 1988]. Devido às suas características intrínsecas, tais como, competição, desafio e interação, eles podem transformar o aprendizado em uma experiência envolvente e divertida [Kafai 2001]. Atualmente, existem diversos jogos para ensinar computação [Battistella e Wangenheim 2016; Boyle et al. 2016], incluindo, por exemplo, *C-Jump* [Singh, Dorairaj e Woods 2007] ou *Saving Princess Sera* [Eagle e Barnes 2009] para ensinar programação, *SORTIA* [Battistella, Wangenheim e Wangenheim 2012] para ensinar estrutura de dados, *SimSE* [Navarro e Hoek 2007] e *Lidar com pessoas difíceis* [Wangenheim, Carvalho e Battistella 2013] para ensinar engenharia de software, *Anti-Phishing Phil* [Sheng et al. 2007] na área de segurança, entre outros.

Atualmente é difícil afirmar que os jogos educacionais disponíveis apresentem equilíbrio entre o design instrucional e o design de jogos [Wangenheim e Shull 2009]. Observa-se também um grande aumento da utilização de jogos educacionais nos últimos anos, no entanto, não há evidências de que atinjam a qualidade desejada, ou então que sejam realmente divertidos e/ou ensinam conforme espera-se [Battistella e Wangenheim 2016; Boyle et al. 2016]. Também se evidencia que os jogos desenvolvidos com os processos atuais podem não contribuir para o desenvolvimento de pensamento lógico, cognitivo e social dos alunos [Mari e Nakamura 2015].

Uma forma de melhorar a qualidade dos jogos educacionais é utilizar um processo sistemático de desenvolvimento. Porém atualmente não existem processos abrangentes orientando o desenvolvimento de jogos para o ensino de computação [Coelho, Reinehr e Malucelli 2015; Battistella, Wangenheim e Fernandes 2014]. Observa-se também que os processos existentes, ou focam no processo de desenvolvimento de software [Coelho, Reinehr e Malucelli 2015] ou em aspectos do design instrucional, não havendo um equilíbrio entre design instrucional e design de jogos [Battistella, Wangenheim e Fernandes 2014].

Deste modo, se evidencia a necessidade de processos sistemáticos guiando o desenvolvimento de jogos educacionais que incluem tanto, aspectos do design instrucional [Branch, 2009], quanto do design de jogos [Fullerton, 2008, Schell, 2010] – para resultar em jogos eficazes em termos de aprendizagem, mas também divertidos. Neste contexto, o presente artigo tem por objetivo apresentar um processo de desenvolvimento de jogos educacionais para o ensino superior de computação – ENgAGED (*EducatioNAl GamEs Development*) – integrando design instrucional e design de jogos.

2. Metodologia de Pesquisa

Para o desenvolvimento do processo foi realizada uma pesquisa exploratória, que tem como objetivo o aprimoramento de ideias ou a descoberta de novas hipóteses, utilizando levantamento bibliográfico, análise de exemplos que “*estimulam a compreensão*” e

entrevistas com pessoas com experiência prática domínio do problema [Gil 2010]. Neste sentido, a pesquisa é dividida 3 etapas (Figura 1).

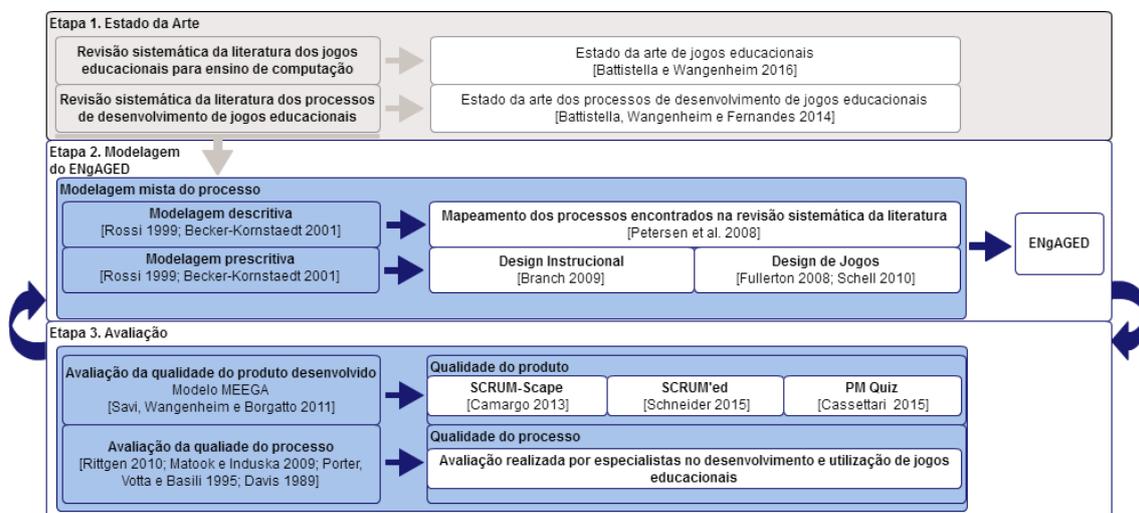


Figura 1. Metodologia de pesquisa

Inicialmente foi feito um levantamento do estado da arte por meio de uma revisão sistemática da literatura [Kitchenham 2004] com o objetivo de identificar como os jogos são desenvolvidos e avaliados, e quais processos/modelos/*frameworks* existem para o desenvolvimento de jogos educacionais. Foi observado que a maioria dos jogos são desenvolvidos de modo *ad-hoc* e geralmente são superficialmente avaliados em relação ao seu impacto no ensino [Battistella e Wangenheim 2016]. Em termos de processos, foram identificadas lacunas tanto em relação à falta de equilíbrio entre o design instrucional e o design de jogos, quanto no detalhamento das fases e atividades [Battistella, Wangenheim e Fernandes 2014].

Com base no levantamento do estado da arte, o processo ENgAGED foi concebido a partir de uma modelagem mista (descritiva e prescritiva) [Rossi 1999; Becker-Kornstaedt 2001]. Inicialmente foi realizada uma modelagem descritiva a partir de um mapeamento dos processos encontrados na literatura seguindo o procedimento proposto por [Petersen et al. 2008]. Complementando esta etapa foi realizada uma modelagem prescritiva, com base no modelo de design instrucional ADDIE [Branch 2009] e em processos do design de jogos [Fullerton 2008, Schell 2010].

O processo foi avaliado para obter um *feedback* inicial. Esta avaliação tem por objetivo *avaliar a qualidade do processo do ponto de vista dos especialistas (professores e alunos) que desenvolveram jogos educacionais, ou professores que utilizam jogos educacionais em sala de aula na área da computação*. Para a definição da medição na avaliação é utilizado o GQM (*Goal, Question, Metric*) [Basili, Caldeira e Rombach 1994] definido as questões de análise, medidas e um questionário para a coleta de *feedback* dos especialistas. As questões de análise e medidas são criadas com base nas seguintes características que estão relacionadas à qualidade de um processo [Rittgen 2010; Matoon e Induska 2009; Porter, Votta e Basili 1995; Davis 1989]: ambiguidade, consistência, completude, compreensibilidade, corretude, flexibilidade, usabilidade e utilidade.

O processo também foi avaliado em relação a qualidade do produto, para isso foi utilizado o modelo MEEGA [Savi, Wangenheim e Borgatto 2011] que avalia os jogos educacionais em termos de motivação, experiência do usuário e aprendizagem. Os respectivos resultados são apresentados como parte dos estudos de caso que utilizaram versões intermediárias do ENgAGED no desenvolvimento dos jogos SCRUM-Scape [Camargo 2013], SCRUM'ed [Schneider 2015] e PM Quiz [Cassettari 2015].

3. ENgAGED - Processo de Desenvolvimento de Jogos Educacionais

O ENgAGED (*EducatioNAl GamEs Development*) é um processo iterativo de desenvolvimento de jogos para o ensino de computação. Com objetivo de equilibrar os aspectos do design instrucional e do design de jogos, ele é formado por fases e atividades dos processos de desenvolvimento de jogos educacionais disponíveis na literatura (modelagem descritiva) [Battistella, Wangenheim e Fernandes 2014], e por modelos de referência tanto na área de design instrucional [Branch 2009] quanto no design de jogos [Fullerton 2008, Schell 2010] (modelagem prescritiva) (Figura 2).

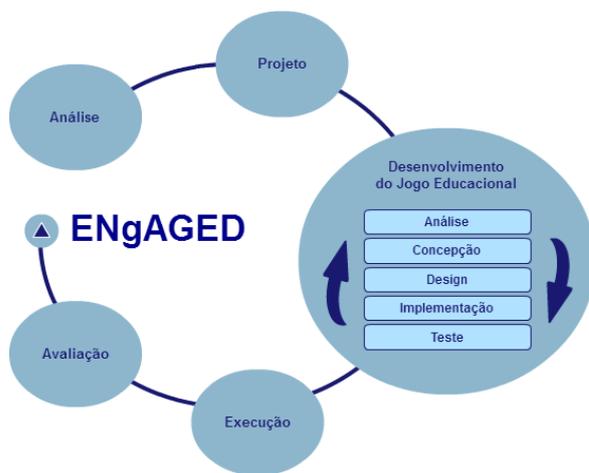


Figura 2. Visão geral do ENgAGED

O processo fundamenta-se no design instrucional ADDIE [Branch 2009] permitindo a produção de uma Unidade Instrucional (UI) envolvendo as fases de análise, projeto, desenvolvimento, execução e avaliação. Neste contexto, uma UI é o conjunto de atividades organizadas em tópicos ou temas principais, sendo apresentadas ao longo de uma disciplina [Hurwitz e Day 2007; Wisconsin 2012]. No caso do processo ENgAGED, a especificação da UI é exclusivamente para o ensino via jogos. A Tabela 1 apresenta uma visão geral das fases e atividades do processo.

O processo é relevante, porque oferecer suporte ao desenvolvimento sistemático de jogos educacionais, contendo fases e atividades criadas com base na literatura do design instrucional, design de jogos e em processos de desenvolvimentos de jogos educacionais disponíveis na literatura. O processo é direcionado a professores e alunos (orientados por professores) dos cursos de graduação em computação.

Tabela 1. Descrição do ENgAGED

Fase 1. Análise da Unidade Instrucional (UI)	
A1.1 Especificar UI do jogo	Especificar o contexto da UI, como curso, disciplina, pré-requisitos da disciplina, objetivos de aprendizagem e conteúdo programático.
A1.2 Caracterizar aprendizes	Caracterizar o público-alvo (alunos), p. ex. em termos de faixa etária, preferências de gênero de jogos, modo de interação, jogos favoritos etc. Caracterizar o ambiente, definindo a infraestrutura disponível para aplicação do jogo.
A1.3 Definir objetivo(s) de desempenho	Definir o(s) objetivo(s) que avaliam o desempenho do aluno ao final da UI. Ele(s) oferece(m) uma direção sobre a condução do conteúdo ao longo da UI.
Fase 2. Projeto da Unidade Instrucional (UI)	
A2.1 Definir avaliação do aluno	Definir como será estruturada a avaliação, para o aluno aprender com o jogo. Tipicamente a avaliação é inserida no próprio jogo, por meio de regras e também apresentando <i>feedbacks</i> quando o aluno acerta ou

	erra um determinado conceito.
A2.2 Definir conteúdo da estratégia instrucional	A estratégia instrucional é <i>Jogo Educacional</i> . Assim, nesta atividade define-se o conteúdo e o sequenciamento do conteúdo ao longo do jogo.
A2.3 Decidir pelo desenvolvimento ou utilizar jogo desenvolvido	Decidir pelo desenvolvimento de um jogo educacional, ou pela utilização de um jogo existente. Caso a opção seja o uso de um jogo existente, a atividade orienta a busca por jogos disponíveis. Caso a opção seja pelo desenvolvimento é necessário seguir para <i>Fase 3 – Desenvolvimento do Jogo Educacional</i> .
A2.4 Revisar o modelo de avaliação do jogo	Revisar o modelo utilizado para avaliar o jogo educacional. Como padrão propõem-se o modelo de avaliação de jogos educacionais MEEGA [Savi, Wangenheim e Borgatto 2011].
Fase 3. Desenvolvimento do Jogo Educacional	
Fase 3.1. Análise do Jogo	
A3.1.1 Levantar requisitos do jogo	Levantar os requisitos para identificação das funções e funcionalidades do jogo. Nesta atividade deve-se também definir como o conteúdo da UI, definidos na fase de projeto, serão distribuídos nos níveis do jogo.
Fase 3.2. Concepção do Jogo	
A3.2.1 Conceber o jogo	Conceber o jogo, descrevendo as principais características, como objetivos do jogo, narrativa, regras, mecânica, elementos do jogo, pontuações e <i>feedback</i> educacional.
Fase 3.3. Design do Jogo	
	Jogo digital
	Jogo não-digital
A3.3.1 Definir linguagem de programação ou <i>game engine</i>	Definir a linguagem de programação ou <i>game engine</i> que será utilizada para o desenvolvimento do jogo. Nesta atividade são realizados testes empíricos com linguagens ou <i>engines</i> a fim de decidir qual será realmente utilizada para o desenvolvimento do jogo.
A3.3.2 Produzir ilustrações ou imagens dos elementos do jogo	Produzir as ilustrações que representam os elementos do jogo. Normalmente os elementos são personagens, cenários, objetos, artefatos, menus ou janelas de opções/configurações do jogo.
A3.3.3 Modelar o jogo	Modelar os níveis do jogo, as bibliotecas adicionadas à <i>game engine</i> ou à linguagem de programação, os <i>feedbacks</i> educacionais e os diálogos dos personagens.
Fase 3.4. Implementação do Jogo	
A3.4.1 Produzir elementos do jogo	Produzir os elementos criados na atividade <i>A3.3.2 Produzir ilustrações ou imagens dos elementos do jogo</i> .
	Jogo digital
	Jogo não-digital
	Codificação ou programação dos cenários, personagens, artefatos e objetos do jogo por meio de linguagem de programação ou <i>game engine</i> .
	Produção dos elementos não-digitais.
Fase 3.5. Testes do Jogo	
A3.5.1 Realizar testes do jogo	Realizar testes para detecção de erros e <i>feedbacks</i> para melhoria do jogo. Os testes são realizados pelos criadores do jogo e especialistas (conhecedores do domínio de aplicação), permitindo assim testar os níveis e funcionalidades do jogo, além de verificar os possíveis problemas antes de sua execução (Fase 4).
Fase 4. Execução da Unidade Instrucional (UI)	
A4.1 Planejar a execução do jogo	Planejar a execução do jogo definindo data para jogar, local aonde será jogado, e equipamentos que serão utilizados ou materiais que devem ser impressos.
A4.2 Instalar o jogo digital	Jogos digitais
	Instalar o jogo conforme a sua plataforma. Caso seja um jogo com arquitetura cliente-servidor, é necessário instalá-lo no servidor e disponibilizá-lo para acesso via internet. Caso seja <i>stand-alone</i> é necessário disponibilizá-lo nos computadores que os alunos utilizarão no momento da execução do jogo. Caso seja um jogo de <i>smartphone</i> deve-se disponibilizá-lo para <i>download</i> .
	Jogos não-digitais
	Não se aplica.
A4.3 Executar o jogo	Executar o jogo em sala de aula, laboratório ou extraclasse.
Fase 5. Avaliação da Unidade Instrucional (UI)	
A5.1 Conduzir avaliação	Conduzir a avaliação após a execução do jogo utilizando o instrumento para coleta de dados definidos na atividade <i>A2.4 Revisar o modelo de avaliação do jogo</i> .
A5.2 Analisar dados da avaliação	Realizar a análise dos dados coletados por meio de estatística descritiva, respondendo as questões de análise definidas na atividade <i>A2.4 Revisar o modelo de avaliação do jogo</i> .

Todas as atividades que constituem o processo são detalhadas por uma descrição das tarefas, os produtos de entrada e saída. Um exemplo deste detalhamento é apresentado na Tabela 2. A descrição detalhada completa é apresentada em um relatório técnico [Battistella e Wangenheim 2015].

Tabela 2. Atividade de definição do(s) objetivo(s) de desempenho

Atividade	A1.3 Definir objetivo(s) de desempenho
-----------	--

Descrição	Esta atividade tem por objetivo definir o(s) objetivo(s) que avaliam o desempenho do aluno ao final da UI. Ele(s) oferece(m) uma direção sobre a condução do conteúdo ao longo do design da UI e oferece <i>feedback</i> do aprendizado dos alunos. Por meio dos objetivos de desempenho é possível avaliar se os alunos adquiriram as competências necessárias para aprendizagem.	
Tarefa	1.3 Definir objetivos de desempenho Nesta tarefa identifica-se os objetivos de desempenho da UI, a qual é formada por três componentes: condição, desempenho (p. ex. baseado na taxonomia de Bloom (1956)), e critério. O objetivo de desempenho, tipicamente é definido por desempenho + condição + critério . O desempenho é a descrição do que o aluno aprendeu, demonstrando domínio daquele objetivo proposto. A condição descreve as circunstâncias que o desempenho será realizado. O critério define o grau de acerto do desempenho, sendo considerado suficientemente aceitável. Na criação de um desempenho é possível utilizar a taxonomia de Bloom (1956), a qual é dividida em três níveis: cognitivo, psicomotor e afetivo.	
Produtos de trabalho	Produto de entrada: P1.1.1 - Plano de ensino da disciplina; P1.2.2 - Caracterização dos aprendizes.	
	Produto de saída: P1.3.1 - Objetivo(s) de desempenho	
	Objetivo(s) de desempenho	Listagem de um ou mais objetivos de desempenho, formado pela seguinte estrutura: condição + desempenho + critério.

O processo ENgAGED também propõem *templates* para todos os produtos de saída, como no exemplo apresentado na Tabela 3.

Tabela 3. Produto de saída da atividade produzir elementos do jogo SCRUM'ed (extraído da descrição do jogo SCRUM'ed [Battistella, Schneider e Wangenheim 2015])

Produto de saída: P3.4.1 - Elementos do jogo implementados			
Jogo Digital			
Versão do código-fonte	SCRUMed b20150520_Data		
Endereço eletrônico	https://marcelofs.com/scrumed/		
Emento do jogo: Personagens			
Nome	Papel	Imagem	
Jogador	SCRUM <i>Master</i>		
...	
Emento do jogo: Cenários			
Nome	Nível	Sequência	Imagem
Palácio	2	1	
...

4. Avaliação Inicial do Processo ENgAGED

Com objetivo de avaliar a qualidade do processo ENgAGED foi realizada uma avaliação do processo.

4.1. Definição da avaliação

A avaliação tem por objetivo *avaliar a qualidade do processo ENgAGED do ponto de vista do criador de jogos*. O conceito de qualidade do processo foi decomposto com base na literatura [Rittgen 2010; Matook e Induska 2009; Porter, Votta e Basili 1995; Davis 1989] em questões de análise conforme apresentado na Tabela 4.

Tabela 4. Questões de análise e medidas da avaliação

Questões de análise	Medidas
QA1) A descrição do processo é ambígua?	M1.1) Quantidade de descrições ambíguas.
QA 2) O processo é consistente?	M2.1) Quantidade de inconsistência por fases, atividades e produtos de trabalho.
QA 3) O processo é completo?	M3.1) Quantidade de fases, atividades e produtos de trabalho faltando.
QA 4) O processo é compreensível?	M4.1) Quantidade de descrições incompreensíveis por fases, atividade e produtos de trabalho.
QA 5) O processo é correto?	M5.1) Quantidade de erros no processo por fases, atividade e produtos de trabalho. M5.2) Erros corrigidos.
QA 6) O processo é aplicável para o desenvolvimento de jogos digitais e não-digitais?	M6. 1) Quantidade de jogos digitais e não-digitais desenvolvidos pelo processo. M6.2) Quantidade de gêneros de jogos desenvolvidos pelo processo. M6.3) Quantidade de descrições das fases, atividades e produtos de trabalho que foram modificados para cada tipo específico de jogo. M6.4) Grau de facilidade para adaptação do processo para um determinado jogo.
QA 7) O processo é fácil de usar?	M7.1) Grau de usabilidade do processo.
QA 8) O processo é útil?	M8.1) Grau de utilidade do processo.

A avaliação foi realizada por um painel de especialistas, composto por professores e alunos da área da computação que desenvolvem jogos educacionais e/ou professores que aplicam jogos educacionais em sala de aula. Deste modo, a avaliação por meio de um painel de especialistas permite que profissionais experientes na área se coloquem na posição de *criadores de jogos* para analisar a qualidade do processo. Para a coleta de dados foi criado um questionário conforme as questões de análise e medidas apresentadas na Tabela 4.

4.2. Execução da avaliação

A avaliação foi realizada entre outubro de 2014 a março de 2015. Foram convidados 50 professores e alunos que desenvolvem ou utilizam jogos educacionais no Brasil. Para realizar a avaliação foi entregue aos especialistas o relatório técnico do ENgAGED [Battistella e Wangenheim 2015] apresentando o processo em detalhe junto com o questionário de avaliação. Ao final 10 especialistas responderam o questionário, sendo que 9 avaliaram completamente o processo e 1 parcialmente. Para a análise dos resultados foram consideradas as 10 avaliações. Entre os especialistas, 6 já desenvolveram jogos, 4 são autores de artigos sobre jogos educacionais. Todos são da área da computação.

4.3. Análise dos dados

De modo geral, o resultado da avaliação indica que o processo apresenta boa qualidade, do ponto de vista da não ambiguidade, consistência, completude, compreensibilidade, corretude, flexibilidade, utilidade e usabilidade.

A maioria dos especialistas não identificaram ambiguidades, exceto dois especialistas que sugeriram maior clareza na definição da atividade *A1.1 Especificar unidade instrucional do jogo*. Um especialista também sugeriu uma melhor distinção das atividades *A3.1.1 Levantar requisitos do jogo* e *A3.2.1 Conceber o jogo*. Não foram identificados problemas de consistência, corretude e compreensibilidade. Os especialistas também consideram o processo flexível para cobrir o desenvolvimento de diferentes tipos de jogos. O processo também foi considerado completo, sem necessidade de acréscimo de fases, atividades ou produtos de trabalhos faltantes. Dois especialistas sugeriram distinguir as atividades para os jogos digitais e não-digitais, e também deixar clara a iteratividade das atividades específicas do design instrucional. Em termos de utilidade, a maioria dos especialistas concorda que o processo pode

permitir o desenvolvimento de jogos mais rapidamente, e pode ter efeitos positivos como a melhora o desempenho no desenvolvimento e aumento da produtividade no desenvolvimento. Em termos de usabilidade, a maioria dos especialistas considerou o processo fácil de aprender a usar, fácil de obter informações para desenvolver, claro e compreensível. O resultado desta avaliação inicial realizada por um painel de especialistas fornece uma primeira indicação da qualidade do processo ENgAGED.

4.4. Ameaças à validade da pesquisa

Os resultados obtidos nesta avaliação precisam ser interpretados com cuidado, levando em consideração as ameaças à sua validade. Entre os fatores que a ameaçam, destaca-se o impacto na validade dos resultados, principalmente pela generalização destes resultados. Outra ameaça está relacionada ao tamanho relativamente pequeno da amostra, considerando a quantidade de especialistas. No entanto, mesmo com uma amostra relativamente pequena é possível considerar os dados analisados relevantes. Outro fator que pode ser considerado ameaça, está relacionado à experiência dos avaliadores na área de jogos educacionais. Para minimizar esta ameaça foi selecionado um painel de especialistas que já desenvolveram ou utilizam jogos educacionais na área da computação. Outra questão principal é a realização de avaliação via um painel de especialistas sem aplicação do processo de fato. Porém, para minimizar esta ameaça, foram realizados três estudos de caso em paralelo à modelagem do processo ENgAGED, sendo desenvolvido os jogos SCRUM-Scape [Camargo 2013], SCRUM'ed [Schneider 2015] e PM Quiz [Cassettari 2015]. Estes estudos de caso confirmam a qualidade dos produtos desenvolvidos pelo processo apresentado no presente artigo.

7. Conclusão

O artigo apresenta o ENgAGED, um processo para o desenvolvimento de jogos para o ensino de computação. Com a intenção de facilitar o desenvolvimento de jogos para ensinar computação ele integra o design instrucional e de jogos, buscando equilíbrio entres ambos os aspectos. Uma primeira avaliação inicial, indica que o processo pode ser útil, sendo considerado consistente, completo, compreensível e correto pelos especialistas. Também foi possível obter uma indicação que ele é flexível para cobrir o desenvolvimento de vários tipos de jogos, também com base em estudos de caso realizado aplicando o processo ao longo da sua modelagem. Assim, o processo pode ser uma opção para guiar o desenvolvimento de jogos educacionais por professores e alunos da área da computação.

Agradecimentos

Este trabalho foi apoiado pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), uma entidade do governo brasileiro focada no desenvolvimento científico.

Referências

- ACM/IEEE-CS. (2013) “Computer Science Curricula 2013”, Estados Unidos da América: ACM e IEEE Computer Society, p. 514.
- Basili, V.; Caldeira, G. e Rombach, H. D. (1994) “Goal Question Metric Approach”. Encyclopedia of Software Engineering, p. 528-532, John Wiley & Sons, Inc.

- Battistella, P. E. e Wangenheim, C. G. von, (2016) “Games for Teaching Computing in Higher Education – A Systematic Review”, IEEE Technology and Engineering Education.
- Battistella, P. E. e Wangenheim, C. G. von, (2015) “ENgAGED: Processo de Desenvolvimento de Jogos para Ensino em Computação”, Relatório Técnico INCoD/GQS.01.2015.P, INCoD/INE/UFSC, Florianópolis, Brasil.
- Battistella, P. E.; Schneider M. F. e Wangenheim, C. G. von, (2015) “Aplicação do Processo ENgAGED: Desenvolvimento do Jogo Educacional SCRUM’ed”, Relatório Técnico INCoD/GQS.02.2015.P, INCoD/INE/UFSC, Florianópolis, Brasil.
- Battistella, P. E.; Wangenheim, C. G. von e Fernandes, J. (2014) “Como jogos educacionais são desenvolvidos? Uma revisão sistemática da literatura”, Workshop sobre Educação em Computação, Brasília, Brasil.
- Battistella, P. E.; Wangenheim, A. von e Wangenheim, C. G. von. (2012) “SORTIA - Um jogo para ensino de algoritmo de ordenação”, Anais do 23º Simpósio Brasileiro de Informática na Educação. Rio de Janeiro, Brasil.
- Becker-Kornstaedt, U. (2001) “Towards systematic knowledge elicitation for descriptive software process modeling”, IESE Report n° 036.01/E, IESE, Kaiserslautern, Alemanha.
- Bloom, B. S. (1956) “Taxonomy of educational objectives: the classification of educational goals”, Handbook I, cognitive domain.
- Boyle, E. A. et al. (2016) “An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and Educational Games”, Computers & Education, 94, p. 178-192.
- Branch, R. M. (2009) “Instructional Design: The ADDIE Approach”, Springer, New York, EUA, p. 203.
- Camargo, A. (2013) “Jogo de RPG para ensinar SCRUM”, 62 f. TCC (Graduação) - Ciências da Computação, Departamento de Informática e Estatística, UFSC, Florianópolis, Brasil.
- Cassettari, F. T. (2015) “Customização de um Jogo Educacional para Revisão de Conhecimento em Gerenciamento de Projetos”, 108 f. TCC (Graduação) - Curso de Sistemas de Informação, Departamento de Informática e Estatística, UFSC, Florianópolis, Brasil.
- Choi, J. e Hannafin, M. (1995) “Situated Cognition and Learning Environments: Roles, Structures and Implications for Design”, Educational Technology Research and Development, 43(2), p. 53-69.
- Coelho Neto, J.; Reinehr, S. e Malucelli, A. (2015) “Processo de Desenvolvimento de Software: uma Análise Exploratória com Profissionais que Desenvolvem Jogos Eletrônicos Educacionais”, Anais do Simpósio Brasileiro de Informática na Educação, Maceió, Brasil.
- Davis, F. D. (1989) “Perceived usefulness, perceived ease of use, and user acceptance of information technology”, MIS Quarterly, 13(3), p. 319-340.
- Eagle, M. e Barnes, T. (2009) “Experimental evaluation of an educational game for improved learning in introductory computing”, Proc. of the 40th Technical Symposium on Computer Science Education, Chattanooga, EUA.
- Freitas, S. I. de. (2006) “Using games and simulations for supporting learning”, Learning, Media and Technology, Digital Games and Learning, v. 31, n. 4, p. 343-358.
- Fullerton, T. (2008) “Game design workshop: A playcentric approach to creating innovatives games”, 3. ed. Burlington: Morgan Kaufmann, p. 470.
- Gil, A. C. (2010) “Como elaborar projetos de pesquisa”, 5ª ed. São Paulo: Atlas, 184 p.
- Hays, R. T. e Singer, M. J. (1988) “Simulation fidelity in training system design: Bridging the gap between reality and training”, Recent Research in Psychology, Springer, p. 415.
- Hurwitz, A e Day, M. (2007) “Children and their art: Methods for the elementary school”, Eighth Edition. Belmont: Thomson Wadsworth, CA, EUA.

- Kafai, Y. B. (2001) "The educational potential of electronic games: From games-to-teach to games-to-learn", Conference on playing by the rules: the cultural policy challenges of video games. Chicago, EUA.
- Kirriemuir, J. e McFarlane, C. A. (2004) "Literature Review in Games and Learning Literature Review in Games and Learning", Bristol: FutureLab Series.
- Kitchenham, B. (2004) "Procedures for performing systematic reviews", Tech. Report TR/SE-0401, Keel University, Keel, UK.
- Matook, S, e Indulska, M. (2009) "Improving the quality of process reference models: A quality function deployment-based approach", *Decis. Support Syst.* v. 47, n. 1, p. 60-71.
- Mitchell, A. e Savill-Smith, C. (2004) "The use of computer and video games for learning: a review of the literature", Learning and Skills Development Agency, Ultralab, London, UK.
- Peterson, K. al. (2008) "Systematic Mapping Studies in Software Engineering", *Proc. of 12th Int. Conf. on Evaluation and Assessment in Software Engineering* Swinton, UK, p. 68-77.
- Porter, A. A.; Votta, L. G. Jr. e Basili, V. R. (1995) "Comparing detection methods for software requirements inspections: a replicated experiment", *IEEE Transactions in Software Engineering*, v.21, n.6, p. 563-575.
- Poulova, P. e Klimova, B. (2015) "Education in Computational Sciences", *International Conference on Computational Science*. Reykjavík, Iceland, p. 1996-2005.
- Prensky, M. (2007) "Digital game-based learning", Paragon House, p. 464.
- Rittgen, P. (2010) "Quality and perceived usefulness of process models", *Proceedings of the ACM Symposium on Applied Computing*. Nova York, EUA, p. 65-72.
- Ruutmann, T. e Hants K. (2011) "Teaching Strategies for Direct and Indirect Instruction in Teaching Engineering", *Anais Int. Conference on Interactive Collaborative Learning*, Piestany, Slovakia.
- Rossi, S. (1999) "Moving Towards Modelling Oriented Software Process Engineering: A Shift from Descriptive to Prescriptive Process Modeling", *Anais da Conf. Product Focused Improvement of Embedded Software Processes*, Oulu, Finlândia, p. 508 - 522.
- Savi, R.; Wangenheim C. G. von e Borgatto, A. (2011) "Um Modelo de Avaliação de Jogos Educacionais na Engenharia de Software", *Anais do Simpósio Brasileiro de Engenharia de Software*, São Paulo, Brasil.
- Schell, J. (2010) "The art of game design: A book of lenses", Elsevier.
- Schneider, M. F. (2015) "SCRUM'ed: um jogo de RPG para ensinar Scrum", 66 f. TCC (Graduação) - Curso de Sistemas de Informação, Departamento de Informática e Estatística, UFSC, Florianópolis, Brasil.
- Sheng, S. et al. (2007) "Anti-Phishing Phil : the design and evaluation of a game that teaches people not to fall for phish", *Proc. of third Symposium on Usable Privacy and Security*, Pittsburgh, EUA.
- Singh, J.; Dorairaj, S. K. e Woods, P. (2007) "Learning computer programming using a board game – case study on C-Jump", *Proc. of the Int. Symposium on Information and Communications Technologies*, Kuala Lumpur, Malaysia.
- Wangenheim, C. G. von; Carvalho, O. P. e Battistella, P. E. (2013) "Ensinar a Gerência de Equipes em Disciplinas de Gerência de Projetos de Software", *RBIE - Revista Brasileira de Informática na Educação*, 20(1), p. 16 - 22.
- Wangenheim, C. G. von e Shull, F. (2009) "To game or not to game?", *IEEE Software*, v. 26, n. 2, p. 92-94.
- Wisconsin. (2012) "Curricular e course (re)design toolkit", University of Wisconsin-Madison, Madison, EUA.