

Paulo Eduardo Battistella

**ENgAGED: Um processo de desenvolvimento de jogos para ensino em computação**

Tese submetida ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa Catarina para a obtenção do Grau de Doutor em Ciência da Computação.

Orientadora: Prof<sup>a</sup>. Dr<sup>a</sup>. rer. nat.  
Christiane Gresse von Wangenheim,  
PMP

Florianópolis  
2016

Ficha de identificação da obra elaborada pelo autor através, do  
Programa de Geração Automática da Biblioteca Universitária da UFSC.

Battistella, Paulo Eduardo

ENgAGED: Um processo de desenvolvimento de jogos para  
ensino em computação / Paulo Eduardo Battistella ;  
orientadora, Christiane Gresse von Wangenheim -  
Florianópolis, SC, 2016.  
401 p.

Tese (doutorado) - Universidade Federal de Santa  
Catarina, Centro Tecnológico. Programa de Pós-Graduação em  
Ciência da Computação.

Inclui referências


1. Ciência da Computação. 2. Jogos Educacionais. 3.  
Design Instrucional. 4. Design de Jogos. I. Wangenheim,  
Christiane Gresse von. II. Universidade Federal de Santa  
Catarina. Programa de Pós-Graduação em Ciência da Computação.  
III. Título. |

Paulo Eduardo Battistella

**ENgAGED: Um processo de desenvolvimento de jogos para ensino em computação**

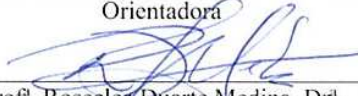
Esta tese foi julgada adequada para obtenção do título de doutor e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.


Florianópolis, 30 de novembro de 2016.


  
\_\_\_\_\_  
Prof.<sup>a</sup> Carina Friedrich Dorneles, Dr.<sup>a</sup>.  
Coordenadora do Programa

**Banca Examinadora:**

  
\_\_\_\_\_  
Prof.<sup>a</sup> Christiane Gresse von Wangenheim, Dr.<sup>a</sup>. rer. nat.  
Universidade Federal de Santa Catarina  
Orientadora

  
\_\_\_\_\_  
Prof.<sup>a</sup> Roseclea Duarte Medina, Dr.<sup>a</sup>.  
Universidade Federal de Santa Maria  
(Videoconferência)

  
\_\_\_\_\_  
Prof. Marcello Thiry Comicholi da Costa, Dr.  
Universidade do Vale do Itajai

  
\_\_\_\_\_  
Prof. Eros Comunello, Dr.  
Universidade do Vale do Itajai



---

Prof. Raul Sidnei Wazlawick, Dr.  
Universidade Federal de Santa Catarina



---

Prof. Ricardo Azambuja Silveira, Dr.  
Universidade Federal de Santa Catarina

Este trabalho é dedicado aos meus colegas de pesquisa, aos meus familiares e à minha companheira.



## AGRADECIMENTOS

À minha querida esposa *Bruna Rafaela Moeller* pelo seu amor, afeto, incentivo e paciência a mim dedicados ao longo deste trabalho.

Aos meus pais *Ary Battistella* e *Josina Santana Battistella* que sempre contribuíram para meu desenvolvimento pessoal e profissional.

Aos meus irmãos *Saray Battistella* e *Sandro Battistella* pelo convívio e incentivo em todas as fases da minha vida.

À minha orientadora Prof<sup>a</sup>. *Christiane Gresse von Wangenheim* pela oportunidade, inspirações, profissionalismo e apoio durante todo este trabalho.

Ao prof. *Aldo von Wangenheim*, coordenador do Instituto Internacional para Convergência Digital – INCoD/UFSC, pela oportunidade de participar dos projetos de pesquisa.

Aos colegas do Grupo de Qualidade de Software – GQS/UFSC e a todos os colegas do Instituto Nacional para Convergência Digital – INCoD/UFSC pelo apoio oferecido ao longo deste trabalho.





Eu ouço, e esqueço. Eu vejo, e me lembro. Eu  
faço, e compreendo.

Confúcio



## RESUMO

A utilização de jogos educacionais para o ensino de Computação vem se mostrando uma eficiente estratégia instrucional que leva os alunos a uma aprendizagem ativa e adquirir novos conhecimentos, habilidades e atitudes. Assim, nos últimos anos diversos jogos (digitais e não-digitais) para ensinar várias áreas de Computação (p. ex. Engenharia de Software, programação, redes de computadores e segurança) foram desenvolvidos. Tipicamente estes jogos são desenvolvidos pelos próprios professores das disciplinas de modo *ad-hoc* e não são avaliados de forma rigorosa. Portanto, é necessária uma base melhor de design instrucional e de design de jogos para assegurar que os jogos desenvolvidos atingem os seus objetivos de aprendizagem. Neste contexto, a Tese apresenta o processo ENgAGED - um processo de desenvolvimento de jogos educacionais, que integra processo de design instrucional e processo de design de jogos. O processo foi desenvolvido com base em uma revisão sistemática da literatura, comparando os processos existentes e de forma sistemática integrando elementos do design instrucional e o design de jogos. O processo foi avaliado do ponto de vista da qualidade do processo e da qualidade do produto. A avaliação da qualidade do processo foi realizada por um painel de 10 especialistas que consideraram o processo é não-ambíguo, útil, consistente, completo, compreensível e correto. O processo também foi avaliado positivamente em relação a flexibilidade e usabilidade para o desenvolvimento de vários tipos de jogos. A qualidade do produto foi avaliado por meio de uma serie de estudo de casos, desenvolvendo 4 jogos educacionais (*SCRUM-Scape*, *SCRUM'ed*, *PM-Quiz* e *Fuga de Gambix*) com o processo ENgAGED. Espera-se que os resultados dessa Tese facilitem o desenvolvimento de novos jogos educacionais para área da Computação, contribuindo para a sua qualidade tanto em termos educacionais quanto de jogos e assim em geral contribuir positivamente ao ensino de Computação.

**Palavras-chave:** Jogo educacional, Design Instrucional, Design de jogo.



## ABSTRACT

The use of educational games for teaching computing has been shown to be an effective instructional strategy that leads students to active learning and acquiring new knowledge, skills, and attitudes. Recently various games (digital and non-digital) have been developed to teach diverse computing areas (e.g. software engineering, programming, computer networks, and security). Typically, these games are developed by the teachers themselves in an ad-hoc manner and many are not rigorously evaluated. Therefore, a better basis for instructional design and game design is required to ensure that the developed games achieve their learning goals. In this context, this thesis presents the process ENgAGED - a process for developing educational games, which integrates parts of the instructional design and game design process. The process was developed based on a systematic literature review, comparing existing processes and systematically integrating elements of instructional design and game design. The process was evaluated from the point of view process quality and quality of the product. The process quality was evaluated by a panel of 10 experts, who considered the process is non-ambiguous, useful, consistent, complete, understandable and correct. The process was also evaluated positively in relation to flexibility and usability for the development of several types of games. The product quality has been evaluated through a series of case studies developing 4 educational games (*SCRUM-Scape*, *SCRUM'ed*, *PM-Quiz* e *Fuga de Gambix*) following the process ENgAGED. We expect that the. Results of this thesis facilitate the development of new educational games for teaching computing contributing to their quality and, thus, contribute positively to computer education.

**Keywords:** Educational game, Instructional Design, Game Design.



## LISTA DE FIGURAS

Figura 1–Metodologia de pesquisa. ....	26
Figura 2– Processo ADDIE. ....	35
Figura 3– Categorias das estratégias instrucionais.....	39
Figura 4– Distribuição dos jogos por área de conhecimento. ....	52
Figura 5– Distribuição por tipo de objetivos de aprendizagem.....	54
Figura 6– Nível de objetivos de aprendizagem: Cognitivo.....	55
Figura 7– Nível de objetivos de aprendizagem: Afetivo.....	55
Figura 8– Nível de objetivos de aprendizagem: Psicomotor.....	56
Figura 9– Jogos digitais vs. não-digitais.....	57
Figura 10– Distribuição de jogos digitais de acordo com sua plataforma.....	58
Figura 11– Distribuição de jogos não-digitais de acordo com sua plataforma. ....	59
Figura 12– Gênero dos jogos. ....	60
Figura 13– Modo de interação. ....	61
Figura 14– Duração das seções do jogo.....	62
Figura 15– Distribuição do tipo de pesquisa ( <i>research design</i> ). ....	64
Figura 16– Distribuição das avaliações, segundo modelo Kirkpatrick. ....	64
Figura 17 – Processo para o mapeamento sistemático.....	69
Figura18 – ENgAGED - <i>Educational Games Development</i> .....	79
Figura 19–Avaliação da Qualidade do Produto .....	110
Figura 20– Composição da Avaliação. ....	111
Figura 21–Gráfico da avaliação da qualidade do processo. Características: corretude, compreensibilidade, completude, consistência e ambiguidade. ....	122
Figura 22–Gráfico da avaliação da qualidade do processo. Característica: utilidade. ....	122
Figura 23–Gráfico da avaliação da qualidade do processo. Características: usabilidade e flexibilidade. ....	123
Figura 24– Medianas da avaliação da motivação. ....	126
Figura 25– Medianas da avaliação da experiência dos usuários. ....	127
Figura 26– Medianas da avaliação da aprendizagem.....	128





## LISTA DETABELAS

Tabela 1 – Componentes do <i>Systems Approach Model</i> .....	34
Tabela 2– Níveis cognitivos de aprendizagem.....	36
Tabela 3– Níveis psicomotores de aprendizagem.....	37
Tabela 4– Níveis afetivos de aprendizagem.....	38
Tabela 5– Visão geral do modelo de quatro níveis de Kirkpatrick .....	40
Tabela 6– Gênero de jogos. ....	42
Tabela 7– Relação entre taxonomia de Bloom e gênero de jogos.....	43
Tabela 8– Plataforma de jogos.....	44
Tabela 9– Modos de interação .....	44
Tabela 10– Dimensões para extração de dados.....	51
Tabela 11–Processo identificados na revisão sistemática da literatura. ....	71
Tabela 12–Fases dos processos encontrados.....	75
Tabela 13 – Exemplo de estrutura para detalhamento das atividades do processo ENgAGED.....	81
Tabela 14–Visão geral do processo ENgAGED. ....	82
Tabela 15 – Comparativo entre ENgAGED e processos encontrados na literatura.....	88
Tabela 16 – Desenvolvimento do jogo <i>SCRUM-Scape</i> . ....	96
Tabela 17 – Desenvolvimento do jogo <i>SCRUM'ed</i> . ....	98
Tabela 18 – Desenvolvimento do jogo <i>PM Quiz</i> . ....	101
Tabela 19 – Desenvolvimento do jogo <i>Fuga de Gambix</i> . ....	103
Tabela 20– Objetivo, questões de análise e métricas para avaliação do processo ENgAGED.....	112
Tabela 21–Relação das métricas e questões.....	114



## **LISTA DE ABREVIATURAS E SIGLAS**

ADDIE – Análise, Projeto, Desenvolvimento, Implementação e Avaliação (tradução do Inglês)

AL – *Active Learning*

ARCS – *Attention, Relevance, Confidence, Satisfaction*

DI – Design Instrucional

ENgAGED – *EducatioNAl GamEs Development*

GBL – *Game-based Learning*

IES – Instituição de Ensino Superior

ISD – *Instructional System Design*

MEEGA – Modelo de Avaliação de Jogos Educacionais

SAM – *Systems Approach Model*

TAM – *Technology Acceptance Model*

TI – Tecnologia da Informação

TRA – *Theory of Reasoned Action*

UI – Unidade Instrucional



# SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>23</b>
1.1 OBJETIVOS.....	24
<i>Objetivo Geral.....</i>	<i>24</i>
<i>Objetivos Específicos.....</i>	<i>24</i>
1.2 METODOLOGIA.....	25
1.3 ADERÊNCIA A CIÊNCIA DA COMPUTAÇÃO .....	29
1.4 JUSTIFICATIVA E RELEVÂNCIA DO TEMA .....	29
1.5 INEDITISMO DO TRABALHO.....	30
1.6 CONTRIBUIÇÃO .....	31
<i>Contribuição científica.....</i>	<i>31</i>
<i>Contribuição social.....</i>	<i>31</i>
1.7 ESCOPO DA TESE.....	32
1.8 ESTRUTURA DA TESE.....	32
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>34</b>
2.1 DESIGN INSTRUCIONAL .....	34
2.2 JOGOS EDUCACIONAIS.....	41
<b>3 ESTADO DA ARTE .....</b>	<b>48</b>
3.1 JOGOS EDUCACIONAIS ENCONTRADOS .....	48
<i>Definição do protocolo da revisão da literatura.....</i>	<i>48</i>
<i>Execução da busca.....</i>	<i>50</i>
<i>Extração da informação.....</i>	<i>52</i>
<i>Discussão.....</i>	<i>65</i>
3.2 PROCESSOS DE DESENVOLVIMENTO DE JOGOS EDUCACIONAIS ENCONTRADOS .....	65
<i>Definição do protocolo da revisão da literatura.....</i>	<i>66</i>
<i>Execução da busca.....</i>	<i>67</i>
<i>Extração da informação.....</i>	<i>67</i>
<i>Discussão.....</i>	<i>77</i>
<b>4 PROCESSO ENGAGED.....</b>	<b>79</b>
4.1 DESCRIÇÃO DO PROCESSO ENGAGED .....	79
4.2 COMPARATIVO ENTRE ENGAGED E TRABALHOS RELACIONADOS.....	87
<b>5 APLICAÇÃO DO PROCESSO ENGAGED .....</b>	<b>95</b>
5.1 DEFINIÇÃO DA APLICAÇÃO DO PROCESSO .....	95
5.2 EXECUÇÃO DA APLICAÇÃO DO PROCESSO .....	95

<i>Jogo SCRUM-Scape</i> .....	95
<i>Jogo SCRUM'ed</i> .....	98
<i>Jogo PM Quiz</i> .....	101
<i>Jogo Fuga de Gambix</i> .....	103
<b>5.3 ANÁLISE DA APLICAÇÃO DO PROCESSO</b> .....	<b>105</b>
<b>6 AVALIAÇÃO DO PROCESSO ENGAGED</b> .....	<b>108</b>
6.1 DEFINIÇÃO DE QUALIDADE .....	108
6.2 DEFINIÇÃO DA AVALIAÇÃO .....	111
6.3 EXECUÇÃO DA AVALIAÇÃO DA QUALIDADE DO PROCESSO ENGAGED.....	120
6.4 ANÁLISE DA AVALIAÇÃO DA QUALIDADE DO PROCESSO ENGAGED.....	121
6.5 EXECUÇÃO DA AVALIAÇÃO DA QUALIDADE DO JOGO EDUCACIONAL .....	124
6.6 ANÁLISE DA AVALIAÇÃO DA QUALIDADE DO JOGO EDUCACIONAL .....	125
6.7 AMEAÇAS A VALIDADE DA PESQUISA .....	128
<b>7 CONCLUSÕES</b> .....	<b>131</b>
<b>REFERÊNCIAS</b> .....	<b>136</b>
<b>APÊNDICE A –JOGOS ENCONTRADOS NA REVISÃO SISTEMÁTICA DA LITERATURA</b> .....	<b>162</b>
<b>APÊNDICE B – CLASSIFICAÇÃO DOS JOGOS COM RELAÇÃO AOS ASPECTOS INSTRUCIONAIS</b> .....	<b>187</b>
<b>APÊNDICE C – CLASSIFICAÇÃO DOS JOGOS COM RELAÇÃO AOS ASPECTOS DE JOGOS</b> .....	<b>196</b>
<b>APÊNDICE D – FORMA DE DESENVOLVIMENTO DOS JOGOS SELECIONADOS NA REVISÃO SISTEMÁTICA DA LITERATURA</b> .....	<b>204</b>
<b>APÊNDICE E – FASES E ATIVIDADES DO PROCESSO ENGAGED</b> .....	<b>218</b>
<b>APÊNDICE F – QUESTIONÁRIO PARA AVALIAÇÃO DO PROCESSO ENGAGED</b> .....	<b>327</b>
<b>APÊNDICE G – QUESTIONÁRIO PARA AVALIAÇÃO DO JOGO EDUCACIONAL</b> .....	<b>333</b>
<b>APÊNDICE H – TCLE - TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO</b> .....	<b>336</b>

<b>APÊNDICE I – APLICAÇÕES DO PROCESSO ENGAGED DETALHADOS.....</b>	<b>338</b>
<b>APÊNDICE J – FASES E ATIVIDADES DA PRIMEIRA VERSÃO DO PROCESSO ENGAGED .....</b>	<b>400</b>





## 1 INTRODUÇÃO

Os cursos de Computação em instituições de ensino superior (IES) são formados por uma complexa matriz curricular. Estes cursos são 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). Além disso, os cursos de base tecnológica são considerados fundamentais para a sociedade (PRATT, 2012). Por exemplo, no contexto do mercado de software e serviços, o Brasil ocupa a sétima posição no *ranking* mundial e atingiu um faturamento de 25,1 bilhões de dólares em 2013 (ABES, 2013). Entretanto, a demanda por profissionais de Computação qualificados está aumentando continuamente, exigindo um ensino superior cada vez mais eficaz. Com isso, as disciplinas dos cursos de Computação precisam atrair estudantes interessados em apreender, para que seja possível a formação contínua de profissionais capazes e responsáveis (ACM/IEEE-CS, 2005; ACM/IEEE-CS, 2013; MINISTÉRIO DA EDUCAÇÃO, 2012).

Ao mesmo tempo, o ensino em Computação precisa ser atraente para motivar os alunos. Embora a natureza da área da Computação e da geração de estudantes tenha mudando nos últimos anos, a maioria dos cursos de Computação ainda ensina de modo tradicional seus conteúdos, o que pode não ser completamente suficientes para motivar e atrair os alunos (BARNES et al., 2008, PARSONS, 2011).

Uma abordagem adotada para conter o crescente desinteresse dos alunos, buscando envolve-los de uma forma mais atraente é o aprendizado baseado em jogos (GBL) (MORRISON e PRESTON, 2009). Na aprendizagem baseada em jogos, as aplicações de jogos são utilizadas como recurso didático tendo como objetivo a aprendizagem (ABT, 2002, PRENSKY, 2007). Tipicamente, sendo projetados para equilibrar conteúdo e jogabilidade.

Neste contexto, um jogo pode ser definido como "*qualquer competição (jogo) entre os adversários (jogadores) que operam sob restrições (regras) para um objetivo (vitória ou lucro)*" (ABT, 2002). Assim, jogos educacionais, também denominados de jogos sérios (*serious games*) são especificamente projetados para ensinar as pessoas um determinado assunto, expandir conceitos, reforçar o desenvolvimento, exercitar e aprender uma habilidade ou buscar uma mudança de atitude enquanto jogam (DEMPSEY, RASMUSSEN e LUCASSEN, 1996).

Portanto, aprendizagem baseada em jogos parece ser uma alternativa promissora para ensinar Computação em instituições de ensino superior. Entretanto, há falhas de informações sistematizadas sobre a existência destes jogos, bem como sobre a forma de desenvolvê-los, o que pode dificultar a adoção deles como estratégia instrucional. Embora existam revisões da literatura que visam a (meta)análise do impacto dos jogos para o ensino (por exemplo, Boyle et al. (2016), Connolly et al. (2012), Ke (2009), Hays (2005) e Vogel et al. (2006)), geralmente cobrem um ou mais escopo de qualquer tipo de assunto (por exemplo, Connolly et al. (2012), Ke (2009)) ou então limitados à uma área de conhecimento da Computação, como Engenharia de Software (WANGENHEIM e SHULL, 2009, CAULFIELD et al. 2011b, CONNOLLY et al., 2007) ou Fundamentos de Programação (LI e WATSON, 2011). Outra limitação é que várias revisões da literatura focam somente em jogos digitais (DEMPSEY, RASMUSSEN e LUCASSEN, 1996, KE, 2009, LI e WATSON, 2011, CONNOLLY et al. 2007).

## **Questão de Pesquisa**

*Como desenvolver jogos educacionais que ensinam competências da Computação de modo efetivo, eficiente e divertido?*

### 1.1 OBJETIVOS

#### **Objetivo Geral**

A fim de orientar a adoção ou criação de jogos educacionais para o ensino nos cursos de graduação em Computação das Instituições de Ensino Superior, a Tese tem por objetivo propor um processo para o desenvolvimento de jogos educacionais.

#### **Objetivos Específicos**

Objetivando a proposição do processo, apresenta-se a seguir os seguintes objetivos específicos:

- I. Análise dos jogos educacionais existentes utilizados para ensino, na área da Computação.
- II. Análise dos processos de desenvolvimentos de jogos educacionais disponíveis na literatura.

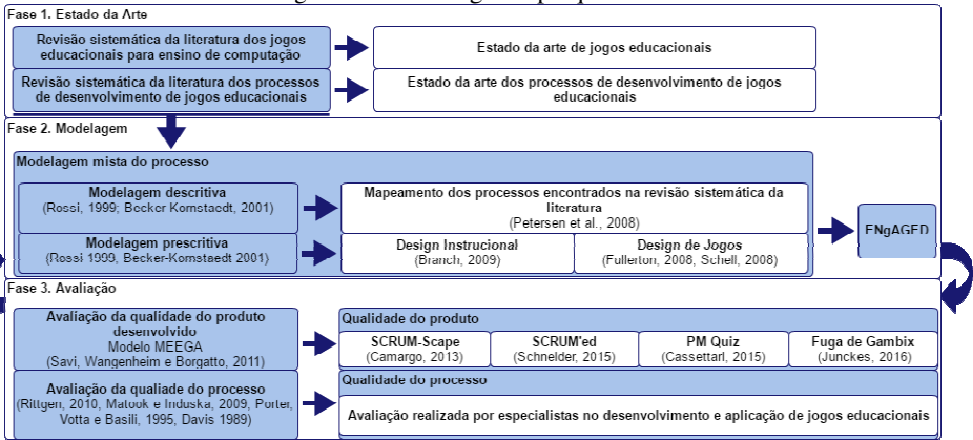
- III. Proposição de um novo processo de desenvolvimento de jogos educacionais para área da Computação.
- IV. Aplicação e avaliação a qualidade do processo para área da Computação.

## 1.2 METODOLOGIA

O método científico utilizado nesta pesquisa é o indutivo, pois as conclusões levam a um conteúdo muito mais amplo do que as premissas nas quais se basearam os argumentos iniciais pesquisados (MARCONI e LAKATOS, 2003). Neste sentido, a pesquisa se enquadra no contexto exploratório por ter 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 no problema pesquisado (GIL, 2010).

Considera-se esta pesquisa interpretativista (SAUNDERS, LEWIS e THORNHILL, 2009), porque apresenta aspectos de interpretação de papéis sociais e fenomenológicos. Desta forma, não sendo possível criar uma independência do pesquisador em relação ao objeto pesquisado. A estratégia de pesquisa envolve pesquisa em arquivos, *survey*, estudo de caso, observação, além de outras estratégias (SAUNDERS, LEWIS e THORNHILL, 2009). Em relação à técnica de pesquisa considera-se como método misto, porque contém alegações de conhecimento pragmático e coleta sequencial de dados quantitativos e qualitativos (CRESWELL, 2007). A metodologia utilizada para alcançar os objetivos desta Tese está dividida em 3 fases, sendo apresentada resumidamente na figura 1.

Figura 1–Metodologia de pesquisa.



Fonte: Elaborada pelo autor.

## 1ª Fase: Estado da arte

Esta fase é estruturada por duas revisões sistemáticas da literatura. A primeira apresenta o estado da arte dos jogos para ensino em Computação e a segunda dos processos de desenvolvimento de jogos educacionais.

### Identificação dos jogos para ensino em computação

Com objetivo de identificar os jogos utilizados para ensino em Computação foi realizada uma revisão sistemática da literatura, seguindo o método proposto por Kitchenham (2004). Este método envolve as seguintes atividades: definição da busca, execução da busca, extração e análise qualitativa das informações encontradas. A revisão da literatura de jogos busca responder as seguintes perguntas de pesquisa: (1) Quais jogos são utilizados para o ensino em Computação e em quais áreas de conhecimento? (2) Quais são os objetivos de aprendizagem? (3) Quais tipos de jogos são utilizados? (4) Como os jogos são desenvolvidos e avaliados?

## **Identificação de processos de desenvolvimento de jogos educacionais existentes**

Com o objetivo de identificar os processos de desenvolvimento de jogos educacionais disponíveis na literatura, foi realizada uma revisão sistemática da literatura, segundo Kitchenham (2004). Esta revisão da literatura busca as respostas para as seguintes perguntas de pesquisa: (1) Quais processos existem para desenvolver jogos educacionais? (2) Quais são as etapas/fases/atividades desses processos? (3) quais similaridades ou diferenças existem entre os processos existentes?

Após o levantamento do estado da arte foi realizado um mapeamento dos processos existentes em nível de fases e atividades, ou seja, as fases e atividades dos processos encontrados foram relacionados e agrupados de acordo com suas similaridades. Ele é fundamentado no processo de mapeamento sistemático para Engenharia de Software proposto por Petersen et al. (2008), mas também nas pesquisas de Budgen et al. (2008), Afzal, Torkar e Feldt (2009) e Silva Neto (2011). De modo geral, as fases do processo de mapeamento sistemático são: definição das questões de pesquisas; condução da busca; palavras-chaves usadas no *abstract*; extração de dados e o processo de mapeamento. Considera-se que as fases são complementares ao método proposto por Kitchenham (2004), exceto a fase de *processo de mapeamento*, a qual foi utilizada na pesquisa para realizar o mapeamento dos processos encontrados na revisão sistemática da literatura.

### **2ª Fase: Modelagem do processo ENgAGED**

Com o objetivo de propor um novo processo de desenvolvimento de jogos educacionais, foi modelado um processo com base em uma unificação dos processos mapeados na fase anterior. Em seguida, foram observadas lacunas nestes processos, sendo necessário complementá-lo com os elementos de design instrucional e design de jogos para assegurar o desenvolvimento de jogos educacionais eficientes, eficazes e atraentes (MERRIL et al., 1996, FILATRO, 2008, FULLERTON, 2008, CLARK, 2009).

De modo geral, a modelagem de um processo pode ser realizada em três abordagens: descritiva, prescritiva e mista (ROSSI, 1999) e (BECKER-KORNSTAEDT, 2001). Na modelagem do processo ENgAGED foi utilizada a modelagem mista, na qual inclui a

modelagem descritiva a partir dos resultados da revisão da literatura e a modelagem prescritiva a qual permitiu completar as lacunas observadas em relação aos modelos de referência, tanto na área de design instrucional, quanto no design de jogos (BECKER-KORNSTAEDT, 2001, ROSSI, 1999, FILATRO, 2008, CLARK, 2009, FULLERTON, 2008, SCHELL, 2008).

### **3ª Fase: Avaliação do processo ENgAGED**

O processo ENgAGED foi avaliado em termos da qualidade do processo, do ponto de vista de especialistas (professores e alunos) que desenvolveram jogos educacionais, ou professores que utilizam jogos educacionais em sala de aula na área da Computação. Para definir as métricas da avaliação, foi adotado o GQM (*Goal, Question, Metric*) (BASILI, CALDIERA, ROMBACH, 1994). A partir do objetivo da avaliação, foram definidas as questões de análise, métricas e um questionário para a coleta de *feedback* dos especialistas. As questões de análise e métricas foram criadas com base nas seguintes características que estão relacionadas à qualidade de um processo (RITTGEN 2010, MATOOK e INDUSKA 2009, PORTER, VOTTA e BASILI 1995, DAVIS 1989): ambiguidade, consistência, completude, compreensibilidade, corretude, flexibilidade, funcionalidade, usabilidade e utilidade. De acordo com o objetivo da avaliação foi realizado um estudo de caso (YIN, 2012) possibilitando a coleta de dados. No estudo de caso, foram desenvolvidos 4 jogos educacionais.

Para avaliar a qualidade do processo, foram realizadas duas avaliações. A primeira avaliou a qualidade do produto do processo, para isso foi utilizado o modelo de avaliação de jogos educacionais MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011). Para esta avaliação, foram realizadas quatro aplicações do processo, sendo desenvolvidos os seguintes jogos: *SCRUM-Scape* (CAMARGO, 2013), *SCRUM'ed* (SCHNEIDER, 2015), *PM Quiz* (CASSETTARI, 2015), e *Fuga de Gambix* (JUNCKES, 2016).

A segunda avaliação foi realizada com um painel de especialistas no desenvolvimento e utilização de jogos educacionais. Nesta avaliação, 10 especialistas avaliaram o processo ENgAGED, sendo todos da área da Computação, 6 desenvolvedores de jogos educacionais e 4 autores de artigos científicos de jogos educacionais.

### 1.3 ADERÊNCIA A CIÊNCIA DA COMPUTAÇÃO

A presente Tese está vinculada ao Programa de Pós-Graduação em Ciência da Computação (PPGCC) que contém os seguintes objetivos:

1. Capacitação de pesquisadores e docentes do magistério superior em Ciência da Computação e áreas afins;
2. Desenvolvimento de novos conhecimentos em Ciência da Computação.

O método definido na pesquisa relaciona-se aos objetivos da linha de pesquisa de Engenharia de Software do PPGCC que busca “*formar indivíduos capazes de conduzir o processo de desenvolvimento de software e de investigar novas metodologias, técnicas e ferramentas para a concepção de sistemas*” (PPGCC, 2016).

A Engenharia de Software é definida como uma disciplina voltada a aplicação de teorias, conhecimentos e práticas para efetivamente e eficientemente construir sistemas de software que satisfaçam os requisitos dos clientes e usuários (ACM/IEEE-CS, 2013). São temas tipicamente abordados na Engenharia de Software: ferramentas para a gestão de desenvolvimento de software, **análise e modelagem de artefatos de software**, avaliação e controle da qualidade, bem como o controle da evolução e reutilização de sistemas de software (ACM/IEEE-CS, 2013).

Levando em consideração a aderência ao PPGCC, a pesquisa está alinhada à área de Engenharia de Software, a qual apresenta um **processo de desenvolvimento de jogos educacionais**.

### 1.4 JUSTIFICATIVA E RELEVÂNCIA DO TEMA

O cenário de TI (Tecnologia da Informação) no mundo está avançando cada vez mais, por exemplo, o Brasil ocupa a sétima posição no ranking mundial do mercado de software e somente em 2014 obteve 60 bilhões de dólares de investimentos na área de TI (ABES, 2013, ABES, 2014). Entretanto, os cursos de Computação possuem uma grande evasão de alunos. Segundo pesquisa realizada pela SEMESP (Sindicato das Entidades Mantenedoras de Estabelecimentos de Ensino Superior no Estado de São Paulo) em 2013, cerca de 73% dos alunos de Ciências da Computação e 67,3% de Sistema de Informação abandonaram os cursos para trabalhar em empresas tecnológicas. O principal motivo identificado na pesquisa está relacionado a grande

oferta de vagas sem exigir diploma. Porém, estas empresas quando estão em situações de crises econômicas acabam dispensando primeiramente os funcionários menos qualificados/graduados. E também, os alunos que abandonam os cursos, raramente retornam a faculdade (SEMESP, 2013).

Considerando as tentativas de tornar os cursos de Computação mais atrativos (GERAB, GERAB e BUENO, 2014, ALBUQUERQUE et al., 2014), a utilização de jogos educacionais pode apoiar e estimular o ensino de Computação por meio das seguintes características: (a) tornar o conteúdo abstrato mais prático; (b) tornar o aprendizado mais atrativo; (c) criar um ambiente de maior imersão ao conteúdo; (d) apreender a atenção do aluno; (e) desenvolver aspectos psicológicos, no caso de jogos grupais; (f) oferecer uma estratégia instrucional diferente das atuais aulas expositivas; (g) desenvolver a competitividade positiva entre os alunos (KIRRIEMUIR e MCFARLANE, 2004, FREITAS, 2006, MITCHELL e SAVILL-SMITH, 2004, PRENSKY, 2007, WANGENHEIM e SHULL, 2009, HAYS e SINGER, 1988).

Porém, grande parte dos jogos educacionais são desenvolvidos de modo *ad-hoc*. Tipicamente os jogos são desenvolvidos pelo próprio professor da disciplina, o qual utiliza sua experiência em sala de aula e de desenvolvimento de software para implementar jogos. Contudo, estes jogos normalmente não equilibram aspectos do design instrucional e design de jogos.

## 1.5 INEDITISMO DO TRABALHO

A literatura apresentada no capítulo do Estado da Arte contempla trabalhos recentes publicados nas últimas décadas na área de jogos educacionais para ensino de Computação. A literatura revela a existência de diversos jogos para ensino de Computação, mas não descrevem como os jogos são desenvolvidos ou avaliados. A literatura também revela que existem poucos processos de desenvolvimento de jogos educacionais disponíveis na literatura. Em geral, grande parte dos processos disponíveis descreve apenas características necessárias para os jogos educacionais. Os poucos processos disponíveis que descrevem com maior detalhe as fases e atividades do desenvolvimento, não equilibram aspectos do design instrucional e design de jogos, tornando difícil avaliar se os jogos desenvolvidos alcançam os benefícios esperados, como motivação e aprendizagem dos alunos.

A partir do estudo de revisão da literatura observa-se a necessidade do desenvolvimento sistemático de jogos educacionais na



área da Computação. Neste contexto, configura-se o ineditismo do presente trabalho, que busca propor um novo processo de desenvolvimento de jogos educacionais para preencher as lacunas dos processos encontrados na literatura, de modo a equilibrar aspectos do design instrucional e design de jogos.

## 1.6 CONTRIBUIÇÃO

A seguir são apresentadas as contribuições da Tese, contextualizadas em termos de contribuição científica e social.

### **Contribuição científica**

Como contribuição científica, a Tese busca oferecer subsídios para a produção de jogos educacionais para ensino em Computação. Neste sentido, busca-se definir as características dos jogos, como por exemplo, gênero, plataforma, licença e custo. Busca-se também identificar as áreas da Computação em que os jogos educacionais são mais utilizados e quais processos de desenvolvimento de jogos estão sendo utilizados tipicamente em sala de aula ou como exercício extraclasse.

A Tese visa também contribuir com a identificação dos perfis de alunos/jogadores de Computação, objetivando alinhar os interesses dos alunos por jogos educacionais e também auxiliar os professores a compreender melhor o público-alvo.

Em relação à área de Engenharia de Software, a Tese busca contribuir com um novo processo de desenvolvimento de jogos educacionais, constituído por fases que agrupam atividades e produtos de entrada e produtos de saída.

### **Contribuição social**

Como contribuição social, a Tese busca auxiliar na produção de jogos educacionais mais éticos, sem apologia à violência e à conteúdos impróprios. Busca-se também contribuir com maior interação e imersão entre os alunos e busca-se aproveitar a experiência a priori dos alunos com jogos (p. ex. computador, console, tabuleiro, cartas) para aplicação de conhecimentos da área de Computação. Portanto, busca-se favorecer o aprendizado e atratividade de conteúdos computacionais por meio da utilização de jogos.

## 1.7 ESCOPO DA TESE

A pesquisa restringe-se ao desenvolvimento de jogos educacionais para área da Computação. Portanto, não está no escopo da pesquisa o desenvolvimento de jogos para outras áreas de conhecimento, como Matemática, Física, Psicologia, Pedagogia.

Os jogos educacionais desenvolvidos durante a pesquisa restringiram-se disciplina de Engenharia de Software. Por meio do processo proposto, foram desenvolvidos os jogos *SCRUM-Scape* (CAMARGO, 2013), *SCRUM'ed* (SCHNEIDER, 2015), *PM Quiz* (CASSETTARI, 2015), e o *Fuga de Gambix* (JUNCKES, 2016). O desenvolvimento foi realizado por alunos de graduação de Ciências da Computação e Sistemas de Informação da UFSC. Mas a avaliação do processo foi realizada por um painel de especialistas de diferentes áreas da Computação.

## 1.8 ESTRUTURA DA TESE

A presente pesquisa está estruturada nos seguintes capítulos: 2 *Fundamentação Teórica* apresentando as definições que fundamentam o design instrucional e jogos educacionais; 3 *Estado da Arte* apresentando os jogos educacionais para ensino de Computação e os processos para desenvolvimento de jogos disponíveis na literatura; 4 *Processo ENgAGED* apresentando o processo proposto na Tese e realizando um comparativo entre os processos encontrados na literatura; 5 *Aplicações do Processo ENgAGED* apresentando um estudo de caso na qual permitiu o desenvolvimento de 4 jogos educacionais para o ensino em áreas da Computação; 6 *Avaliação do Processo ENgAGED* apresentando a definição, execução e análise da avaliação do processo realizada por um painel de especialistas e também com base nas avaliações dos jogos desenvolvidos; *Conclusão* apresentando as considerações finais da Tese.



## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentadas as teorias que fundamentam o design instrucional e jogos educacionais.

### 2.1 DESIGN INSTRUCIONAL

Entre os principais objetivos da educação é possível destacar o ensino de novas competências. Para alcançar os objetivos esperados na educação, tipicamente utiliza-se processos de designs instrucionais (FINCHER, 2012). O design instrucional pode ser visto como um sistema organizado e estruturado que tem por objetivo promover a aprendizagem (BRANCH, 2009). Por meio do design instrucional é possível visualizar a importância de cada elemento, sem apresentar ênfase excessiva há um único elemento deste sistema (DICK e CAREY, 1996). Normalmente, estes elementos são alunos, professores, materiais didáticos e ambientes de aprendizagem (BRANCH, 2009, DICK e CAREY, 1996). No design instrucional busca-se criar “*experiências de ensino que fazem a aquisição de conhecimentos e habilidades mais eficiente, eficaz e atraente*” (MERRIL et al., 1996).

A partir do design instrucional é possível desenvolver unidades instrucionais de qualidade. Uma unidade instrucional é considerada uma série de lições, exercícios ou atividades organizadas em torno de um tópico ou tema principal. A unidade instrucional apresenta ao professor uma visão concisa do conteúdo, incluindo os materiais que serão utilizados em aula (HURWITZ e DAY, 2007). Em geral, existem diversos tipos de processos de design instrucional que permitem a produção de diferentes unidades instrucionais, como *Systems Approach Model* (SAM) (DICK e CAREY, 1996) (tabela 1). Tipicamente estes processos são denominados de *Instructional Systems Design* (ISD).

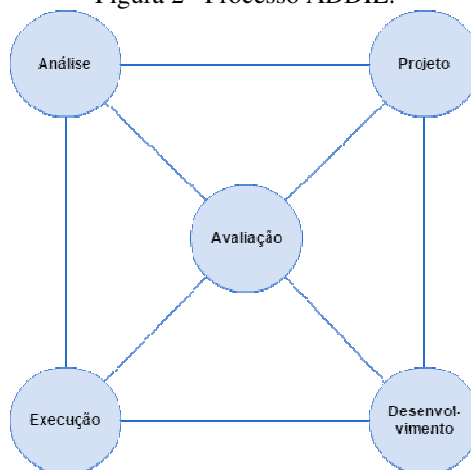
Tabela 1 – Componentes do *Systems Approach Model* (DICK e CAREY, 1996).

<b>Identificar metas instrucionais</b>	Identificar o que os alunos devem ser capazes de compreender ao final da instrução. Portanto, distância entre os objetivos desejados e a situação atual é identificada.
<b>Conduzir análise instrucional</b>	Identificar os passos relevantes para a realização de um determinado objetivo e as competências necessárias para o aluno atingir as metas.
<b>Analisar aprendizes e contextos</b>	Identificar as características do público-alvo, inclusive as competências anteriores, experiências e características diretamente relacionadas com a competência que está sendo ensinada; e realizar uma análise das configurações de

	desempenho e aprendizagem.
<b>Escrever os objetivos de desempenhos</b>	Definir os objetivos com base em todas as informações recolhidas até agora. Os objetivos consistem nas descrições de performance, condições e critérios.
<b>Desenvolver instrumentos de avaliação</b>	Desenvolver os instrumentos de avaliação, a fim de ser capaz de determinar se os alunos conseguiram atingir os objetivos.
<b>Desenvolver estratégias instrucionais</b>	Sequenciar e organizar o conteúdo, especificar as atividades de aprendizagem, e decidir sobre a forma de entregar o conteúdo e as atividades.
<b>Projetar e conduzir avaliações formativas</b>	Planejar e realizar a avaliação formativa, que é um julgamento dos pontos fortes e fracos da instrução em seus estágios de desenvolvimento, a fim de rever a instrução melhorando a eficácia e o reuso.
<b>Revisar instrução</b>	Utilizar os resultados da avaliação formativa para a revisão da unidade instrucional.
<b>Desenvolver e conduzir avaliações somativas</b>	Planejar e efetuar uma avaliação somativa para estudar o valor da unidade instrucional, como um todo.

No entanto, entre os processos de design instrucional disponíveis na literatura, o ADDIE (*Analyze, Design, Develop, Implement, Evaluate*) (figura 2) destaca-se como amplamente difundido e eficaz para produção de conteúdos de aprendizagem (MOLENA, 2003, BRANCH, 2009).

Figura 2– Processo ADDIE.



Fonte: BRANCH, 2009.

O ADDIE é um processo aplicado para produção de conteúdos que visam aprendizagem baseada em desempenho, sendo centrada no aluno (BRANCH, 2009). O processo ADDIE tem como função orientar situações complexas, normalmente ocorridas em ambientes educacionais ou em outros contextos de aprendizagens (BRANCH, 2009). A seguir são apresentadas, resumidamente, as fases do processo ADDIE.

**A fase de análise** tem por objetivo identificar possíveis lacunas de desempenho dos alunos. Para isso, é necessário identificar o público-alvo, identificar contexto (como disciplina, objetivos da disciplina, pré-requisitos, conteúdo programático) (BRANCH, 2009). Nesta fase, também é necessário definir os objetivos que avaliam o desempenho do aluno ao final da unidade instrucional. Os objetivos oferecem uma direção sobre a condução do conteúdo ao longo do design da unidade instrucional e oferece *feedback* do aprendizado dos alunos (MARZANO, 2007). Por meio dos objetivos de desempenho é possível avaliar se os alunos adquiriram as competências necessárias para aprendizagem (MECCLELLAND, 1973). A competência pode ser definida como a capacidade de ter conhecimento, habilidade e atitude em diversas situações (como a formação superior ou experiência prática) (RUAS, 2004). O conhecimento é o saber que está relacionado a *saber fazer algo*. A habilidade é o conhecimento aplicado no dia a dia. A atitude é o *querer fazer*, ou ter motivação para aplicar o conhecimento e exercitar as habilidades aprendidas (MASCARENHAS, 2008).

No entanto, em nível de graduação, espera-se que os alunos adquiram competência em Computação nos níveis cognitivos de conhecimento, compreensão e aplicação (ACM/IEEE-CS, 2005, ACM/IEEE-CS, 2013). Por exemplo, como orienta a taxonomia de níveis de aprendizagem de Bloom (1956), apresentado na tabela 2.

Tabela 2– Níveis cognitivos de aprendizagem (BLOOM, 1956).

6. Avaliação	Apresentar opinião e fazer julgamentos sobre informações e ideias.
5. Síntese	Compilar informações juntas de uma maneira distinta.
4. Análise	Examinar e dividir informações em partes, identificando causas. Fazer inferência e encontrar provas...
3. Aplicação	Aplicar o novo conhecimento. Conseguir resolver problemas de novas situações por meio da aplicação de um novo conhecimento, fatos e técnicas
2. Compreensão	Comparação demonstrativa, organizando, comparando, traduzindo e interpretando, fatos e ideias.
1. Conhecimento	Memorização de materiais previamente aprendidos por fatos, termos, conceitos básicos e respostas.

Os alunos também precisam adquirir habilidades, tais como, comunicação, trabalho e gestão de equipe em disciplinas (ACM/IEEE-CS, 2005, ACM/IEEE-CS, 2013). Estas habilidades geralmente são consideradas como domínio psicomotor e descreve a capacidade de manipular fisicamente uma ferramenta ou instrumento. No entanto, no contexto da Computação, também são incluídas as habilidades de trabalho em equipe, comunicação, liderança. Tais objetivos de aprendizagem se concentram em mudanças e/ou desenvolvimento de comportamentos ou habilidades. Elas podem ser classificadas como apresentado na tabela 3 (SIMPSON, 1972).

Tabela 3– Níveis psicomotores de aprendizagem  
(SIMPSON, 1972).

7. Originalidade	Criação de novos padrões de movimento para atender a uma determinada situação ou problema específico. Os resultados da aprendizagem enfatizam a criatividade com base em habilidades altamente desenvolvidas, como a construção de uma nova teoria de comunicação.
6. Adaptação	As habilidades são bem desenvolvidas e o indivíduo pode modificar os padrões de movimento para atender as necessidades especiais, por exemplo, responder eficazmente às experiências inesperadas.
5. Respostas complexas	Desempenho hábil, dos atos que envolvem padrões complexos e sem hesitação tendo desempenho automático.
4. Mecanismo	Estágio intermediário de uma habilidade complexa. Respostas aprendidas se tornam habituais, realizando movimentos com alguma confiança e proficiência.
3. Resposta guiada	Estágios iniciais de uma aprendizagem de uma habilidade mais complexa, que inclui imitação, tentativa e erro. A adequação ao desempenho é alcançada através da prática, por exemplo, repetir um processo de comunicação demonstrando anteriormente.
2. Prontidão	Prontidão para agir, por exemplo, conhecer e agir em uma sequência de etapas em um processo de comunicação.
1. Percepção	Habilidade de usar estímulos sensoriais para orientar as atividades, por exemplo, detectar os sinais de uma comunicação não-verbal.

Além disso, o aluno também necessita desenvolver comportamento maduro, tendo atitudes profissionais, responsáveis e éticas. Para isso, os objetivos de aprendizagem precisam abordar aspectos emocionais e sentimentos. Apresenta-se na tabela 4 a classificação destes objetivos de aprendizagem (KRATHWOHL et al., 1973).

Tabela 4– Níveis afetivos de aprendizagem  
(KRATHWOHL et al., 1973).

5. Caracterizar	O estudante apresenta um determinado valor que exerce influência em seus comportamentos, até se tornar uma característica pessoal.
4. Organizar	O estudante pode apresentar diferentes valores, informações e ideias organizando em seu próprio esquema, contrastando diferentes valores, resolvendo conflito entre eles, e criando um sistema único de valores pessoais.
3. Valorizar	Quando o aluno atribui valor para objetivos, fenômenos ou informações.
2. Responder	O aluno participa ativamente do processo de aprendizado e reage de alguma forma expressando sua vontade de responder (motivação).
1. Receber	O aluno participa com atenção passiva.

**A fase de projeto** tem por objetivo definir o conteúdo da unidade instrucional e como ele será sequenciado, facilitando a interpretação e construção de conhecimentos e habilidades para os alunos (BRANCH, 2009). O conteúdo é o ponto focal para envolver o aluno no processo de construção de conhecimento (BRANCH, 2009). No entanto, o conteúdo precisa ser estrategicamente introduzido durante as aulas (BRANCH, 2009). Para isso utiliza-se uma estratégia instrucional, a qual desenvolve-se conhecimentos, habilidades e procedimentos durante a sessão de aprendizagem (como aula, palestra) (BRANCH, 2009, DICK e CAREY, 1996). O conteúdo de uma sessão de aprendizagem precisa ser distribuído de modo eficaz, tendo início, meio e fim (BRANCH, 2009). Existem diversos tipos de estratégias instrucionais, como aula-expositiva, vídeo, palestra, laboratórios, leitura, exercício ou jogos (LABURÚ e CARVALHO, 2013, IEEE, 2014, DICK e CAREY, 1996, BRANCH, 2009). Em geral, elas podem ser classificadas em cinco categorias, tal como a figura 3 (WEB POSTER WIZARD, 2009, SASKATCHEWAN EDUCATION, 1991).



Figura 3– Categorias das estratégias instrucionais.



Fonte: WEB POSTER WIZARD, 2009, SASKATCHEWAN EDUCATION, 1991.

**A fase de desenvolvimento** tem por objetivo gerar e validar a estratégia instrucional. De modo geral nesta fase o professor precisa produzir os conteúdos, selecionar ou desenvolver o material de apoio (como exercícios, atividades extraclasse), desenvolver guias para os alunos e revisar os conteúdos desenvolvidos (BRANCH, 2009).

**A fase de execução** tem por objetivo preparar o ambiente de aprendizagem (como sala de aula, laboratório de informática) e envolver e motivar os alunos. Portanto, nesta fase a estratégia instrucional é aplicada, levando os alunos a preencher as lacunas de conhecimento e adquirindo novas competências (como conhecimentos, habilidades e atitudes) de acordo com os objetivos de desempenho planejados na fase de análise da unidade instrucional (BRANCH, 2009).

**A fase de avaliação** tem por objetivo avaliar a qualidade da estratégia instrucional após a sua execução (BRANCH, 2009). Nesta fase é necessário determinar os critérios de avaliação, selecionar as ferramentas de avaliação, e conduzir a avaliação com os alunos. A avaliação pode ser conduzida em diferentes níveis avaliativos, como no modelo para avaliação de aprendizagem de Kirkpatrick (KIRKPATRICK e KIRKPATRICK, 2006), conforme apresenta a tabela 5.

Tabela 5– Visão geral do modelo de quatro níveis de Kirkpatrick (KIRKPATRICK e KIRKPATRICK, 2006).

N.	Nível de avaliação	Descrição e características	Exemplos de métodos de avaliação e instrumentos
1	<b>Reação</b>	Avaliam como os participantes se sentem sobre a formação ou experiência.	Tipicamente realiza um questionário distribuído no final de uma experiência de aprendizagem. Por exemplo, <i>happy-sheets</i> ; formulários de <i>feedback</i> , reações verbais, <i>survey</i> .
2	<b>Aprendizagem</b>	Avalia o aumento do conhecimento ou habilidades (antes e depois).	Avaliações e testes antes e depois da aprendizagem. Por exemplo, entrevistas ou observações.
3	<b>Comportamento</b>	Avalia o grau de um novo aprendizado adquirido durante a aprendizagem, medindo assim o desempenho real em um ambiente de trabalho.	Observação e entrevistas ao longo do tempo necessárias para avaliar a mudança, sua relevância e sustentabilidade.
4	<b>Resultados</b>	Avalia o efeito prático da aprendizagem no negócio.	Estudos de pós-aprendizagem; observação como parte da formação contínua; medidas, como erros, retrabalho; entrevistas com os alunos e Professores; entrevistas com grupos de pessoas.

Portanto, a utilização do processo ADDIE na produção de uma unidade instrucional pode levar os alunos ao desenvolvimento de novas competências (BRANCH, 2009). A utilização deste processo contribui para o ensino de Computação em IES, porém, tradicionalmente a estratégia instrucional frequentemente utilizada é a aula-expositiva (ACM/IEEE-CS, 2013). Essa estratégia é adequada para apresentar conceitos abstratos e informações factuais para um grande grupo de alunos (CHOI e HANNAFIN, 1995, FREEMAN et al., 2014). Mas, a aprendizagem experimental tem se mostrado eficaz, especialmente para realizar uma aprendizagem mais profunda (ITIN, 1999). A aprendizagem experimental é um processo que faz sentido a partir de experiências diretas, por exemplo, através de simulações e jogos de RPG (*role-playing games*) (KOLB, 1984). Atualmente, usa-se diversos jogos educacionais no ensino de Computação, principalmente na Engenharia de Software, Programação ou Segurança (BATTISTELLA e WANGENHEIM, 2016c; BOYLE et al., 2016). Neste contexto, a presente pesquisa foca exclusivamente na produção de unidades

instrucionais que utilizem como estratégia instrucional jogos educacionais.

## 2.2 JOGOS EDUCACIONAIS

Um jogo é uma competição (jogo) entre adversários (jogadores) que operam sob restrições (regras) para um determinado objetivo (vitória ou lucro) (ABT, 2002). Jogos educacionais são especificamente projetados para ensinar os alunos um determinado assunto, expandir conceitos, reforçar o conhecimento, exercitar e aprender uma habilidade, ou então buscar uma mudança de atitude enquanto jogam (QIAN e CLARK, 2016, DEMPSEY, RASMUSSEN e LUCASSEN, 1996).

Jogos educacionais são considerados poderosas estratégias instrucionais e acredita-se resultar em uma ampla gama de benefícios. Eles aumentam a eficácia da aprendizagem, interesse, motivação e persistência dos alunos (QIAN e CLARK, 2016, KIRRIEMUIR e MCFARLANE, 2004, FREITAS, 2006, MITCHELL e SAVILL-SMITH, 2004, PRENSKY, 2007). Devido as suas características intrínsecas, tais como, competição, desafio e interação, eles podem transformar o aprendizado em uma experiência envolvente e divertida (KAFAI, 2001).

A partir da utilização de jogos educacionais, promove-se a aprendizagem baseada em jogos (*game-based learning* – GBL). Na GBL, os jogos são concebidos de modo a equilibrar a aprendizagem com a jogabilidade (QIAN e CLARK, 2016). GBL pode trazer diversos benefícios que possibilitam a aprendizagem da aplicação do conhecimento em situações práticas de forma motivadora e divertida (POULOVA e KLIMOVA, 2015, FREEMAN et al., 2014). A aprendizagem baseada em jogos desenvolve uma aprendizagem ativa, permitindo uma maior participação dos alunos e leva a uma maior compreensão do conteúdo. A aprendizagem ativa ocorre quando os alunos estão engajados mais em atividades do que apenas em ouvir. Com isso, eles ficam mais envolvidos no diálogo, debate, escrita, resolução de problemas e com pensamento crítico (FREEMAN et al., 2014, BONWELL e EISON, 1991).

No entanto, para que estes jogos educacionais alcancem os benefícios esperados, leva-se em consideração aspectos do design de jogos, além de aspectos instrucionais apresentados na seção anterior. No processo de desenvolvimento, considera-se aspectos fundamentais do design de jogos, como gênero, plataforma, modo de interação, regras,

narrativa, mecânica e elementos do jogo (como personagens, cenários, objetos).

O **gênero** de um jogo é a categorização que agrupa os tipos de jogos de acordo com suas características ou taxonomias (HERZ, 1997, ADAMS e ROLLINGS, 2006, SCHIFFLER, 2006, WOLF, 2001). Existem diversas tentativas parcialmente conflitantes para classificar jogos em gêneros ou em tipologias semelhantes (MICHAEL e CHEN, 2006), (SUSI, JOHANNESSON e BACKLUND, 2007), (SAWYER e SMITH, 2008). Estas classificações ou taxonomias de jogos são ou muito amplas (para qualquer tipo de domínio/área) ou se concentra em um domínio específico (por exemplo, educação em saúde). Portanto, é necessário adaptar essas taxonomias de jogos existentes para aplicá-las aos jogos educacionais. Neste contexto, Herz (1997) apresenta um sistema para o entretenimento que é semelhante ao utilizado comercialmente na indústria de jogos, distinguindo jogos de ação (jogos de reação, incluindo tiro e plataformas) jogos de aventura (resolvendo puzzles lógicos para progredir através de um mundo virtual), jogos de luta, jogos de quebra-cabeça, jogos de *role-playing*, simulações e jogos de estratégias. Embora tal classificação pode parcialmente ser usada para jogos educacionais, ainda há falta de uma definição clara de gêneros de jogos educacionais. Neste contexto, foram adaptadas as taxonomias de gêneros existentes (HERZ, 1997, ADAMS e ROLLINGS, 2006, SCHIFFLER, 2006, WOLF, 2001) para classificação de jogos educacionais. A tabela 6 apresenta a taxonomia que será utilizada ao longo desta pesquisa para classificar os jogos educacionais.

Tabela 6– Gênero de jogos.

<b>Gênero</b>	<b>Descrição</b>
<b><i>Ação</i></b>	Um jogo que exige que os jogadores utilizem reflexos rápidos, com precisão para superar obstáculos, resolver desafios ou para responder perguntas.
<b><i>Adivinhação</i></b>	Um jogo em que o objetivo é identificar algum tipo de informação, como uma palavra, a partir de desenhos ou imitação de outros jogadores.
<b><i>Aventura</i></b>	Um jogo em que o jogador assume o papel de protagonista da história e interage através da exploração e a solução de desafios/missões (como decodificar mensagens, encontrar e utilizar itens).
<b><i>Corrida</i></b>	Um jogo em que o jogador comanda um veículo ou participa de uma corrida, tentando se mover mais rápido do que os oponentes para alcançar um objetivo específico ou ser mais rápido que um determinado tempo.
<b><i>Estratégia</i></b>	Um jogo que é focado na jogabilidade e exige pensamentos táticos e

	cuidados, a fim de conseguir a vitória.
<i>Puzzle</i>	Um jogo que envolve personagens que controlam um conjunto de comandos, para navegar através de labirintos ou para organizar os objetos de maneira correta.
<i>Quiz</i>	Um jogo onde o jogador precisa responder perguntas para uma determinada área de conhecimento.
<i>Role-playing game (RPG)</i>	Um jogo onde o jogador controla ações de um protagonista e com este personagem vive imerso em um mundo fictício. Em jogo de role-playing os personagens interagem com este mundo e ficam mais fortes.
<i>Roll-and-move</i>	Jogos de tabuleiro em que as fichas são movidas com base nos resultados mostrados em um ou mais dados (por exemplo: <i>Snake and Ladders</i> e <i>Monopoly</i> ).
<i>Simulação</i>	Um jogo desenvolvido para colocar o jogador no controle de um determinado ambiente ou atividade, o qual busca ser o mais realista possível.

Para auxiliar na definição do gênero do jogo, apresenta-se abaixo uma relação entre os níveis de aprendizagem da Taxonomia de Bloom e gênero de jogos, criada com base em Battistella e Wangenheim (2016c) (tabela 7). Na tabela 7 foram identificados os níveis de aprendizagem segundo a Taxonomia de Bloom e classificados segundo os gêneros de 107 jogos educacionais para ensino de Computação.

Tabela 7– Relação entre taxonomia de Bloom (1965) e gênero de jogos.

Nível	Taxonomia de Bloom	Gênero de jogos
Nível cognitivo	1. <b>Conhecimento</b>	<i>Quiz</i> , <i>puzzle</i> , <i>roll-and-move</i> , estratégia
	2. <b>Compreensão</b>	Ação-aventura, <i>puzzle</i> , estratégia, RPG, Adivinhação
	3. <b>Aplicação</b>	Simulação, ação-aventura, corrida, RPG
	4. <b>Análise</b>	Simulação
	5. <b>Síntese</b>	Simulação
	6. <b>Avaliação</b>	Simulação
Nível psicomotor	1. <b>Percepção</b>	Ação-aventura, corrida
	2. <b>Prontidão</b>	Ação-aventura, corrida
	3. <b>Resposta Guiada</b>	RPG, Simulação
	4. <b>Mecanismo</b>	Simulação, <i>puzzle</i> , ação-aventura
	5. <b>Respostas complexas</b>	Estratégia, Simulação
	6. <b>Adaptação</b>	Simulação
	7. <b>Originalidade</b>	Simulação
Nível afetivo	1. <b>Receber</b>	Estratégia, RPG
	2. <b>Responder</b>	Simulação, <i>Quiz</i> , RPG
	3. <b>Valorizar</b>	RPG, simulação, ação-aventura
	4. <b>Organizar</b>	Simulação
	5. <b>Caracterizar</b>	Simulação

A **plataforma** é a classificação dos jogos de acordo com sua estrutura (digitais e não-digitais). Existe uma forte predominância de jogos educacionais digitais disponíveis na literatura, incluindo PC (*stand-alone* e online), *console games* ou *mobile games*. No entanto, jogos não-digitais também são utilizados para fins instrucionais, incluindo jogos de carta, tabuleiro, *papel & lápis* e acessórios. As plataformas dos jogos abordados nesta pesquisa estão descritas de acordo com a tabela 8 (HAINEY et al., 2016, CONNOLLY et al., 2012, CAULFIELD, VEAL e MAJ, 2011).

Tabela 8– Plataforma de jogos.

Classificação		Definição
<b>Jogo digital</b>		Jogo eletrônico que envolve a interação humana com uma interface de usuário para gerar <i>feedback</i> em um dispositivo eletrônico.
PC	<i>Stand-alone</i>	Jogo que é jogado em um computador de uso geral.
	<i>Online</i>	Jogo que é jogado em algum tipo de rede de computadores (internet), usando computadores.
<i>Console</i>		Jogo que é jogado em um dispositivo específico que conecta a uma televisão ou monitor de vídeo.
<i>Mobile</i>		Jogo que é jogado a partir de um dispositivo móvel, como celular, <i>tablete</i> , <i>smartphone</i> .
<b>Jogo não-digital</b>		Jogo que não é jogado em dispositivo eletrônico, sendo chamado também de jogo manual.
<i>Tabuleiro</i>		Jogo que envolve marcadores ou peças movidas ou colocadas sobre uma superfície pré-marcada ou tabuleiro, de acordo com um conjunto de regras.
<i>Cartas</i>		Jogo que usa cartas de baralho como recurso primário com o qual o jogo é jogado.
<i>Papel e lápis (Paper &amp; pencil)</i>		Jogo que pode ser jogado apenas com papel e lápis.
<i>Acessórios (prop)</i>		Jogo que é jogado com acessórios ou adereços (objetos portáteis).

**Modo de interação** é uma característica importante dos jogos que permite identificar a interação entre os jogadores. Portanto, o modo de interação é a maneira como o jogador pode interagir com os outros jogador(es) e/ou computador(es)/*console/mobile* (FULLERTON, 2008) (tabela 9).

Tabela 9– Modos de interação

Modo de Interação	Definição
<i>Singleplayer</i>	O jogo jogado por um único jogador, normalmente competindo contra ele próprio, buscando alcançar um objetivo pré-definido.
<i>Multiplayers</i>	O jogo com múltiplos jogadores, interpretado por diversos adversários competindo entre si.

<b>Multigrupos</b>	O jogo jogado por grupos de jogadores competindo com outros grupos.
--------------------	---

**As regras** são restrições contidas no jogo que visam orientar o jogador, determinando condutas corretas ou erradas. As regras são as normas, orientações, fluxo de ações, ou situações que limitam a atuação do jogador dentro do jogo. Elas definem a qualidade do jogo e apresentam um número fixo de diretrizes abstratas que descrevem como o jogo deve funcionar (SALEN e ZIMMERMAN, 2009). No entanto, regras complexas podem influenciar o jogador a abandonar o jogo. E regras excessivamente simples também podem levar o jogador a desistir do jogo, por falta de desafio ou motivação (SALEN e ZIMMERMAN, 2009, FULLERTON, 2008).

A **mecânica** do jogo são as ações do personagem realizada durante o jogo, elas são regras específicas para limitar o que o personagem pode realizar durante o jogo. Um exemplo de mecânica do jogo digital: utilizar o teclado do computador para correr, saltar, caminhar e abaixar o personagem no cenário do jogo. Um exemplo de mecânica de jogo não-digital: jogar dois dados para movimentar a ficha do jogador (BRATHWAITE e SCHREIBER, 2009).

A **narrativa** é uma história com narrações ordenadas, descrevendo acontecimentos ocorrido(s) no(s) cenário(s) onde se passa o jogo. Na narrativa, descreve-se o local aonde vivem os personagens, a época, as relações entre os personagens do jogo. A narrativa é o enredo do jogo que precisa motivar o jogador a se tornar cada vez mais imerso (KREMERS, 2009).

**Os elementos** do jogo são os personagens, artefatos ou cenários que integram o ambiente do jogo. A seguir são apresentados os elementos básicos em um jogo tendo como base design de jogos (FULLERTON, 2008; KREMERS, 2009; SCHELL, 2008):

- **Personagem:** O personagem é um papel fictício utilizando por um ator ao longo do jogo. Um personagem deve ter um nome, tipo (jogador, adversário, máquina) e uma descrição das suas ações no jogo.
- **Artefatos:** Os artefatos são objetos que terão maior interação com o jogador, normalmente sendo atribuído algum

ponto, por exemplo, o jogador precisa percorrer uma nave espacial buscando peças da nave, ou então, em um jogo de corrida o jogador deve pegar o maior número de moedas de ouro.

- **Cenários:** Os cenários são conjuntos de objetos ou acessórios que ocupam a cena do jogo. Eles são enredos ou ambientes nos quais os jogadores movimentarão seus personagens permitindo deste modo prosseguir e avançar no jogo. São exemplos de cenários: castelos, cidades, naves espaciais, pista de corrida.





### 3 ESTADO DA ARTE

Atualmente muitos jogos são desenvolvidos e utilizados no ensino de Computação nas mais diversas instituições de ensino. Neste contexto, busca-se responder as duas questões de pesquisa que norteiam a Tese: 1. Quais são os jogos educacionais utilizados nas diversas áreas de conhecimento da Computação e 2. Quais são os processos utilizados para o desenvolvimento de jogos educacionais. Para responder estas questões, são apresentadas a seguir duas revisões sistemáticas da literatura objetivando também identificar o estado da arte de jogos para ensino em Computação.

#### 3.1 JOGOS EDUCACIONAIS ENCONTRADOS

Nesta seção apresenta-se a revisão sistemática da literatura que tem por objetivo apresentar as informações sobre jogos para ensino em Computação. A revisão da literatura também é apresentada no artigo *Games for Teaching Computing in Higher Education – A Systematic Review* (BATTISTELLA e WANGENHEIM, 2016c).

A metodologia utilizada para realizar a revisão sistemática da literatura é fundamentada pelo método proposto por Kitchenham (2004). Especificamente busca-se responder as seguintes perguntas de pesquisa:

1. Quais jogos são utilizados para o ensino em Computação e em quais áreas de conhecimento?
2. Quais são os objetivos de aprendizagem?
3. Quais tipos de jogos são utilizados?
4. Como os jogos são desenvolvidos e avaliados?

#### **Definição do protocolo da revisão da literatura**

Foram examinados todos os materiais na língua inglesa (p. ex. artigos, manuais, sites) que tratassem de jogos para ensino de Computação disponíveis na web entre 2000 e 2013. Na pesquisa foram considerados os jogos desenvolvidos antes de 2000 desatualizados e também os jogos até o início do ano de 2013, sendo a data em que a revisão sistemática foi realizada.

**Critérios de inclusão e exclusão:** A revisão sistemática abrange o ensino em qualquer competência (conhecimento, habilidade ou atitude) na área da Computação. No entanto, foram excluídos da pesquisa os jogos destinados exclusivamente para treinar os jogadores para um produto de uma empresa e também jogos que não tratassem de conteúdos da Computação.

Considera-se qualquer tipo de jogo educacional, incluindo jogos digitais e não-digitais. Por outro lado, foram excluídos quaisquer tipos de jogos utilizados como outro tipo de método de ensino, por exemplo, estudo de caso de um problema específico, exercícios ou simulações baseado em projetos. Também foram excluídos os jogos que ensinavam a jogar jogos comerciais, como palavras cruzadas e *tetris*, e foram excluídas as simulações, como as plataformas para ensino em inteligência artificial em que os alunos programam o jogo.

Foca-se nos jogos que trabalham com a aprendizagem baseada em jogos, sendo excluído qualquer material sobre aprendizado que não tratasse de área de jogos educacionais, por exemplo, plataformas digitais nas quais os alunos devem programar um jogo, sem a necessidade de apresentar aspectos do design instrucional. Também foram excluídas pesquisas nas quais os alunos implementam jogos. Por exemplo, programação de um jogo como exercício para a matéria de introdução a programação. Nosso foco também está voltado ao ensino superior, nas Instituições de Ensino Superior, principalmente em nível de graduação.

Também foram excluídos materiais que apenas descrevem uma proposta de jogo ou modelo de jogo que será desenvolvido. E foram considerados os jogos que apresentassem pelo menos uma descrição em inglês, disponível na web.

Em termos de qualidade das informações encontradas, o material disponível precisa ser suficientemente compreensível. Também foram excluídos jogos citados em materiais disponíveis na internet, os quais não foram encontrados em pesquisas específicas.

**Fontes de dados e estratégias de busca:** As pesquisas foram realizadas na web via a ferramenta de busca *Google*<sup>1</sup> a fim de obter uma ampla visão, não só de jogos publicados em artigos científicos, mas também de jogos comerciais. Foram procurados os termos jogos e educação (foi usado também derivações do termo, como aprendizagem, ensino, treinamento) em combinação com outros termos referentes às áreas de conhecimento da Computação.

---

<sup>1</sup> [www.google.com](http://www.google.com)

**Termos de busca:** *game AND(education OR educational OR learning OR teaching OR training OR “serious game”) AND (“software requirements” OR “software architecture” OR “software design” OR “software engineering” OR “software process” OR “software quality” OR “software verification” OR “software validation” OR “software testing” OR “software maintenance” OR computing OR “computer science” OR “information systems” OR “Information Technology” OR IT OR “artificial intelligence” OR algorithms OR “data structures” OR “programming languages” OR “computer network” OR network OR compilers OR “computer architecture” OR “computer graphics” OR “database” OR “distributed systems” OR “digital logic” OR “embedded system” OR “human-computer interaction” OR “operating systems” OR “security” OR “system administration” OR “numerical methods”)*

### **Execução da busca**

As pesquisas utilizando a ferramenta de busca *Google* iniciaram em dezembro de 2012 e finalizaram em janeiro de 2013. Utilizando os termos de busca, ao total foram encontrados 1.506.000.000 de resultados. Para cada busca realizada, foram analisados os 300 primeiros resultados apresentados pelo *Google*. Entende-se que resultados acima de 300 apresentam baixa relevância.

Na primeira fase de pesquisa, foram analisados os títulos e resumos considerando os critérios de inclusão e exclusão, sendo selecionados 194 jogos. Entretanto, foram encontradas diversas simulações ou jogos que estavam fora do escopo de pesquisa. Ao longo das pesquisas foram realizadas buscas secundárias em cima de materiais encontrados, por exemplo, ao identificar um artigo falando superficialmente de um jogo, foi pesquisado sites e outros materiais que detalhassem melhor o jogo. Na segunda fase os 194 jogos foram pesquisados com maior profundidade, em alguns casos os jogos foram jogados, tendo como resultado final apenas 107 jogos considerados relevantes (Apêndice A).

Para extração de dados, enquadraram-se os jogos nas dimensões listadas na tabela 10, conforme seção de fundamentação teórica.

Tabela 10– Dimensões para extração de dados.

Dimensão		Descrição
Nome		Nome do jogo
Áreas de conhecimento da Computação		Identificação da área de conhecimento da Computação conforme ACM/IEEE-CS (ACM/IEEE-CS, 2005).
Foto/ <i>Screenshot</i>		Uma foto do material o <i>screenshot</i> do jogo.
Descrição do jogo		Breve descrição do jogo.
Referências		Referências do jogo, seja eles artigos, sites ou materiais que tragam informações sobre o jogo.
Objetivos de aprendizagem		Identificação dos objetivos de aprendizagem, os quais o aluno precisa alcançar após jogar.
Níveis dos objetivos de aprendizagem		Classificação dos objetivos de aprendizagem (conhecimento, habilidades e atitudes) como definidos nas tabelas 2, 3 e 4.
Público-alvo		Identificação do público-alvo (p. ex.: graduandos de Computação, profissionais, Professores).
<i>Feedback</i> sobre a aprendizagem		Modo como o jogo apresenta <i>feedbacks</i> aos alunos.
Mídia		Classificação em digital ou não-digital.
Plataforma		Subclassificação em tipos de jogos como definido na tabela 7.
Gênero		Classificação dos gêneros com base na tabela 8.
Modo de interação		Identificação dos modos de interação, com base na tabela 9.
Duração das sessões de jogos		Identificação de uma sessão de uma típica de jogo.
Linguagem		Identificação das línguas em que o jogo está disponível.
Licença		Identificação da licença sob a qual o jogo é publicado.
Custo (US\$)		Custo para adquirir o jogo e/ou material do jogo, em US\$.
Design instrucional		Identificação de abordagens de design instrucional adotado.
Desenvolvimento de jogos		Identificação do processo de desenvolvimento do jogo.
Avaliação de jogos	Variáveis	Identificação das variáveis de avaliação.
	Níveis de avaliação	Classificação do nível de avaliação, em conformidade ao modelo de quatro níveis de Kirkpatrick (KIRKPATRICK e KIRKPATRICK, 2006), apresentados na tabela 5.
	Design da pesquisa	Classificação do tipo de design de pesquisa, seguindo designs de pesquisas usadas no contexto da educação, conforme Apêndice E - A2.4 Revisar o modelo de avaliação do jogo.
	Instrumentos para coleta de dados	Identificação do tipo de instrumento utilizado para coleta de dados, como, entrevistas, observação, questionário, teste.

O material foi analisado por meio da leitura dos artigos e sites, sendo que alguns jogos disponíveis para acesso livre também foram jogados. Entretanto, a extração dos dados, principalmente em relação à aspectos didáticos foi prejudicada em muitos casos pela maneira em que os dados foram descritos. A maioria das pesquisas não apresentavam informações básicas, como não fornecer informação sobre design

instrucional. Nos casos de jogos digitais faltam informações de como estes jogos foram desenvolvidos. Se disponível, as descrições de avaliações dos jogos não apresentavam detalhes.

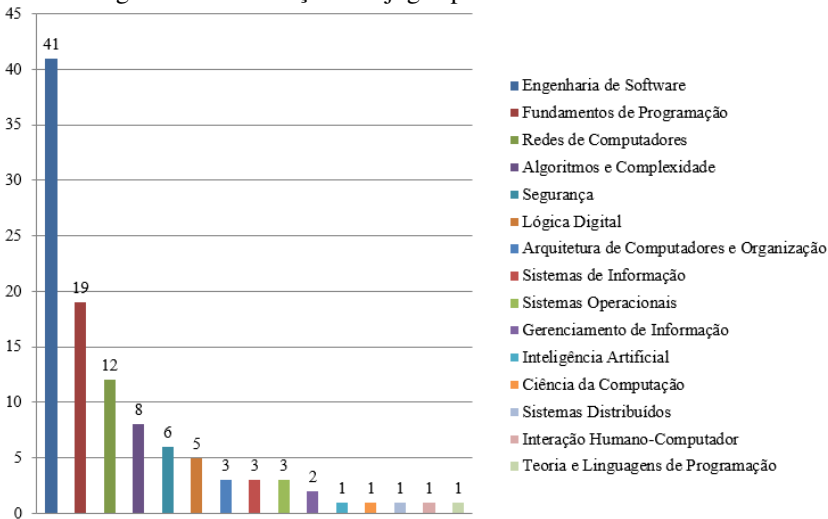
## Extração da informação

Nesta seção apresenta-se os resultados da revisão sistemática da literatura, demonstrando o atual estado da arte de jogos para o ensino de Computação no ensino superior.

### (1) Quais jogos são utilizados para o ensino em computação e em quais áreas de conhecimento?

Foram encontrados 107 jogos para ensino em Computação, isso demonstra que existe uma tendência para a aprendizagem baseada em jogos na área da Computação. O apêndice A apresenta os 107 jogos identificados na revisão sistemática da literatura, contendo nome do jogo, área de conhecimento da Computação, *screenshot*, descrição do funcionamento do jogo e referências. Na figura 4 apresenta-se a quantidade de jogos identificados na revisão sistemática da literatura, distribuídos nas áreas de conhecimento da Computação.

Figura 4– Distribuição dos jogos por área de conhecimento.



Fonte: Elaborada pelo autor.

Foram identificados em termos de área de conhecimento, que a grande parte dos jogos concentra-se na área de Engenharia de Software, sendo ao total 41 jogos. Este fato, pode estar relacionada a necessidade específica da Engenharia de Software de aplicar os conceitos de modo prática (GKRITSI, 2011). Como muitas vezes não há tempo suficiente nas disciplinas de Engenharia de Software para proporcionar aos alunos uma sólida compreensão e experiência prática, outros métodos de ensino, incluindo jogos podem ser tornar uma alternativa importante. Na área de Engenharia de Software grande parte dos jogos são do gênero de simulação, como (BAKER et al., 2003), (DANTAS et al., 2004), (SHAW e DERMOUDY, 2005), (SCRUM LEGO CITY, 2009), (ZWIKAEL e GONEN, 2007). Nestes jogos o jogador assume o papel de um gerente de projetos e, em seguida, tem que planejar, executar, monitorar e controlar um projeto de software.

Outras áreas de conhecimento as quais existem uma quantidade considerável de jogos são: fundamentos de programação, redes de computadores, algoritmos e segurança. No caso de jogos para a área de fundamentos de Computação, objetiva-se auxiliar os alunos novatos na aprendizagem básica de programação, como, por exemplo, o jogo Wu Castelo (EAGLE e BARNES, 2009), no qual o jogador precisa instanciar laços e em seguida ver o resultado rodando no cenário do jogo, sem exigir codificação.

Similar à área de Engenharia de Software, os jogos para a área de redes de computadores também são, normalmente, simulações em que o trabalho do jogador é identificar problemas em redes e sub-redes e corrigi-las para que funcionem adequadamente (SUBTNET GAME, 2008). Nesta área de conhecimento, os jogos permitem uma simulação de um cenário real, ensinando desta forma competências exigidas para esta área. O mesmo ocorre para jogos para área de segurança.

Jogos para aprendizagem nas áreas de algoritmo e complexidade são fortemente focados em ter o estudante executando diferentes algoritmos e ter experiências com a complexidade de algoritmos, tipicamente com uma competição entre os jogadores. Assim, eles representam uma forma de envolver os alunos em uma aprendizagem ativa e também em uma forma divertida a fim de contribuir para a compreensão e memorização do conhecimento a ser aprendido.

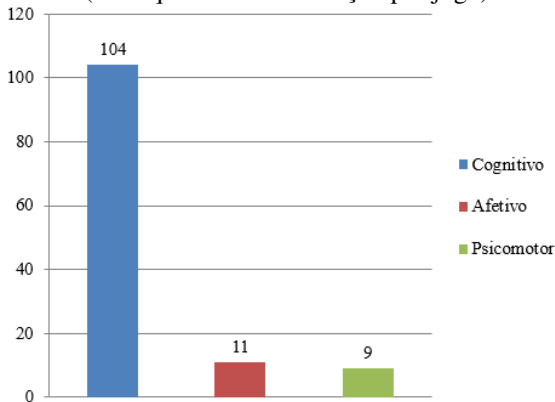
Outra área para a qual foram identificados um bom número de jogos é a área de lógica digital. Nesta área, normalmente os jogos precisam demonstrar o funcionamento de um circuito elétrico, funcionando como simuladores nos quais o jogador precisa corrigir ou construir componentes ou circuitos.

Para várias áreas de conhecimento, incluindo, gestão em tecnologia da informação (TI) interação homem-computador, teoria de linguagens de programação, poucos jogos foram encontrados ( $\leq 3$ ). Isso demonstra que um número considerável de jogos educacionais encontram-se em um pequeno número de áreas de conhecimento.

## (2) Quais são os objetivos de aprendizagem?

Com base nas informações coletadas dos materiais dos jogos, foram classificados os objetivos de aprendizagem utilizando as taxonomias definidas no capítulo 2 - Fundamentação Teórica. As classificações dos jogos foram feitas na maioria dos casos com base no entendimento do que foi lido nos materiais, porque em grande parte dos jogos e documentos os autores não explicavam claramente os objetivos de aprendizagem. No apêndice B são apresentadas as classificações para os 107 jogos encontrados contendo as seguintes informações: objetivos de aprendizagem, níveis dos objetivos de aprendizagem e *feedback*. Os objetivos de aprendizagem estão disponíveis nas figuras 5, 6, 7 e 8.

Figura 5– Distribuição por tipo de objetivos de aprendizagem (mais que uma classificação por jogo).



Fonte: Elaborada pelo autor.



Figura 6– Nível de objetivos de aprendizagem: Cognitivo.

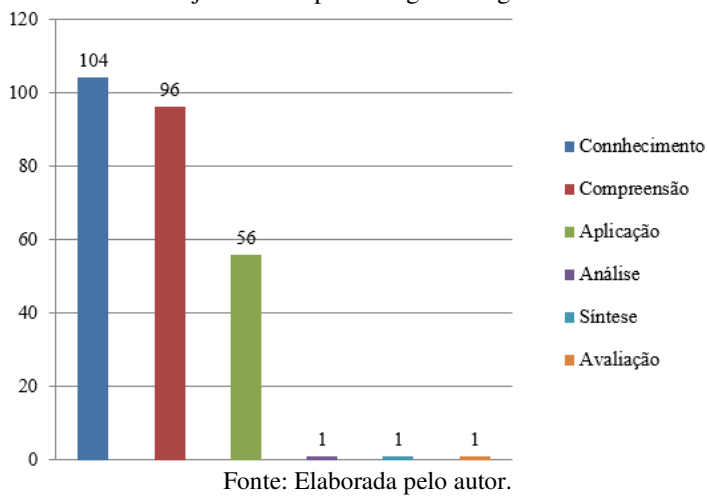


Figura 7– Nível de objetivos de aprendizagem: Afetivo.

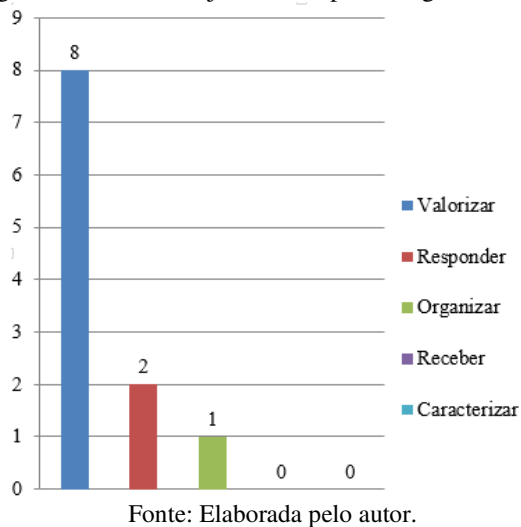
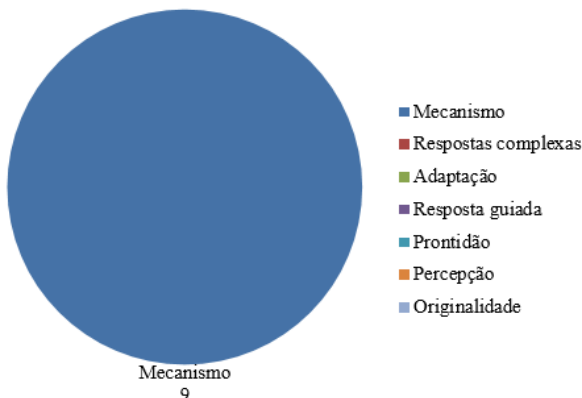


Figura 8– Nível de objetivos de aprendizagem: Psicomotor.



Fonte: Elaborada pelo autor.

Analisando os tipos de objetivos de aprendizagem é possível observar que a grande maioria dos jogos tem como objetivo o conhecimento em nível cognitivo da Taxonomia de Bloom. Apenas um pequeno número de jogos visa ensino de habilidades, como a comunicação ou trabalho em equipes (YE et al., 2007) e (HAKULINEM, 2011), ou uma mudança de atitude como reconhecer as dificuldades existentes na gerência de equipes (CARVALHO, 2012).

Considerando cada nível de objetivos de aprendizagem separadamente, é possível observar que a maioria dos jogos visam níveis cognitivos mais baixos da Taxonomia de Bloom, de acordo com as expectativas de aprendizagem em níveis de graduação (ACM/IEEE-SE, 2004). Neste contexto, os jogos foram mais usados para reforçar os conhecimentos previamente ensinados utilizando diferentes métodos de ensino (designs instrucionais distintos). O nível de aprendizagem psicomotor/habilidades apresentou apenas jogos em termos de mecanismos visando a aprendizagem de certas habilidades, como comunicação, trabalho em equipe, tomada de decisão ou resolução de problemas. Os objetivos de aprendizagem afetivos visam principalmente o reconhecimento de certos valores ou a criação de novos sistemas de valores.

A maioria dos jogos são destinados a ser jogados apenas uma vez, principalmente os jogos não-digitais. Uma exceção é o jogo *Hard Choices* (BROWN et al., 2010), o qual é necessário jogá-lo duas partidas para melhor compreendê-lo. Mesmo alguns jogos digitais (por exemplo: (ZWIKAEL e GONEN, 2007)) são projetados para

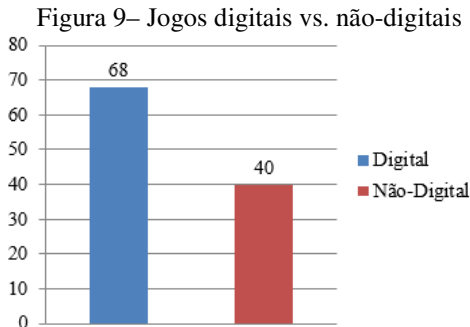
acompanhar o processo de aprendizagem em diferentes níveis, sempre crescendo em termos de dificuldade do jogo.

Outro fator em relação aos objetivos de aprendizagem é a forma como o *feedback* é dado aos alunos, em relação ao seu desempenho durante o jogo. Embora, diversos jogos digitais apresentem resultados ou um *log* de erros ao fim do jogo, eles não apresentam uma orientação sobre o que foi feito de errado e como corrigir os erros. Em casos de jogos não-digitais o *feedback* é dado por meio do Professor.

### (3) Quais tipos de jogos são utilizados?

De modo geral, apresenta-se no apêndice C as características dos tipos de jogos selecionados, sendo informado o tipo de mídia, plataforma, gênero, modo de interação, duração das seções de jogo, linguagem, licença e custo.

Como fator típico em outras áreas de conhecimento, a maioria dos jogos encontrados para área da Computação são digitais, sendo encontrados na revisão sistemática 68 jogos digitais e 40 não-digitais, conforme apresentado na figura 9.



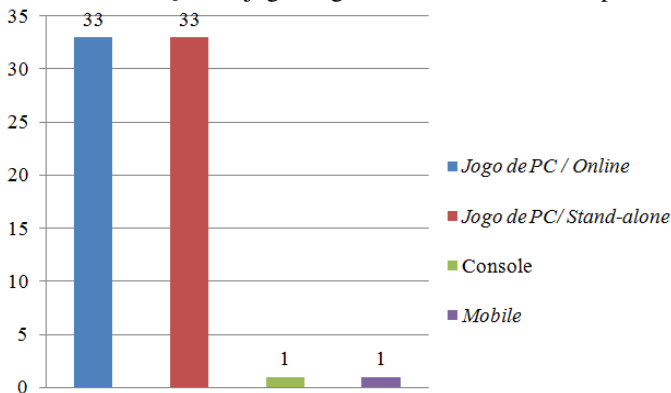
Fonte: Elaborada pelo autor.

No entanto, um número considerável de jogos não-digitais existe e está crescendo em termos de popularidade, uma tendência que também foi observada na indústria do entretenimento (CORE, 2013). Aparentemente o fato de que eles podem ser jogados em grupo traz vários benefícios, como aumento da interação social, concorrência e engajamento. Como muitos jogos digitais são *singleplayer*, eles não trazem estas reações tão fortemente. Isso também fornece um benefício muitas vezes esquecido ao utilizar jogos: a oportunidade de compartilhar

experiências e trocar conhecimentos. Outra grande vantagem é que os jogos não-digitais apresentam um limiar baixo de desenvolvimento, enquanto jogos digitais exigem mais esforço para seu desenvolvimento (KAIN, 2012). Além disso, eles também são mais fáceis de serem adaptados ou melhorados conforme as necessidades de uma unidade instrucional, seja mudando algumas regras, inserindo exercícios ou modificando algo específico do jogo a fim de adequar aos objetivos de aprendizagem.

Analisando os tipos de jogos digitais com mais detalhe, é possível observar que o número predominante deles são jogos para plataformas de PCs (figura 10). E também, ainda existem vários jogos *stand-alone*, mas a tendência são jogos online. Foram identificados alguns jogos que utilizam redes sociais com *Second Life*<sup>2</sup> (YE, CHANG e POLACK-WAHL, 2007) ou publicações via *Facebook*<sup>3</sup> (LEONG, KOH e RAZEEN, 2011). Foi identificado também que videogames não são utilizados para o ensino de Computação, com exceção do jogo *Project Execution Game (PEG)* (MORSI e RICHARDS, 2012). Até o momento, foi identificado que jogos para *smartphones* ou celulares são exceção, como *Cargo Bot* (CARGOBOT, 2012).

Figura 10– Distribuição de jogos digitais de acordo com sua plataforma.



Fonte: Elaborada pelo autor.

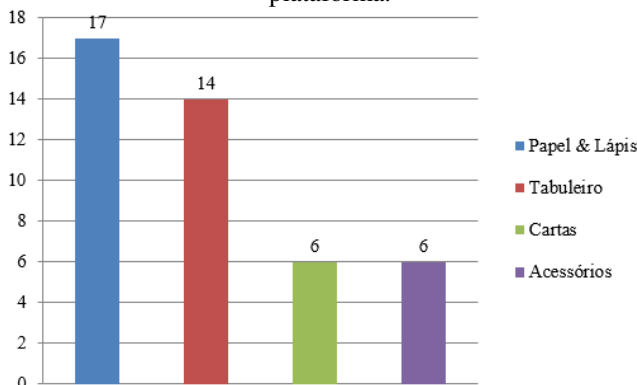
Em relação aos jogos não-digitais, sua grande maioria são do tipo *Papel & Lápis*. Existe também um grande número de jogos de tabuleiro, seguindo por jogos de cartas. Menos comuns são os *prop games*

<sup>2</sup><http://secondlife.com>

<sup>3</sup> <http://facebook.com>

(acessórios), que utilizam outros objetos como Lego© (SYBERFELDT e SYBERFELDT, 2010) ou laranjas (ORANGE GAME, 2002).

Figura 11– Distribuição de jogos não-digitais de acordo com sua plataforma.

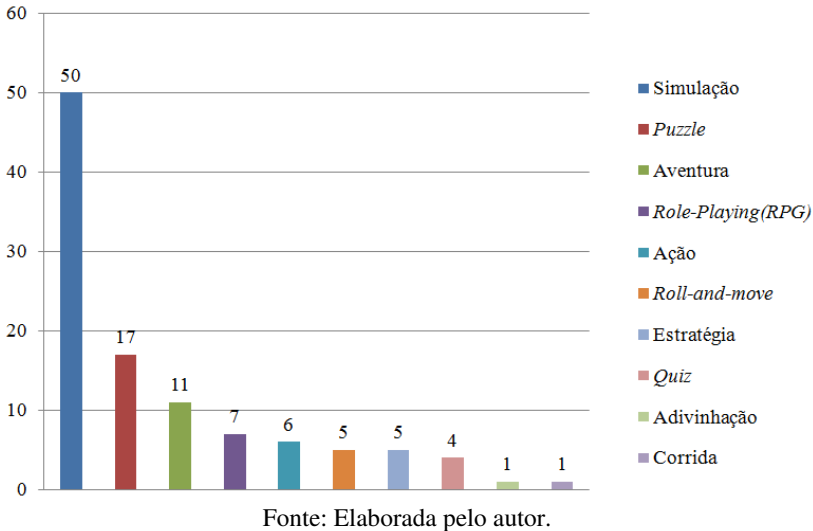


Fonte: Elaborada pelo autor.

Em termos de gênero, é possível perceber que a maioria são simulações (figura 12). Isso reflete a tendência no uso de jogos de simulação para o ensino e com isso os Professores visam aplicar um conhecimento de modo prática e também desafiam os alunos a tomarem decisões críticas com base no conhecimento teórico e prático. Simulações são aceitas como modo complementar métodos tradicionais de ensino, estimulando e motivando o aprendizado por meio de um ambiente realista para os estudantes.

Além disso, como estes jogos representam uma combinação de simulação que permite visualizar conceitos complexos, eles permitem uma abordagem de aprendizagem ativa provocando uma aprendizagem experimental, mantendo os alunos envolvidos (MITCHELL e SAVILL-SMITH, 2004), (KIRRIEMUIR, 2003). Eles podem envolver os alunos em uma experiência simulada mais agradável do que pode ocorrer no mundo real (KIRRIEMUIR, 2003). Especialmente para o ensino de Engenharia de Software a maioria dos jogos são de simulação (PEIXOTO et al., 2011), como SimSE (NAVARRO e HOEK, 2004) um jogo na qual o usuário é o gerente de projetos e tem por objetivo conduzir uma equipe para o desenvolvimento de softwares.

Figura 12– Gênero dos jogos.



Vários jogos para área de redes de computadores também são de simulação, como Cyber CIEGE (CYBERCIEGE, 2011). Neste jogo os jogadores configuram estações de trabalho, servidores e sistemas operacionais. Em seu cenário mais longo, os usuários avançam por uma série de etapas e devem proteger a empresa a qual recebe cada vez mais ataques.

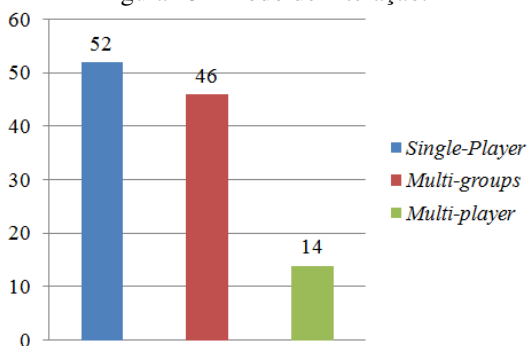
Além disso, muitos jogos não-digitais também são simulações, como SCRUMIA (SCRUMIA, 2009) ou *Requirements Activity: Lego City* (SCRUM LEGO CITY, 2009), nos quais os processos são executados pelos alunos manualmente. Nestes jogos os alunos precisam construir objetos, como chapéus de papel, casas de Lego, aplicando os conceitos relevantes da disciplina a partir da área de conhecimento a qual faz parte, por exemplo, em gerência de projetos.

Também foram encontrados vários jogos do gênero *puzzle*, principalmente para área de conhecimento de programação, nos quais os jogadores precisam programar robôs ou objetos para solucionar um problema, como navegar em um labirinto, por exemplo, Robot Trouble (ROBOT TROUBLE, 2011), Robozzle (ROBOZZLE, 2011), Treasure Hunt (TREASURE HUNT, 2011). Existe também, um bom número de jogos do gênero de aventura, na qual o jogador participa de uma aventura interativa, e tem que resolver desafios e missões a fim de ganhar o jogo. Exemplos são hACME (NERBRATEN e ROSTAD,

2009), *Secret Testing Ninja* (BELL, SHETH, e KAISER, 2011) Sistema Digital (SRINIVASAN, BUTLER-PURRY e PEDERSEN, 2008). Além disso, também encontra alguns jogos de RPG usados principalmente para ensinar conceitos de Computação, como *Saving Princess Sera* (EAGLE e BARNES 2009), JV2M (GÓMEZ-MARTÍN et al., 2006), *EleMental: The Recurrence* (CHAFFIN et al., 2009).

Em termos de modo de interação é possível observar que a maioria dos jogos digitais são *singleplayer* (Figura 13). Entretanto, esta ênfase nos jogos *singleplayer* significa que o *jogo social* (modo de interação social) e a interação entre os jogadores foi excluída como funcionalidade do jogo. Até o momento, foram poucos os jogos classificados como *multiplayer*, os quais oferecem a oportunidade de modo parcerias, competição ou rivalidade e, assim, aumentando o prazer pelo jogo. Este tipo de interação social proporcionada pelos jogos não-digitais é também um dos seus pontos mais fortes. Quanto aos jogos não-digitais, há normalmente uma atmosfera emocionante de comunicação e concorrência, que muitas vezes, além dos objetivos de aprendizagem também incentivam a expressão verbal e ao mesmo tempo cultiva habilidades sociais e traços de paciência e persistência, como (ITZHAKI e MATTHEWS, 2013) e (WANGENHEIM, CHRISTIANE, SAVI e BORGATTO, 2012).

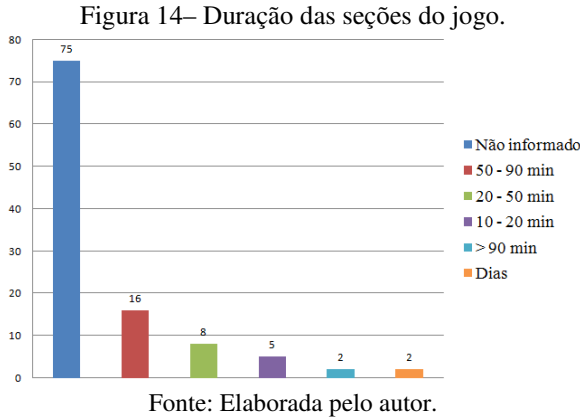
Figura 13– Modo de interação.



Fonte: Elaborada pelo autor.

Algumas descrições de jogos identificam claramente a duração da aplicação do jogo. Entretanto, para os jogos em que esta informação não está disponível, parecem ser feitos para a duração de curtos períodos. Foram encontrados poucos jogos com duração de 90 minutos ou que levassem dias para ser concluídos, por exemplo, jogos em que o

aluno pode levar um semestre para concluir o jogo (Figura 14). De modo geral, entende-se que estes jogos, com duração menor, podem ser aplicados em uma aula (unidade instrucional).



Os jogos para ensino em Computação parecem ser mais concebidos para uma aplicação pontual, em vez de todo o contexto instrucional, ou seja, estes jogos são utilizados para reforçar a aprendizagem em um conteúdo específico. Poucos jogos são criados para acompanhar uma unidade instrucional inteira, como Z-Buffer (KUK et al., 2012), o qual apresenta vários níveis e exige aprofundamento do conteúdo a medida que se joga o jogo. Este fato se deve para garantir uma ordem didática dos temas que são introduzidos para os alunos, conforme o ritmo da aula.

Em termos de custos, foi identificado que a maioria dos jogos possui acesso livre e os que apresentam custos, os valores são baixos. Outro fator importante para adoção dos jogos em sala de aula é a licença para o uso do jogo, mas considerando os valores baixos destes jogos, é possível considerá-los como uma boa opção para empregá-los em sala de aula.

#### (4) Como os jogos são desenvolvidos e avaliados?

O design de jogos educacionais pode ser conceituado como a interseção da teoria de aprendizagem, teoria de design de jogos e a experiência do professor sobre o assunto. Entretanto, foi identificado que a maioria dos jogos não inclui descrições de como os jogos foram desenvolvidos ou avaliados, não apresentando as teorias utilizadas de



design instrucionais e design de jogos. No apêndice D apresenta-se os jogos encontrados na revisão sistemática da literatura que foram desenvolvidos e avaliados.

Em relação aos objetivos instrucionais destes jogos, durante as pesquisas, buscou-se descrições de como os jogos eram alinhados às teorias instrucionais. Contudo, não foi encontrada na maioria dos casos nenhuma descrição. Desta forma, informações como contexto educacional, público-alvo, estratégias de ensino raramente foram fornecidas.

Como exceção foi encontrado o jogo Anti-Phishing Phil (SHENG et al., 2007). Outros exemplos como (MELERO; HERNÁNDEZ-LEO e BLAT et al., 2012), (THIRY; ZOUKAS e SILVA, 2011), (SCRUMIA, 2009) os quais citam ADDIE (MOLENDIA et al., 1996, BRANCH, 2009) ou ISD *Model* (DICK e CAREY, 1996) como design instrucional. No caso do Z-Buffer (KUK et al., 2012) foi utilizado mapas conceituais para sua concepção.

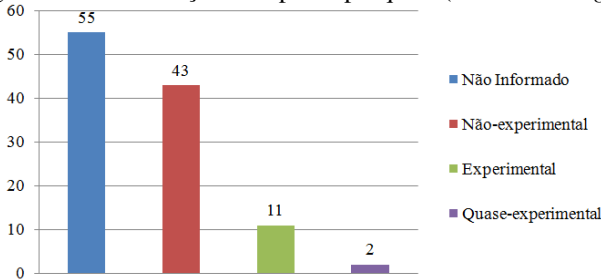
Por outro lado, alguns jogos apenas descrevem a utilização de processos de desenvolvimento de software, como (DO e LEE, 2009), (MORSI e RICHARDS, 2012), (GÓMEZ-MARTÍN et al., 2006). Ou então, descrevem detalhes de implementação ou indicando plataformas e tecnologias (BELFORE-II, ADCOCK e WATSON, 2009), (JORDAN, 2006), (BARNES et al., 2007), (EAGLE e BARNES, 2009), (HARPER, MILLER e SHEN, 2011) e (ROSSIOU e PAPADAKIS, 2007).

De modo geral, foi identificado que não são utilizados conceitos fundamentais da teoria de jogos (design de jogos) para o desenvolvimento dos jogos encontrados na revisão sistemática. Entretanto, foram encontrados em alguns poucos documentos de modo explícita a utilização de elementos do design de jogos, como competição, recompensa, formato da interface, narrativa.

Contudo, foi identificada uma tendência destes jogos serem avaliados, apesar de muitas vezes apresentarem uma abordagem não-experimental (figura 15). Tipicamente se aplica o jogo em sala de aula e em seguida os alunos respondem uma avaliação (pré-teste).

Destaca-se que o jogo SimSE apresentou uma série de experimentos (NAVARRO e HOEK, 2007), (NAVARRO, 2006) e os jogos SESAM (DRAPPA e LUDEWIG, 2000) e Cyber CIEGE (CYBERCIEGE, 2011) apresentaram avaliações quase-experimentais.

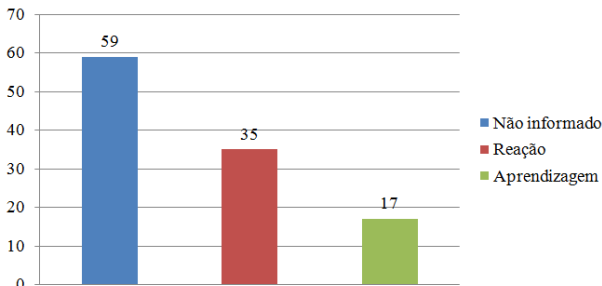
Figura 15– Distribuição do tipo de pesquisa (*research design*).



Fonte: Elaborada pelo autor.

Quando se mapeou os níveis de avaliação de Kirkpatrick, foram identificadas as avaliações com predomínio no nível 1 (reação), realizando avaliações com base na percepção subjetiva dos alunos (figura 16). Também foram identificadas poucas avaliações no nível 2 (aprendizagem) comparando os pré-testes e pós-testes após o jogo. Não há resultados longitudinais para analisar a transferência real de competências aprendidas no contexto profissional.

Figura 16– Distribuição das avaliações, segundo modelo Kirkpatrick.



Fonte: Elaborada pelo autor.

Em relação aos objetivos da avaliação, a maioria dos estudos se classifica como pesquisas exploratórias buscando identificar a eficácia da aprendizagem e agradabilidade e pontos fortes e fracos do jogo.

Poucos estudos compararam a eficácia da aprendizagem do jogo com outros métodos de estudo, como (ROSSIOU e PAPADAKIS, 2007). Entretanto, nas avaliações subjetivas, muitos alunos afirmaram preferirem os jogos a outros métodos de estudos tradicionais.

## Discussão

Por meio da revisão sistemática da literatura foi identificado que há uma tendência em utilizar jogos para ensinar Computação nas Instituições de Ensino Superior, principalmente por haver uma diversidade em termos de gêneros e quantidade de jogos. Como resultado observa-se:

1. Os jogos são utilizados principalmente para área de Engenharia de Software, Fundamentos de Programação, Redes de Computadores, Algoritmo e Complexidade, e Segurança.

2. A maioria são jogos digitais, principalmente para PCs, mas também existe uma tendência para produção de jogos não-digitais (p. ex. *papel&lápis*, tabuleiro).

3. Grande parte dos jogos possui o gênero de simulação, principalmente os jogos para área de Engenharia de Software e Redes de Computadores.

4. A maioria dos jogos visa uma aprendizagem em níveis cognitivos mais baixos e parecem ser usados para reforçarem o conhecimento.

5. A maioria dos documentos que descrevem os jogos não apresenta detalhes de como os jogos foram desenvolvidos, se utilizaram design instrucional ou design de jogos.

6. Os jogos também não são avaliados comumente e quando são, a grande parte das pesquisas são não-experimentais.

### 3.2 PROCESSOS DE DESENVOLVIMENTO DE JOGOS EDUCACIONAIS ENCONTRADOS

Esta seção apresenta a revisão sistemática da literatura dos processos de desenvolvimento de jogos educacionais. A revisão da literatura também é apresentada no artigo *Como jogos educacionais são desenvolvidos? Uma revisão sistemática da literatura* (BATTISTELLA, WANGENHEIM, e FERNANDES, 2014).

Uma das primeiras perguntas de pesquisa que buscou-se verificar são os processos de desenvolvimento de jogos educacionais existentes e como são utilizados. Para isso, entende-se que eles precisam conter aspectos educacionais, onde tais jogos devem aparentemente ser desenvolvidos baseados em algum tipo de processo de design instrucional. Estes processos tipicamente envolvem as fases de: *Análise, Design, Projeto, Desenvolvimento, Implementação e Avaliação*, conforme p.ex. ADDIE (MOLEND, 2003, BRANCH, 2009).

Porém, por outro lado, levando em consideração as características intrínsecas dos jogos, também considera-se aspectos de jogos, como competição, interação, entretenimento e diversão. Assim, os processos de desenvolvimento de jogos educacionais também devem envolver tipicamente fases como *Brainstorming*, *Protótipo Físico*, *Apresentação (opcional)*, *Protótipo de Software*, *Concepção da documentação do jogo e Produção* Fullerton (2008).

### **Definição do protocolo da revisão da literatura**

O objetivo da revisão da literatura é levantar o estado da arte para responder às seguintes perguntas de pesquisa: (a) quais processos existem para desenvolver jogos educacionais? (b) quais são as etapas/fases/atividades desses processos? (c) quais similaridades ou diferenças existem entre os processos existentes?

Definiu-se como ferramenta de busca o *Google Acadêmico*<sup>4</sup>, porque esta ferramenta torna possível a busca simultânea em diversos meios de publicação científicas online.

Na revisão da literatura dos jogos educacionais foi utilizado *Google* porque buscou-se identificar artigos e sites de jogos educacionais. Diferentemente da revisão da literatura dos processos de desenvolvimento que limitou a busca a artigos científicos.

**Crítérios de inclusão/exclusão:** São considerados apenas os artigos que se enquadravam nos seguintes critérios: (a) Escritos na língua inglesa, (b) Publicados entre 1993 e 2013, considerando artigos anteriores desatualizados; (c) Descrevem as fases utilizados no processo de desenvolvimento de jogos educacionais. Para não reduzir demais o espaço de busca não limitou-se os processos exclusivamente a jogos educacionais para Computação, mas leva-se em consideração também processos para o desenvolvimento de jogos relacionados a outros domínios de conhecimento.

São excluídos da pesquisa os artigos que: (a) Mostram apenas os jogos educacionais, e não como foram desenvolvidos; (b) Processos descritos superficialmente, sem descreverem em detalhes suas fases.

**Termos de busca:** *game AND (“game-based learning” OR education OR educational OR learning OR teaching OR training OR “serious game” OR instructional) AND (method OR methodology OR*

---

<sup>4</sup>[www. http://scholar.google.com.br/](http://scholar.google.com.br/)

*approach OR framework OR process) AND (design OR implementation OR develop OR creation).*

### **Execução da busca**

Várias buscas foram executadas em Agosto de 2013 separadamente, utilizando tanto a busca com todos os termos, quanto às buscas mais amplas usando somente partes do termo de busca. A busca foi realizada em três fases. A primeira fase busca a seleção dos processos por meio da leitura dos títulos e abstracts. A segunda fase busca identificar entre os artigos selecionados que realmente apresentam processos de desenvolvimento de jogos educacionais, para isso artigos considerados relevantes no primeiro passo foram lidos completamente. A terceira fase busca identificar entre os processos selecionados na fase 2, quais processos apresentam descrições claras de suas fases e não apresentem apenas características de jogos educacionais.

Na primeira fase, foram encontrados 47 artigos por meio da leitura dos títulos e *abstracts* apresentados como resultado da busca na ferramenta *Google Acadêmico*. Entre os 47 artigos selecionados foram considerados relevantes 12 artigos após uma leitura completa dos artigos. E entre os 12 artigos, apenas 5 descrevem claramente as fases do processo. Durante a busca foram encontrados muitos artigos enquadrados nos termos, contudo, não havia relação com a pesquisa, por exemplo, simulações na área de inteligência artificial e jogos digitais de modo geral. Foram analisadas apenas as 10 primeiras páginas dos resultados do *Google Acadêmico*, pois a partir da décima página os artigos deixaram de possuir relação com o tema de pesquisa. Ao processo de busca foi adicionado um artigo relevante (STAALDUINEN e FREITAS, 2011), encontrado em buscas *ad-hoc* que foram feitas em paralelo. Com isso, foram identificados no total 6 processos de desenvolvimento de jogos educacionais dentro dos critérios de inclusão/exclusão definidos.

### **Extração da informação**

Uma visão geral dos processos encontrados é apresentada de modo gráfica na tabela 11. As fases de cada um dos processos são detalhadas na tabela 12, permitindo com isso, a comparação entre eles. A partir dos artigos encontrados foram extraídos os dados e interpretados. Em seguida os dados foram utilizados no mapeamento sistemático das fases e atividades dos processos.

O processo proposto por Staalduinen e Freitas (2011) pode ser utilizado para desenvolver um jogo novo ou para traçar desempenhos de aprendizagem com jogos existentes. Como ponto fraco este processo não deixa clara as fases de desenvolvimento do jogo, sendo focado exclusivamente na área didática. Sommeregger e Kellner (2012) apresentam um processo com fases de projeto conceitual e design do jogo bem definidas focadas no desenvolvimento do jogo, como, definir público-alvo, definir objetivos de aprendizagem, definir métodos de aprendizagem baseados nas teorias construtivista, behaviorismo, cognitivismo. Na fase de implementação orientam a utilização de ambientes de desenvolvimento (*game engine*), como Open Sludge, Wintermute Engine e e-Adventure, não orientando o desenvolvedor que deseja produzir o próprio jogo independentemente de ambiente de desenvolvimento de jogos.

Marcos e Zagalo (2011) apresentam um processo de desenvolvimento de jogos educacionais fundamentados no processo de artes digitais. Neste processo o desenvolvimento do jogo em si é enfatizado, contudo não aborda aspectos didáticos como Sommeregger e Kellner (2012). Como ponto forte do processo proposto por Marcos e Zagalo (2011), é possível destacar a necessidade de auxílio na seleção das tecnologias empregadas para desenvolver o jogo, fase para desenhar o esboço do jogo como recomendado por Fullerton (2008). Além disso, apresenta uma fase de grande relevância para o processo que é específica para produção das histórias do jogo, sendo de modo geral um processo interativo. Kirkley, Tomblin e Kirkley (2005) apresentam um processo de desenvolvimento de jogos educacionais que mescla o processo de design instrucional, utilizando ISD (DICK e CAREY, 1996), e o processo de desenvolvimento de jogos, sendo chamado de *Simulation-Games Instructional System Design Model* (SG-ISD). Entretanto, o artigo não descreve as fases do processo e foca na apresentação da ferramenta de autoria implementada para desenvolver jogos. Esta ferramenta foca apenas no processo de concepção e modificação do jogo. Loh (2009) apresenta um processo que busca integrar aos jogos educacionais à aprendizagem e à avaliação. Aparentemente o processo é bem estruturado, enfatizando aspectos como público-alvo, mecanismos do jogo, eficácia de avaliação. Mas apresenta pontos fracos, como ter as fases iniciais de definição do público-alvo, financiamento e conteúdo no mesmo nível hierárquico da definição das narrativas. Por exemplo, para Marcos e Zagalo (2011) a fase da narrativa é realizada somente depois da fase de concepção. E no

caso de Sommeregger e Kellner (2012) existe distinção entre as fases de concepção e de design do jogo.

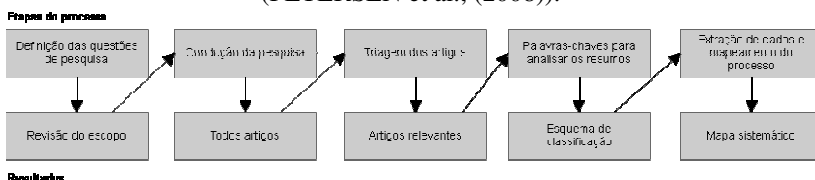
No artigo apresentado por Kickmeier-Rust et al. (2006) são enfatizadas características importantes que justificam a utilização de jogos para ensino, como ambiente imersivo, histórias envolventes, equilíbrio inteligente entre desafios e habilidades. Entretanto, os jogos educacionais nem sempre apresentam a mesma qualidade de jogos comerciais. Neste contexto, o processo proposto por Kickmeier-Rust et al. (2006) apresenta uma abordagem interdisciplinar por meio da ciência cognitiva, neurociência, pedagogia, design de jogos, objetivando equilibrar as qualidades dos jogos comerciais nos jogos educacionais.

Kickmeier-Rust et al. (2006) também sugere a produção de uma ontologia do domínio de conhecimento, podendo ser um ponto fraco deste processo, exige um especialista em ontologias. Contudo, o processo é focado em aspectos pedagógicos e concepção do jogo, não enfatizando a implementação do jogo, também não apresentando as fases de teste e avaliação como Sommeregger e Kellner (2012) e Loh (2009).

## Mapeamento dos processos

Dando continuidade à revisão sistemática da literatura dos processos para desenvolvimento de jogos educacionais, apresenta-se o mapeamento dos processos identificados. O mapeamento é fundamentado nos processos de mapeamento sistemático para Engenharia de Software principalmente segundo Petersen et al. (2008) e as pesquisas de Budgen et al. (2008), Silveira Neto (2011), Afzal, Torkar e Feldt (2009) complementaram a pesquisa. Considerando de modo geral, as seguintes fases de mapeamento de processos: definição das questões de pesquisas; condução da busca; palavras-chaves usadas no abstract; extração de dados e processo de mapeamento (figura 17).

Figura 17 – Processo para o mapeamento sistemático (PETERSEN et al., (2008)).



Fonte: PETERSEN et al. 2008.

Neste mapeamento foram agrupadas as fases semelhantes dos processos com as 5 fases do design instrucional ADDIE (MOLEND, 2003, BRANCH, 2009), como apresentado na tabela 12. Foram identificadas fases semelhantes entre os processos e também fases complementares. Por exemplo, Marcos e Zagalo (2011) e Loh (2009) apresentam fases de narrativa, ambas foram enquadradas na fase de projeto do design instrucional ADDIE (MOLEND, 2003, BRANCH, 2009). Com base na tabela 12 observa-se que alguns processos não apresentam atividades, mas somente apresentam fases genéricas que não detalha-se o desenvolvimento do jogo.

Além de muitos processos se complementarem entre si, algumas fases apresentavam similaridades entre os processos, como a definição dos objetivos instrucionais, prototipação e preocupação com o tipo de jogador.

Destaca-se como a maior diferença entre os processos o fato de que algumas fases são descritas com maior detalhe do que as fases similares de outro processo. Como o caso de Staalduinen e Freitas (2011) que propõem a fase de definição dos objetivos, e Kickmeier-Rust et al., (2006) contém uma abordagem mais ampla onde busca definir o currículo ou conteúdo a ser trabalhando, os principais tópicos para aprendizagem e por último os objetivos de aprendizagem.

**(a) Quais são os processos existentes para o desenvolvimento jogos educacionais?** Foram identificados poucos processos existentes na literatura, contrastando com a grande quantidade de jogos existentes. Entre os processos existentes, apenas 6 processos descrevem claramente suas fases e permitem maior entendimento sobre seu funcionamento. Alguns artigos encontrados apresentaram apenas características educacionais ou da teoria de jogos, não sendo considerado um processo. Outros artigos estavam relacionados a área de inteligência artificial, simulação ou jogos de educação física. Dos processos identificados (Tabela 11) todos apresentaram aspectos de design instrucional e design de jogos, porém ainda são fortemente voltados para o design instrucional, exceto o processo apresentado por Marcos e Zagalo (2011).

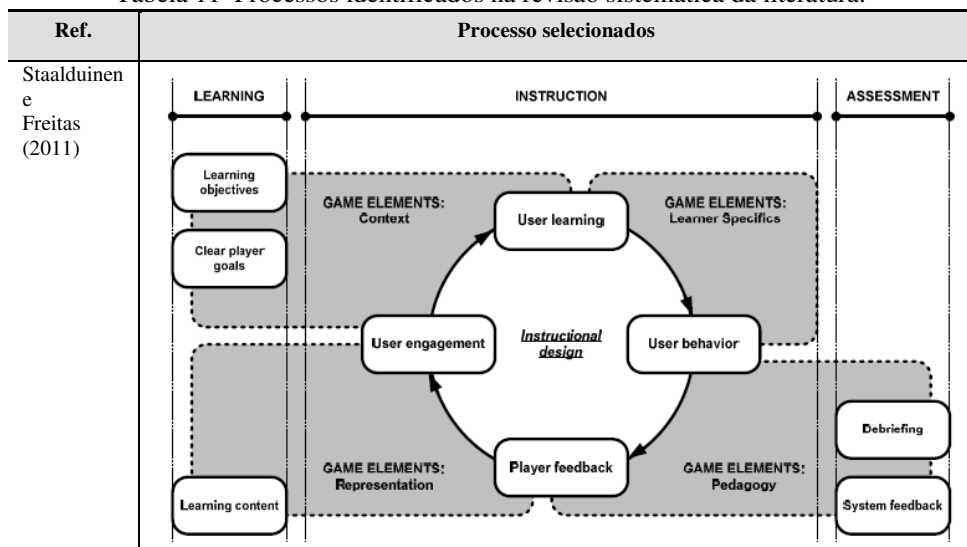
**(b) Quais são as etapas/fases/atividades desses processos?** Apresenta-se na tabela 11 as fases dos 6 processos considerados relevantes pelos autores, na qual é possível verificar que as fases são fortemente orientadas pelo design instrucional, por outro lado as fases apresentadas por Marcos e Zagalo (2011) são exclusivamente focadas



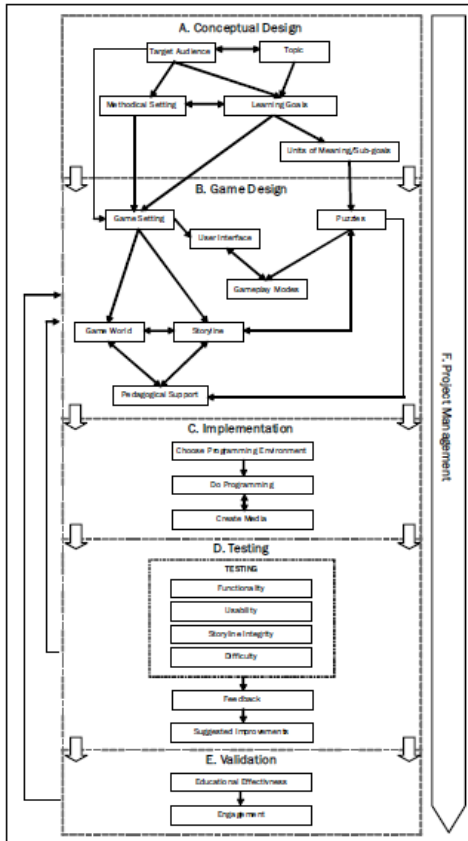
em design de jogos digitais. De modo geral, as fases se concentram principalmente na análise e no projeto dos jogos. As fases específicas para o desenvolvimento dos jogos não estão amplamente detalhadas.

(c) **Quais similaridades ou diferenças existem entre os processos existentes?** A tabela 12 apresenta as diferenças e semelhanças entre os processos encontrados na literatura. Como principal semelhança é possível destacar que grande parte dos processos concentram-se suas atividades no projeto e desenvolvimento de jogos. Mas fica evidente que poucos processos apresentam fases segundo Design Instrucional de referência, como o ADDIE ou ISD. Por exemplo, existem processos que não realizam análise ou avaliação do jogo, fases que são extremamente relevantes segundo o design instrucional.

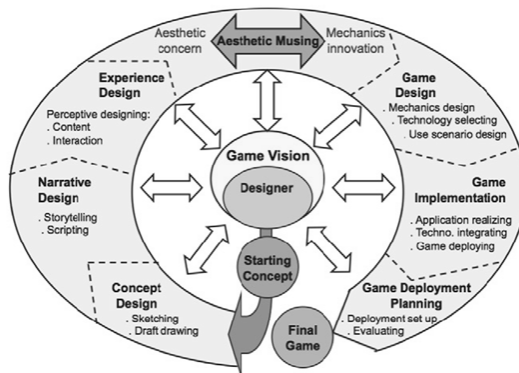
Tabela 11–Processos identificados na revisão sistemática da literatura.



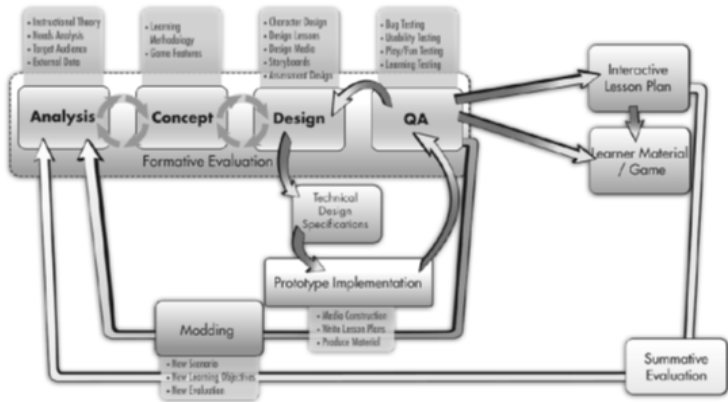
Sommeregger e Kellner (2012)



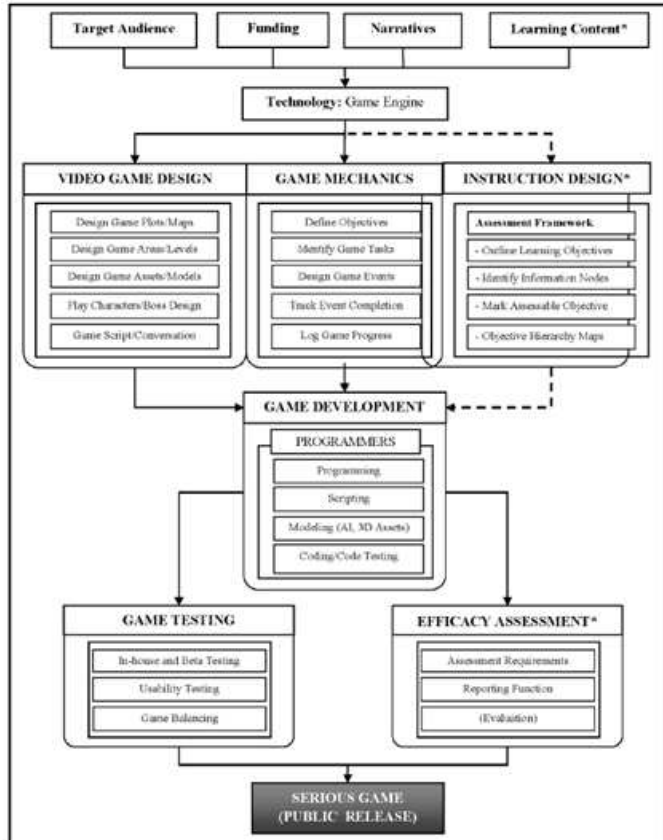
Marcos e Zagalo (2011)



Kirkley,  
Tomblin e  
Kirkley  
(2005)



Loh (2009)



Kickmeier-Rust et al. (2006)

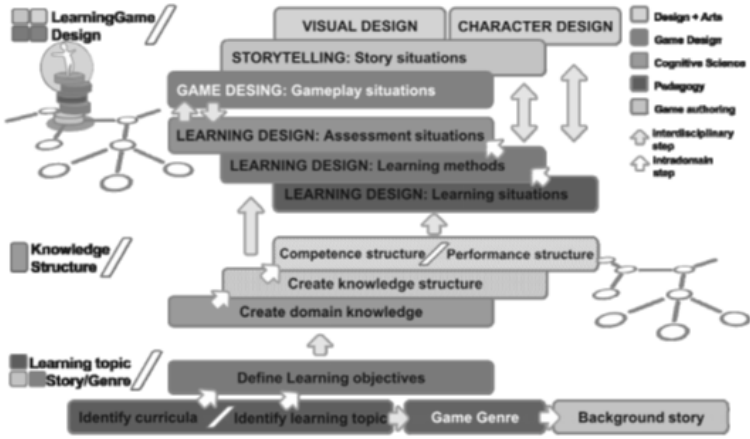


Tabela 12–Fases dos processos encontrados.

Artigos	Análise	Projeto	Desenvolvimento	Execução	Avaliação
Staalduinen e Freitas (2011)	1 Aprendizagem 1.1 Objetivos da aprendizagem 1.2 Objetivo dos jogadores 2 Conteúdo	3 Instrução 3.1 Aprendizagem do usuário 3.2 Comportamento do usuário 3.3 Feedback ao usuário 3.4 Envolvimento do usuário			4 Avaliação 4.1 Questionário 4.2 Feedback dos usuários sobre o sistema
Sommeregger e Kellner (2012)	1 Design conceitual	2 Design do jogo	3 Implementação	4 Testes	5 Avaliação
Marcos e Zagalo (2011)	1 Ideia inicial	2 Design 2.1 Conceito 2.2 Narrativas 2.3 Experiência	3 Jogo 3.1 Design do jogo 3.2 Implementação	4 Planejamento da implantação	
Kirkley, Tomblin e Kirkley (2005)	1 Análise	2 Concepção 3 Design 4 Garantia de qualidade 5 Plano de aula interativo 6 Material da aula/jogo 7 Especificação técnica de design	8 Implementação do protótipo 9 Modelagem		10. Avaliação
Loh (2009)	1 Público-alvo 2 Diversão 3 Conteúdo de aprendizagem	4 Narrativas 5 Tecnologia: máquina do jogo 5.1 Vídeo game 5.2 Mecanismo do jogo 5.3 Design instrucional	6 Desenvolvimento do jogo	7 Testes	8 Avaliação da eficácia 9 Publicação do jogo
Kickmeier-Rust et al. (2006)	1 Identificar currículo 1.1 Identificar tópicos de aprendizagem	3 Gênero do jogo 4 Estória 5 Criar domínio de conhecimento	9 Design do visual 10 Características do design		

	2 Definir objetivos de aprendizagem	6 Criar estrutura de conhecimento 7 Competência e performance 8 Aprendizagem e design do jogo 8.1 Situação de aprendizagem 8.2 Métodos de aprendizagem 8.3 Situação de avaliação 8.4 Situações do jogo 8.5 Estórias			
--	-------------------------------------	--	--	--	--

## Discussão

A revisão sistemática da literatura dos processos de desenvolvimento de jogos educacionais encontrou inicialmente 47 pesquisas que tratavam do desenvolvimento de jogos educacionais. Contudo, apenas 6 pesquisas foram selecionadas entre as 47, porque descreviam com maiores detalhes as fases e atividades dos processos. Grande parte das pesquisas encontradas apresentaram apenas características necessárias para os jogos educacionais.

A seguir são apresentadas as principais lacunas identificadas nos 6 processos selecionados na revisão sistemática da literatura:

1. Entre os processos selecionados, apenas Marcos e Zagalo (2011) não apresentam fases e atividades ligadas aos aspectos instrucionais.

2. Existem processos que focam suas atividades em uma fase do design instrucional e apresentam lacunas em relação a outras fases (KIRKLEY, TOMBLIN e KIRKLEY, 2005).

3. Existem processos que apresentam apenas fases do design de jogos e design instrucional, mas apresentam lacunas em relação ao detalhamento das atividades (SOMMEREGGER e KELLNER, 2012).

4. Existem processos que não são completos, porque não apresentam fases e atividades do design instrucional e design de jogos (STAALDUINEN e FREITAS, 2011, MARCOS e ZAGALO, 2011, KIRKLEY, TOMBLIN e KIRKLEY, 2005, KICKMEIER-RUST, et al. 2006).

5. Existem processos que não apresentam integração das fases e atividades. Como no processo de Loh (2009) que apresenta atividades da fase de projeto do jogo, na fase de análise da unidade instrucional.

6. Em geral, os processos apresentam lacunas em relação a avaliação da unidade instrucional. Apenas Staalduinen e Freitas (2011) apresentam maior suporte a avaliação da unidade instrucional, mas mesmo neste processo não são apresentados modelos específicos para avaliação de jogos educacionais.

No próximo capítulo apresenta-se o processo ENgAGED que busca preencher as lacunas dos processos encontrados na literatura com detalhamento das fases e atividades do design instrucional e design de jogos.





## 4 PROCESSO ENGAGED

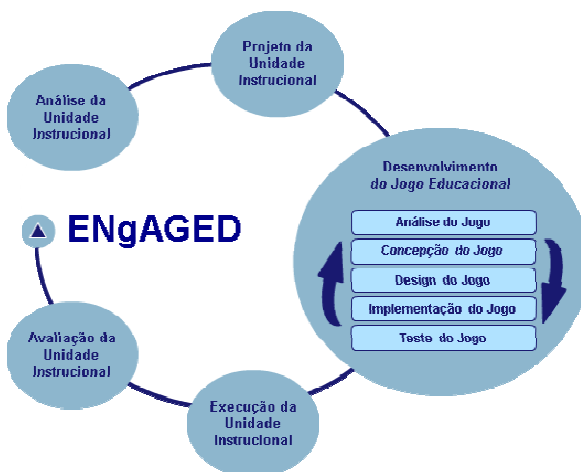
Neste capítulo apresenta-se o processo de desenvolvimento de jogos educacionais ENgAGED e também um comparativo entre os processos encontrados na revisão da literatura e o processo ENgAGED.

### 4.1 DESCRIÇÃO DO PROCESSO ENGAGED

O ENgAGED (*EducatioNAl GamEs Development*) é um processo de desenvolvimento, modelado com o objetivo de produzir jogos educacionais para aplicação em cursos de Computação nas instituições de ensino superior (IES). Neste contexto, a presente pesquisa utiliza como definição para o conceito de processo: “conjunto de fases, ações e atividades interrelacionadas, que são executadas para alcançar produtos, resultados ou serviços predefinidos” (SCHWABERE e BEEDLE, 2001). Assim, um processo é caracterizado por suas entradas, ferramentas, técnicas e as saídas resultantes (SCHWABERE e BEEDLE, 2001).

O processo ENgAGED (figura 18) é modelado com 10 fases, sendo 5 relativas ao design instrucional e 5 ao design de jogos. Também é formado por 19 atividades que interrelacionam ou integram aspectos do design instrucional e design de jogos.

Figura18 – ENgAGED - *Educational Games Development*.



Fonte: Elaborada pelo autor.

O ENgAGED é direcionado a professores ou alunos da Computação e oferece suporte para o desenvolvimento de jogos educacionais, mesmo não havendo conhecimento de design instrucional ou design de jogos. Todas as atividades do processo apresentam exemplos de como executá-las, tendo em alguns casos listagens (nomes, descrições e características) de programas ou ferramentas que podem ser utilizadas para desenvolver os jogos. Em um cenário ideal, seria interessante o professor ter disponível uma equipe multidisciplinar de pedagogos, especialistas do domínio, design instrucional, design gráfico, design de jogos. Porém este cenário não ocorre tipicamente em IES brasileiras. Mas um apoio que o professor normalmente pode ter, são alunos de graduação ou pós-graduação sendo orientandos e oferecendo suporte para o desenvolvimento de jogos. Com isso, define-se o contexto do ENgAGED para instituições de ensino superior, as quais não apresentam muitos recursos financeiros disponíveis para contratação de equipes multidisciplinares ou até mesmo para compra de softwares proprietários.

O processo foi modelado de modo misto, utilizando a modelagem prescritiva e descritiva (ACUÑA, 2000), (ROSSI, 1999) e (THIRY et al., 2006). A modelagem descritiva define como o processo é executado por meio de observação e determina as melhorias dele. Esta modelagem pode apresentar três fases principais: (a) elicitación do conhecimento do processo, (b) a formalização do conhecimento, (c) revisão do modelo proposto (BECKER-KORNSTAEDT, 2001). A modelagem prescritiva descreve como o processo deve ser executado, com isso estabelecendo regras, diretrizes e padrões (ACUÑA e FERRÉ, 2001).

Considerando a modelagem descritiva e prescritiva, o ENgAGED é formado pelas fases dos processos identificados na revisão sistemática da literatura, pelos processos de design instrucional ADDIE (MOLENDÁ, 2003, BRANCH, 2009) e SAM (DICK e CAREY, 1996), e pelo processo de design de jogos proposto por Fullerton(2008) e Schell (2008). Contudo, na modelagem também foram considerando outros processos e pesquisas de design de jogos e instrucional (THOMPSON, BARNABY-GREEN e CUS-WORTH, 2007, DESPAIN, 2012, FILATRO, 2008, CLARK, 2009).

O processo é descrito por meio de fases, atividades, tarefas e produtos de trabalho, os quais são descritos conceitualmente, abaixo (ACUÑA, 2000, BENALI e DERNIAME, 1992, FINKELSTEIN, KRAMER e NUSEIBEH, 1994, WANGENHEIM et al., 2010):

- **Fase:** Fase é um conjunto de atividades agrupadas em etapas, apresentando uma sequência lógica ou estruturada.

- **Atividade:** Atividade é um estágio do processo que produz mudanças visíveis no estado do produto. A atividade pode ter entradas, saídas, resultados intermediários, genericamente chamados de produtos. A atividade implementa procedimentos, regras, políticas e objetivos para transformar um produto.

- **Tarefa:** Tarefa consiste na execução de uma determinada atividade do processo. A tarefa apresenta uma granularidade mais fina que a atividade. Uma atividade do processo pode ter uma ou mais tarefas associadas a ela. De modo geral, a tarefa descreve os passos para executar dada atividade.

- **Produtos de trabalho:** Produtos de trabalho são as entradas e saídas de uma atividade de um processo. Eles podem ser produzidos e consumidos ao longo do processo podendo ter longos ciclos de vida, sendo criados, acessados e modificados.

A tabela 13 apresenta um exemplo utilizado para o detalhamento das atividades do processo ENgAGED.

Tabela 13 – Exemplo de estrutura para detalhamento das atividades do processo ENgAGED.

<b>Atividade</b>	<b><i>AI.2 Caracterizar aprendizes</i></b>
<b>Descrição</b>	<i>A análise dos aprendizes tem por objetivo identificar o público-alvo, ou seja, a qual público o conteúdo será ensinado, sendo necessário identificar o perfil e as características.</i>
<b>Tarefas</b>	<b><i>1.2.1 Caracterizar aprendizes</i></b> <i>Tarefa de caracterização do perfil de aprendizes, relacionando as habilidades, conhecimentos e identificando as preferências (hobbies), tipo de equipamento que utiliza normalmente para ao final definir personas, ou seja, cria-se personagens fictícios com os perfis característicos dos aprendizes.</i>
<b>Produtos de trabalho</b>	O produto de trabalho é dividido em produtos de entrada e saída. Os produtos de entrada são os parâmetros que permite o processamento do processo, sendo apresentado em forma de produto de saída.  Na atividade <i>AI.2 Caracterizar aprendizes</i> o produto de entrada é <b><i>PI.2.1 - Questionário de caracterização dos aprendizes e contexto</i></b> . E o produto de saída é Produto de saída: <b><i>PI.2.2 - Caracterização dos aprendizes</i></b> .

No início, a modelagem do processo ENgAGED não apresentou detalhamento das fases e atividades. No desenvolvimento do jogo *SCRUM-Scape* (CAMARGO, 2013) o processo apresentou descrições em alto nível das fases e atividades. No desenvolvimento do jogo educacional *SCRUM'ed* (SCHNEIDER, 2015) houve um maior detalhamento das fases e das atividades, sendo aprimorado a medida que

o jogo era desenvolvido. Em paralelo ao desenvolvimento do jogo *SCRUM'ed*, o jogo *PM Quiz* (CASSETTARI, 2015) também foi desenvolvido. O desenvolvimento do jogo *Fuga de Gambix* (JUNCKES, 2016) contribuiu para adequar o processo para o desenvolvimento de jogos não-digitais.

Na tabela 14 apresenta-se uma visão geral do processo ENgAGED, porém, no apêndice E são apresentados os *templates* do processo. O processo também pode ser acessível por meio do relatório técnico em “*ENgAGED: Processo de Desenvolvimento de Jogos para Ensino em Computação*” (BATTISTELLA e WANGENHEIM, 2015).

Tabela 14–Visão geral do processo ENgAGED.

<b>Fase 1. Análise da Unidade Instrucional (UI)</b>	
<b>A1.1 Especificar UI do jogo</b>	Especificar o contexto da UI, como curso, disciplina, pré-requisitos da disciplina, objetivos da disciplina e conteúdo programático.
<b>A1.2 Caracterizar aprendiz</b>	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.
<b>A1.3 Definir objetivo(s) de desempenho</b>	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.
<b>Fase 2. Projeto da Unidade Instrucional (UI)</b>	
<b>A2.1 Definir avaliação do aluno</b>	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.
<b>A2.2 Definir conteúdo da estratégia instrucional</b>	A estratégia instrucional é <i>Jogo Educacional</i> . Assim, nesta atividade define-se o conteúdo e o sequenciamento do conteúdo ao longo do jogo.
<b>A2.3 Decidir pelo desenvolvimento ou utilizar jogo desenvolvido</b>	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> .
<b>A2.4 Revisar o modelo de avaliação do jogo</b>	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).
<b>Fase 3. Desenvolvimento do Jogo Educacional</b>	
<b>Fase 3.1. Análise do Jogo</b>	
<b>A3.1.1 Levantar requisitos do jogo</b>	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.
<b>Fase 3.2. Concepção do Jogo</b>	

<b>A3.2.1 Conceber o jogo</b>	Conceber o jogo, descrevendo as principais características, como objetivos do jogo, narrativa, regras, mecânica, elementos do jogo, pontuações e <i>feedback</i> educacional.	
<b>Fase 3.3. Design do Jogo</b>		
	<b>Jogo digital</b>	<b>Jogo não-digital</b>
<b>A3.3.1 Definir linguagem de programação ou <i>game engine</i></b>	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.	Não se aplica.
<b>A3.3.2 Produzir ilustrações ou imagens dos elementos do jogo</b>	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.	Produzir as imagens que representarão os elementos do jogo não-digital, como tabuleiros, cartas, fichas, dados, peças de montar (lego), etc.
<b>A3.3.3 Modelar o jogo</b>	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.	Modelar as etapas realizadas no jogo não-digital apresentando um fluxograma com a sequência de passadas das jogadas realizadas pelo jogador, do início ao fim do jogo. Também é necessário modelar o <i>feedback</i> educacional do jogo.
<b>Fase 3.4. Implementação do Jogo</b>		
<b>A3.4.1 Produzir elementos do jogo</b>	Produzir os elementos criados na atividade A3.3.2 <i>Produzir ilustrações ou imagens dos elementos do jogo</i> .	
	<b>Jogo digital</b> Codificação ou programação dos cenários, personagens, artefatos e objetos do jogo por meio de linguagem de programação ou <i>game engine</i> .	<b>Jogo não-digital</b> Produção dos elementos não-digitais.
<b>Fase 3.5. Testes do Jogo</b>		
<b>A3.5.1 Realizar testes do jogo</b>	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).	
<b>Fase 4. Execução da Unidade Instrucional (UI)</b>		
<b>A4.1 Planejar a execução do jogo</b>	Planejar a execução do jogo definindo data para jogar, local aonde será jogado, e equipamentos que serão utilizados ou materiais que devem ser impressos.	
<b>A4.2 Instalar o jogo digital</b>	<b>Jogos digitais</b>	<b>Jogos não-digitais</b>
	Instalar o jogo conforme a sua	Não se aplica.

	plataforma. Caso seja um jogo com arquitetura cliente-servidor, é necessário instá-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> .	
<b>A4.3 Executar o jogo</b>	o	Executar o jogo em sala de aula, laboratório ou extraclasse.
<b>Fase 5. Avaliação da Unidade Instrucional (UI)</b>		
<b>A5.1 Conduzir avaliação</b>		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> .
<b>A5.2 Analisar dados da avaliação</b>		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> .

A seguir são apresentadas as fases do processo ENgAGED contendo seus objetivos e atividades.

## F1 – ANÁLISE DA UNIDADE INSTRUCIONAL

Esta fase tem por objetivo definir, o escopo da unidade instrucional e para isso é necessário a definição dos objetivos a serem alcançados e o público-alvo. A fase de análise é fundamentada principalmente no design instrucional ADDIE (MOLEND, 2003, BRANCH, 2009) e ISD (DICK e CAREY, 1996), neste sentido são priorizadas as atividades relacionadas didática do conteúdo. Como resultado desta fase espera-se que o Professor defina claramente o contexto e as metas instrucionais na qual o jogo será projetado, implementando, aplicado e avaliado. Nesta fase, é necessário realizar uma pesquisa para identificar os perfis dos alunos de Computação. Neste contexto, foi realizado uma primeira pesquisa, com 406 alunos e ex-alunos de Computação, tendo os resultados apresentados no artigo “*Caracterização do Público-Alvo de Jogos Educacionais na área da Computação*” (BATTISTELLA e WANGENHEIM, 2016b).

## F2 – PROJETO DA UNIDADE INSTRUCIONAL

Esta fase tem por objetivo a definição das estratégias instrucionais para alcançar os objetivos propostos na fase anterior, e também desenvolver os instrumentos avaliativos. Nesta fase o professor definirá, por exemplo, se utilizará jogos existentes ou desenvolverá um novo jogo, ou ambos. Como resultado desta fase, espera-se que o professor tenha definido as estratégias instrucionais que serão utilizadas ao longo da unidade instrucional.

## F3 - DESENVOLVIMENTO DO JOGO EDUCACIONAL

Esta fase tem por objetivo o desenvolvimento do jogo educacional contendo fases e atividades para orientar o Professor na implementação do jogo. As fases do processo de desenvolvimento do jogo são formadas principalmente por fases identificadas na revisão sistemática da literatura e do design de jogos proposto por Fullerton (2008). Entretanto também apresenta fases relativas ao design instrucional. É importante ressaltar que o processo de desenvolvimento do jogo não é sequencial, mas iterativa. A seguir são apresentadas as fases correspondentes à fase de desenvolvimento do jogo educacional.

### F3.1 – Análise do jogo

Esta fase tem por objetivo realizar o levantamento dos requisitos do jogo. Os requisitos são funções ou funcionalidades indispensáveis, necessárias e requeridas no jogo. Normalmente este levantamento é realizado por meio de *brainstorming*, questionário, entrevista, observação ou casos de uso.

### F3.2 – Concepção do jogo

Esta fase tem por objetivo conceber o jogo, sendo necessário definir o objetivo do jogo, gênero, plataforma, modo de interação dos jogadores, história do jogo, personagens, artefatos, cenários, diálogos, menus de interação com jogador, regras do jogo, critérios para vencer o jogo e a forma de *feedback* educacional para o jogador.

### **F3.3 – Design do jogo**

Esta fase tem por objetivo definir as linguagens de programação, *game engine* e componentes (bibliotecas) que serão utilizadas pelo jogo. Também será necessário nesta fase escolher ou desenhar as imagens dos elementos do jogo (protótipos). Os elementos do jogo são *personagens*, *artefatos*, *cenários*, *diálogos* e *menus*. Ao final desta atividade é necessário modelar o jogo, criando a arquitetura do jogo, diagrama de estados descrevendo a evolução do personagem ou a sequência de cenários do jogo. E caso necessário, é possível criar o diagrama de classe e entidade-relacionamento.

Esta fase também busca uma documentação detalhada do processo de desenvolvimento, para permitir, por exemplo, que um jogo desenvolvido por um aluno de graduação seja continuado por outro aluno. Deste modo, é possível dar continuidade ao processo de desenvolvimento, aperfeiçoando o jogo desenvolvido a cada iteração do processo.

### **F3.4 – Implementação do jogo**

Esta fase tem por objetivo codificar os elementos do jogo produzidos na fase de design. Vale destacar que o processo não é rígido, de modo que é possível implementar os elementos de jogo, mesmo que toda a fase de design ainda não esteja finalizada. Com objetivo de documentação do processo de desenvolvimento, nesta fase todos os elementos do jogo serão salvos em forma de *screenshots* caso seja um jogo digital, ou com fotos digitais caso seja um jogo não-digital.

### **F3.5 – Teste**

Esta fase tem por objetivo realizar os testes do jogo desenvolvido, entretanto, considerando que é um processo iterativo, não é necessário o desenvolvimento completo do jogo para realizar os testes, por exemplo, é possível desenvolver os primeiros níveis do jogo e realizar os testes. O objetivo destes testes é a identificação de erros do jogo. Os testes são realizados inicialmente pelo criador de jogos. Em seguida deve-se selecionar uma amostra do público-alvo, tipicamente ex-alunos da disciplina, para testar o jogo.



#### F4– EXECUÇÃO DA UNIDADE INSTRUCIONAL

Esta fase tem por objetivo planejar e executar a unidade instrucional, ou seja, a partir do momento em que o jogo educacional está desenvolvido, ele será aplicado pelo Professor em sala de aula. Sendo necessário planejar a execução da unidade instrucional, considerando os computadores disponíveis no laboratório, quantidade de alunos, tempo disponível, se todos os alunos terão acesso ao jogo. Como resultado desta fase, espera-se que o jogo seja aplicado como estratégia instrucional. Eventualmente, um Professor pode optar pela utilização de um jogo já desenvolvido, então passaria da fase *F2 Projeto* para a fase *F4 Execução*.

#### F5– AVALIAÇÃO DA UNIDADE INSTRUCIONAL

Esta fase tem por objetivo definir, planejar e executar a avaliação da unidade instrucional. Por meio desta fase o professor analisará os resultados da unidade instrucional e identificará se os objetivos definidos na fase de análise foram alcançados. Portanto, esta fase apresenta características expressamente do design instrucional.

Como modelo de referência para esta fase, recomenda-se a utilização do modelo de avaliação de jogos educacionais MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011).

#### 4.2 COMPARATIVO ENTRE ENGAGED E TRABALHOS RELACIONADOS

No capítulo de Fundamentação Teórica foram apresentados aspectos fundamentais do design instrucional, como a necessidade de realizar a caracterização do público-alvo, definição dos objetivos e avaliação. Neste capítulo também foram abordadas as características básicas necessárias em um jogo educacional, levando em consideração aspectos do design de jogos, como gênero, plataforma, regras, narrativas e mecânicas do jogo.

Considerando os aspectos fundamentais do design instrucional e do design de jogos os processos encontrados na revisão sistemática e apresentados no capítulo de Estado da Arte foram comparados com o processo ENgAGED. Na tabela 15 são apresentadas as comparações entre os processos encontrados e o ENgAGED.

Tabela 15 – Comparativo entre ENgAGED e processos encontrados na literatura.

<b>Design Instrucional</b> ADDIE (BRANCH, 2009) e ISD (DICK e CAREY, 1996)		<b>ENgAGED</b>	<b>Staalduinen e Freitas (2011)</b>	<b>Sommeregger e Kellner (2012)</b>	<b>Marcos e Zagalo (2011)</b>	<b>Kirkley, Tomblin e Kirkley (2005)</b>	<b>Loh (2009)</b>	<b>Kickmeier- Rust et al. (2006)</b>
<b>Fases</b>	<b>Atividades</b>							
<b>Analisar</b>	Identificar Metas Instrucionais	A	N	A	N	A	PA	A
	Analisar Aprendizizes e Contexto	A	N	A	N	A	A	N
	Conduzir Análise Instrucional	A	A	N	N	A	N	A
	Escrever Objetivos de Desempenho	A	A	A	N	N	A	A
<b>Projetar</b>	Revisar Instrução	A	N	N	N	N	N	N
	Desenvolver Instrumento de Avaliação	A	PA	N	N	N	N	A
	Desenvolver Estratégias Instrucionais	A	A	A	N	A	A	A
<b>Desenvolver</b>	Desenvolver e Selecionar Materiais Instrucionais	A	N	A	N	N	N	N
<b>Executar</b>	Planejar necessidade dos Aprendizizes e Ambiente	A	N	N	N	N	N	N
<b>Avaliar</b>	Conduzir Avaliação	A	A	AP	N	A	A	N
	Analisar Dados Coletados	A	A	N	N	PA	PA	N

<b>Design de Jogos</b> (FULLERTON, 2008, KREMERS, 2009, SALEN e ZIMMERMAN, 2009, OMERINCK, 2004, SCHELL, 2008, OLIVERIRA, 2012)		<b>ENgAGED</b>	<b>Staalduinen e Freitas (2011)</b>	<b>Sommeregger e Kellner (2012)</b>	<b>Marcos e Zagalo (2011)</b>	<b>Kirkley, Tomblin e Kirkley (2005)</b>	<b>Loh (2009)</b>	<b>Kickmeier-Rust et al. (2006)</b>
<b>Analisar</b>	Levantar requisitos	A	N	N	N	N	N	N
<b>Conceber</b>	Definir Objetivos	A	A	A	N	N	A	A
	Definir Gênero	A	N	N	N	N	N	A
	Definir Plataforma	A	N	A	N	N	N	N
	Definir Modo de Interação	A	N	A	A	N	N	N
	Definir Regras	A	N	A	N	N	N	N
	Definir Mecânica	A	A	N	A	N	N	A
	Definir Narrativas	A	N	A	A	A	A	A
	Definir Elementos (personagens, cenários, objetos, etc.)	A	A	A	A	A	A	N
	Definir os Critérios para Vencer	A	N	N	N	N	N	N
	Definir a Pontuação	A	N	N	N	N	N	N
	Feedback	A	A	N	N	N	N	N
	Integração Design Instrucional e Design de Jogos	A	PA	PA	N	PA	PA	PA

<b>Design de Jogos</b> (FULLERTON, 2008, KREMERS, 2009, SALEN e ZIMMERMAN, 2009, OMERINCK, 2004, SCHELL, 2008, OLIVERIRA, 2012)		<b>ENgAGED</b>	<b>Staalduinen e Freitas (2011)</b>	<b>Sommeregger e Kellner (2012)</b>	<b>Marcos e Zagalo (2011)</b>	<b>Kirkley, Tomblin e Kirkley (2005)</b>	<b>Loh (2009)</b>	<b>Kickmeier-Rust et al. (2006)</b>
<b>Projetar</b>	Definir Linguagens de Programação ou Game Engine	A	N	N	A	A	A	A
	Produzir Ilustrações dos Elementos (personagens, cenários, objetos, etc.)	A	A	N	A	A	A	N
	Definir Acessibilidade	A	N	N	N	N	N	N
	Definir Fluxograma do Jogo	A	N	N	N	N	N	N
	Descrever Diálogo dos personagens	A	N	N	N	N	N	N
<b>Implementar</b>	Implementação dos Elementos do Jogo (personagens, cenários, objetos, etc.)	A	PA	A	A	A	A	A
<b>Testar</b>	Realizar Testes	A	N	A	N	A	A	A
<b>Executar</b>	Instalar Jogo Digital	A	N	N	A	N	A	A

**Legenda:** (A) Atendido, (PA) Parcialmente Atendido, (N) Não atendido.

Com base no comparativo entre os processos encontrados na literatura é possível identificar lacunas em relação ao processo ENgAGED. Em relação ao design instrucional, observa-se lacunas em diversas fases dos processos atuais, como na de análise da unidade instrucional. Por exemplo, os processos de Staalduinen e Freitas (2011), Sommeregger e Kellner (2012) e Loh (2009) que não realizam a identificação das metas instrucionais, permitindo a produção de jogos que não estão alinhados ao conteúdo programático da disciplina ou então, não consideram os pré-requisitos da unidade instrucional.

Outra lacuna presente nos processos de Staalduinen e Freitas (2011) e Kickmeier-Rust et al. (2006) é não realizar a caracterização dos aprendizes. A caracterização dos aprendizes é uma atividade básica e fundamental ao design instrucional, porque permite a produção de conteúdos instrucionais alinhados ao público-alvo.

Outra lacuna está relacionada ao processo de Kirkley, Tomblin e Kirkley (2005), que apesar de realizar a identificação das metas instrucionais e analisar os aprendizes, não define os objetivos de desempenho dos alunos. A partir dos objetivos de desempenho é possível verificar após a avaliação da unidade instrucional, se os aprendizes realmente aprenderam os conteúdos previstos. Outra lacuna identificada nos processos atuais está relacionada a não realização da revisão da instrução. Esta atividade permite que ao longo do desenvolvimento da estratégia instrucional sejam realizadas revisões para identificar possíveis problemas no material que está sendo produzido.

Na fase da avaliação os processos atuais também apresentam lacunas. De modo geral, os processos orientam a realização da avaliação dos jogos utilizando um questionário que não é sistemático, por exemplo, indicando a utilização de um modelo de avaliação de jogos educacionais, como MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011).

Entre os processos, apenas o processo de Marcos e Zagalo (2011) não apresentam fases e atividades ligadas ao design instrucional, porque é proposto com base no desenvolvimento de artes digitais. Com isso, todos os aspectos necessários para garantir o mínimo de alinhamento aos aspectos pedagógicos tornam-se questionáveis.

Observa-se também lacunas nos processos em relação ao design de jogos. Estas lacunas são identificadas em atividades como no levantamento dos requisitos dos jogos, que é fundamental em processos de desenvolvimento de softwares, ou em atividades da fase de concepção que permitem a definição do gênero, plataforma, modo de

interação, regras, mecânica, critérios para vencer o jogo e *feedback* ao jogador. No processo apresentado por Staalduinen e Freitas (2011), Sommeregger e Kellner (2012), Loh (2009) e Kickmeier-Rust et al. (2006) existem lacunas relativas a prototipação do jogo. A prototipação permite a produção das ilustrações dos elementos do jogo (como, personagens, cenários, objetos), sendo comumente utilizado em processos de design de jogos. Esta atividade é fundamental para o design de jogos porque permite projetar as narrativas, cenários, personagens, regras e da mecânica antes da implementação do jogo.

Os processos atuais também não prevêm em atividades específicas de acessibilidade que permitem a inclusão de aprendizes com grau moderado de deficiência física, como daltonismo. Outra lacuna dos processos atuais está relacionada a definição de fluxogramas e diálogos do jogo. Por meio do fluxograma os níveis do jogo e a movimentação dos personagens no cenário podem ser projetados, permitindo a integração das regras e conteúdos de aprendizagem no jogo.

Observou-se também que todos os processos apresentam fases para implementação do jogo. Porém, não existe uma padronização entre as atividades desta fase, e também não são apresentados detalhes de como os jogos devem ser implementados.

Além de preencher as lacunas dos processos atuais, o processo ENgAGED foi modelado também com o objetivo de auxiliar professores de cursos da Computação ou alunos de graduação (orientados por professores) no desenvolvimento de jogos. Observa-se que as instituições de ensino superior brasileiras apresentam poucos recursos financeiros disponíveis para financiar uma equipe multidisciplinar de desenvolvimento de jogos. Deste modo, o processo busca orientar os criadores de jogos (professores ou alunos) no desenvolvimento de jogos, mesmo sem apresentarem conhecimento prévio de design instrucional ou design de jogos.

As atividades do processo apresentam descrições detalhadas de como devem ser realizadas, e algumas atividades que exigem ferramentas para desenvolvimento, como na produção das ilustrações dos elementos do jogo, são listas ferramentas proprietárias ou *open source/free* descrevendo suas funcionalidades e benefícios.

O processo também apresenta integração entre atividades do design instrucional com o design de jogos. Por exemplo, na fase de concepção do jogo, apresenta-se uma relação entre os níveis cognitivos de Bloom (1956) com os gêneros de jogos. Ou então, nas fases de

concepção e projeto existem atividades específicas para produção dos *feedbacks* educacionais e dos diálogos do jogo.

Outra lacuna identificada, em relação ao design de jogos, está relacionado ao levantamento de requisitos. Todos os processos encontrados não descrevem como realizar o levantamento de requisitos. No ENgAGED é necessário que defina-se os requisitos funcionais e não-funcionais do sistema, e define-se sua prioridade (essencial, importante, desejável). É possível que o levantamento dos requisitos realizados nos processos encontrados seja realizado em outras atividades, mas sem haver um detalhamento de como realizá-lo.





## 5 APLICAÇÃO DO PROCESSO ENGAGED

Neste capítulo apresenta-se um estudo de caso (YIN, 2012) que apresenta os jogos educacionais desenvolvidos a partir do processo ENgAGED.

### 5.1 DEFINIÇÃO DA APLICAÇÃO DO PROCESSO

A aplicação do processo ENgAGED tem por objetivo realizar um estudo de caso na qual jogos educacionais são produzidos sendo alinhados a aspectos do design de jogos, como identificação do público-alvo, levantamento dos objetivos de desempenho, desenvolvimento da estratégia instrucional e avaliação da unidade instrucional (BRANCH, 2009). Espera-se também que os jogos educacionais desenvolvidos estejam alinhados aos aspectos do design de jogos, como concepção e design dos elementos do jogo, criação das narrativas, regras e mecânica do jogo, e também a realização de testes (FULLERTON, 2008, SCHELL, 2008).

### 5.2 EXECUÇÃO DA APLICAÇÃO DO PROCESSO

Foram desenvolvidos 4 jogos educacionais utilizando o processo ENgAGED. Os jogos foram desenvolvidos por alunos de graduação dos cursos de Ciência da Computação e Sistemas de Informação da Universidade Federal de Santa Catarina, como trabalho de conclusão de curso (TCC). O processo de desenvolvimento foi acompanhado pela professora da disciplina *INE5427 - Planejamento e Gestão de Projetos* e por um aluno de doutorado. O acompanhamento foi realizado a partir de reuniões periódicas com os alunos enquanto os alunos cursavam as disciplinas de Projetos I e II de TCC. Os jogos desenvolvidos foram *SCRUM-Scape* (CAMARGO, 2013), *SCRUM'ed* (SCHNEIDER, 2015), *PM Quiz* (CASSETTARI, 2015), e o *Fuga de Gambix* (JUNCKES, 2016).

A seguir são apresentados os jogos desenvolvidos de modo resumido. A descrição completa do desenvolvimento dos jogos é apresentada no Apêndice I.

#### **Jogo SCRUM-Scape**

O jogo *SCRUM-Scape* (CAMARGO, 2013) foi desenvolvido no segundo semestre de 2013 utilizando a primeira versão do processo

ENgAGED. A primeira versão do ENgAGED possuía 11 fases e 34 atividades, tendo apenas descrições em alto nível das atividades. Esta primeira versão foi influenciada pela modelagem das fases e atividades dos processos de desenvolvimento de jogos educacionais encontrados na literatura.

A partir do desenvolvimento do jogo *SCRUM-Scape* foi eliminada uma fase *Implantação do jogo* que foi integrada a fase de *Execução da Unidade Instrucional*. Em relação as atividades da fase de *desenvolvimento do jogo* foram identificadas atividades erradas, que seriam tarefas de uma mesma atividade, como *definir objetivos do jogo*, *definir regras*, *definir níveis*. Estas atividades foram remodeladas, e inseridas na fase de *concepção do jogo* e as atividades erradas foram corrigidas e inseridas como tarefas da atividade *conceber o jogo*. No Apêndice J apresenta as fases e atividades da primeira versão do processo ENgAGED.

Após o desenvolvimento do jogo *SCRUM-Scape* foi criada uma nova versão do processo contendo 10 fases (5 design instrucional e 5 do design de jogos) e 19 atividades. No entanto, as atividades ainda apresentaram descrições de detalhamento, informando como deveriam ser executadas e sem exemplos.

Na tabela 16 apresenta-se uma visão geral do processo de desenvolvimento do jogo *SCRUM-Scape* desenvolvido com a primeira versão do ENgAGED.

Tabela 16 – Desenvolvimento do jogo *SCRUM-Scape*.

<p><b>Fase 1. Análise da Unidade Instrucional (UI)</b></p> <p>O jogo tem por objetivo ensinar SCRUM para os alunos da disciplina de gerência de projetos do curso de graduação em Ciências da Computação. Foi definido como contexto da unidade instrucional que seria aplicada aos alunos da 6ª fase do curso de graduação de Ciência da Computação, na disciplina de Gerência de Projetos. Considerando que a disciplina possui 72 horas/aula, foi criada uma aula de 1 hora e 40 minutos para aplicar o jogo em sala de aula.</p> <p>Os objetivos de desempenho da unidade instrucional são:</p> <ol style="list-style-type: none"> <li>1. Lembrar os nomes dos principais artefatos, cerimônias e papéis do SCRUM.</li> <li>2. Entender o objetivo dos artefatos e suas relações com as cerimônias.</li> <li>3. Lembrar as responsabilidades dos papéis do SCRUM Master, Development Team e Product Owner e as suas relações com os artefatos.</li> <li>4. Entender os objetivos das cerimônias durante um projeto.</li> </ol>
<p><b>Fase 2. Projeto da Unidade Instrucional (UI)</b></p> <p>A estratégia instrucional utilizada foi o desenvolvimento de um jogo para ensinar SCRUM, tendo por objetivo avaliar o conhecimento dos alunos nos seguintes conteúdos:</p> <ol style="list-style-type: none"> <li>1. Responsabilidades de Papéis: SCRUM Master, Development Team e Product Owner.</li> <li>2. Artefatos: Product Backlog, Sprint Backlog e Taskboard.</li> <li>3. Cerimônias: Daily SCRUM, Sprint Retrospective, Sprint Review e Sprint Planning.</li> </ol>

### Fase 3. Desenvolvimento do Jogo Educacional

#### Fase 3.1. Análise do Jogo

De modo geral os requisitos do jogo são: a aplicação do jogo não pode ultrapassar o tempo de um encontro em sala de aula, o qual possui aproximadamente 2 horas; ter como gênero RPG; ter como plataforma computador pessoal e sistema operacional Windows; ser jogado de modo individual; a medida que o jogador avança precisa avançar no conteúdo.

#### Fase 3.2. Concepção do Jogo

Na fase de concepção foram criadas narrativas de acordo com o perfil dos alunos, ou público-alvo, definido na fase de análise. Em relação ao layout do jogo foi definido a utilização de um cenário baseado em uma prisão formada por 3 blocos. Tendo como desafio do jogador passar pelos 3 blocos para vencer o jogo. No primeiro bloco liberta os personagens aliados das celas da prisão e nos blocos subsequentes luta contra monstros.

#### Fase 3.3. Design do Jogo

Foi definida a utilização da *game engine* RPG Maker<sup>5</sup> para desenvolver o jogo. A partir da *game engine* utilizada foi possível desenvolver os elementos do jogo (cenário e personagens) com os recursos que a própria ferramenta de criação de jogos disponibiliza.

#### Fase 3.4. Implementação do Jogo

A implementação do jogo continuou sendo realizada na *game engine* RPG Maker, não sendo necessária a codificação das telas, uma vez que a engine permite a compilação do jogo em um arquivo executável.



#### Fase 3.5. Testes do Jogo

A medida que os cenários foram construídos realizava-se testes para verificar a funcionalidade do jogo e também identificar possíveis erros na sequência lógica dos desafios, erros de português nas mensagens trocadas entre os personagens do jogo. Os testes foram realizados pelo professor da disciplina, aluno que desenvolveu o jogo e pelo aluno de doutorado que acompanhou o desenvolvimento.

### Fase 4. Execução da Unidade Instrucional (UI)

O jogo foi aplicado em uma disciplina de Gerência de Projetos na Universidade Federal de Santa Catarina (UFSC) durante uma aula e com um grupo de profissionais de TI sendo realizado à distância. A execução ocorreu em novembro de 2013 envolvendo um total de 17 participantes. Durante a execução, foi primeiramente explicada as regras do jogo e em seguida os participantes jogaram o jogo individualmente.

<sup>5</sup>RPGMaker: [www.rpgmakerweb.com](http://www.rpgmakerweb.com)

<b>Fase 5. Avaliação da Unidade Instrucional (UI)</b>
---

A avaliação seguiu modelo de avaliação de jogos educacionais MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011). O planejamento da avaliação foi realizado em paralelo com o planejamento da execução do jogo. A coleta de dados ocorreu ao final da execução por meio de um questionário online. Os dados foram analisados conforme definido no MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011) e as sugestões de melhoria foram utilizadas para melhorar o SCRUM-Scape. No Apêndice I apresenta-se os resultados da avaliação do jogo.
---

Nesta primeira aplicação do processo ENgAGED destaca-se o desenvolvimento de uma unidade instrucional contendo início, meio e fim, como orientando no processo de design instrucional (BRANCH, 2009). Em relação aos aspectos do design de jogos, o processo não descrevia em detalhes suas fases e atividades. Contudo, esta limitação foi compensada pela utilização da *game engine* RPG *Maker* que disponibiliza gratuitamente elementos do jogo (personagens e cenários).

### Jogo SCRUM'ed

O jogo *SCRUM'ed* (SCHNEIDER, 2015) foi desenvolvido utilizando a versão completa do processo ENgAGED, contendo descrições detalhadas das fases e atividades. Na tabela 13 apresentada no capítulo 4 mostra um exemplo de detalhamento das atividades do processo ENgAGED. Neste detalhamento apresenta-se o nome da atividade, a descrição da atividade, as tarefas a serem realizadas pela atividade (uma atividade pode conter diversas tarefas) e os produtos de trabalho (entrada e saída). Ao longo do desenvolvimento do jogo *SCRUM'ed* as descrições das atividades, tarefas foram melhorados tendo como objetivo deixar claro *o que deve ser feito* e apresentando exemplos *de como fazer*.

Este jogo também é utilizado como principal exemplo de aplicação do ENgAGED estando disponível para consulta no relatório técnico *Aplicação do Processo ENgAGED: Desenvolvimento do Jogo Educacional SCRUM'ed* descrito por Battistella, Schneider e Wangenheim (2015). Na tabela 17 apresenta-se uma visão geral do processo de desenvolvimento do jogo SCRUM-Scape.

Tabela 17 – Desenvolvimento do jogo *SCRUM'ed*.

<b>Fase 1. Análise da Unidade Instrucional (UI)</b>
---

O contexto do jogo é o curso de graduação em Ciências da Computação, da Universidade Federal de Santa Catarina (UFSC), em Florianópolis Brasil. O jogo foi desenvolvido para aplicação na disciplina de Planejamento e Gestão de Projetos de Software, tipicamente realizado pelos alunos do 6º ano do curso de Computação. Como pré-requisito da disciplina.
---

<p>espera-se que os alunos tenham cursado disciplinas introdutórias de Engenharia de Software. Para aplicação do jogo, foi estimado um total de 25 alunos, entre os 16 e 30 anos. A duração da aplicação do jogo é de 2 horas/aula, sendo desenvolvido para aplicação após aulas expositivas de SCRUM.</p> <p>Ao final da aplicação do jogo, espera-se que os alunos sejam capazes de:</p> <ol style="list-style-type: none"> <li>1. Compreender os Papéis: SCRUM Master.</li> <li>2. Compreender os Artefatos: Sprint Planning, Daily Scrum, Sprint Review.</li> <li>3. Compreender as Cerimônias: Product Backlog, Sprint Backlog, Task-board.</li> </ol>
<p><b>Fase 2. Projeto da Unidade Instrucional (UI)</b></p> <p>Nesta fase de design da unidade instrucional o conteúdo deve ser apresentado e organizado para ser utilizado no jogo.</p> <p>O conteúdo abordado no jogo é o SCRUM, que é uma metodologia ágil de desenvolvimento de software. Esta metodologia visa simplificar o desenvolvimento de softwares com base na transparência de informação entre os membros da equipe, inspeção frequente do que está sendo desenvolvido e ser adaptável para mudanças que ocorrem ao longo do projeto (LARMAN, 2003), (SCHWABER e BEEDLE, 2001), (SCHWABER e SUTHERLAND, 2013). O SCRUM apresenta papéis, artefatos e cerimônias.</p>
<p><b>Fase 3. Desenvolvimento do Jogo Educacional</b></p>
<p><b><i>Fase 3.1. Análise do Jogo</i></b></p> <p>Entre os principais requisitos funcionais levantados no desenvolvimento do jogo SCRUM'ed, destaca-se: (a) conter ao longo do jogo os conceitos básicos dos papéis, artefatos e cenários do SCRUM; (b) avaliar o jogador em relação ao domínio do conteúdo do jogo; (c) o jogador deve ser penalizado quando realizar ações erradas durante o jogo.</p> <p>Entre os principais requisitos não funcionais levantados no desenvolvimento do jogo, destaca-se: (a) o jogo deve ter duração de até 100 minutos; (b) deve ser um jogo digital e em 3D; (c) o jogo deve rodar no sistema operacional Windows 7 ou versões superiores; (d) O jogo não deve apresentar conteúdo difamatório, obsceno, ofensivo, odioso ou violento.</p>
<p><b><i>Fase 3.2. Concepção do Jogo</i></b></p> <p>O objetivo do jogo SCRUM'ed é liderar uma equipe SCRUM para resolver desafios em um cenário medieval. O gênero do jogo é <i>Role-playing-game</i> (RPG), sendo desenvolvido para computadores <i>stand-alone</i> e modo de interação <i>singleplayer</i>.</p> <p>Entre as principais regras do jogo, destaca-se: o jogador principal (avatar) deve interagir com os personagens e executar as atividades solicitadas pelo cliente. Para cada atividade resolvida com sucesso e interações bem sucedidas com os personagens do jogo, o jogador ganha cenouras.</p> <p>O jogo apresenta a seguinte narrativa: o personagem principal comanda a equipe SCRUM em um cenário medieval. Desde o primeiro nível o jogador deve interagir com os personagens para conhecer os membros da equipe. Em seguida o cliente entra no cenário do castelo e apresenta as atividades que a equipe deve executar. Após uma reunião, a equipe parte para o jardim do palácio para realizar as atividades solicitadas pelo cliente. No final do dia, os membros da equipe retornam ao castelo e avaliam o resultado do trabalho.</p>
<p><b><i>Fase 3.3. Design do Jogo</i></b></p> <p>O jogo SCRUM'ed foi desenvolvido utilizando game engine Unity 3D. Esta engine foi escolhida por apresentar diversos <i>assets</i> (elementos de jogo) disponíveis para baixar gratuitamente e também por apresentar uma ampla gama de comunidades e fóruns na internet para tirar dúvidas de desenvolvimento.</p>

Nesta fase os personagens foram desenhados. O jogador terá o papel do SCRUM Master e comandará a equipe. O Tristan que é o Product Owner, responsável pelo gerenciamento do Product Backlog. O Homer, que é um dos membros da equipe SCRUM.

Exemplo de personagem e cenários produzidos:



#### **Fase 3.4. Implementação do Jogo**

Nesta fase os elementos do jogo, personagens, cenários, artefatos e outros objetos são implementados. No caso do jogo SCRUM'ed os elementos são implementados utilizando a ferramenta Unity 3D. Alguns personagens utilizados no jogo foram adquiridos a partir do Unity Store, que permite baixar ou comprar diversos personagens de jogo prontos.

Exemplo de personagem e cenários implementados:



#### **Fase 3.5. Testes do Jogo**

Na fase de testes foram realizados dois tipos de testes. O primeiro foi o teste de sistema, sendo realizado pelos autores do jogo, principalmente pelo autor que implementou o jogo. Neste busca-se falhas e erros no jogo, simulando o usuário final. O segundo foi o teste de aceitação, sendo realizado por alguns componentes do laboratório GQS (Grupo de Qualidade de Software) da Universidade Federal de Santa Catarina (UFSC). Nestes testes são realizadas simulações de operações rotineiras no jogo, para verificar se o comportamento do jogo está de acordo com o solicitado.

#### **Fase 4. Execução da Unidade Instrucional (UI)**

A aplicação do jogo foi realizada em duas etapas. Na primeira, foi realizada a aplicação do jogo com os membros do GQS – Grupo de Qualidade de Software da Universidade Federal de Santa Catarina (UFSC). A aplicação do jogo ocorreu em maio de 2015, contando com a participação de 10 alunos de graduação e pós-graduação dos cursos de Computação. A segunda aplicação foi realizada na disciplina INE5427 - Planejamento e Gestão de Projetos, do curso de Bacharelado em Ciências da Computação da UFSC em junho de 2015, com 13 alunos em um laboratório de informática do Departamento de Informática e Estatística (INE) da UFSC.

#### **Fase 5. Avaliação da Unidade Instrucional (UI)**

Para a avaliação do jogo SCRUM'ed foi utilizado o modelo de avaliação de jogos educacionais MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011). De modo geral os alunos avaliaram positivamente os jogos em relação a motivação, experiência de usuário e aprendizagem. No Apêndice I apresenta-se os resultados da avaliação do jogo.

Nesta aplicação do processo ENgAGED as fases e atividades apresentaram maior grau de detalhamento. Com base nesta versão completa foram realizadas diversas iterações das fases de concepção, design e implementação do jogo. Por exemplo, a primeira versão do jogo não apresentava narrativa, regras e os elementos do jogo estavam sendo desenvolvidos. A medida que as narrativas e regras do jogo eram inseridas, os níveis do jogo se tornavam mais claros e definidos. Foram realizadas diversas versões do jogo, algumas buscavam corrigir erros de conteúdo e outros buscavam melhorias na interação dos personagens e narrativa.

### Jogo PM Quiz

O jogo *PM (Project Management) Quiz* (CASSETTARI, 2015) foi desenvolvido em paralelo ao desenvolvimento do jogo *SCRUM-‘ed*. No desenvolvimento deste jogo, não foram necessárias modificações no detalhamento das fases e atividades do processo ENgAGED.

No desenvolvimento deste jogo, foi realizada uma customização de um *quiz game* para ensinar conceitos de gerência de projetos. Nesta customização foi utilizada a ferramenta online *Kahool*<sup>6</sup>. Na tabela 18 apresenta-se uma visão geral do processo de desenvolvimento do jogo *PM Quiz*.

Tabela 18 – Desenvolvimento do jogo *PM Quiz*.

<b>Fase 1. Análise da Unidade Instrucional (UI)</b>
Em relação a contexto ou ambiente onde a unidade instrucional será aplicada foi identificado: (a) aplicação na disciplina INE5427 Planejamento e Gestão de Projetos/INE5617 Gerência de Projetos, (b) tem o objetivo de “proporcionar ao aluno uma compreensão dos principais conceitos de gerenciamento de projetos seguindo o PMBOK”,(c) tem uma quantidade estimada de 25 alunos, tendo uma variação tipicamente entre 15 à 30 alunos, (d) a unidade instrucional terá a duração de 1 hora e 40 minutos, e (e) apresenta como conteúdo programático: (1) visão geral dos grupos de processos: iniciação, planejamento, execução, monitoramento e encerramento e (2) visão geral de uma das áreas de conhecimento: tempo, custo, escopo, riscos, partes interessadas, qualidade, recursos humanos, integração, comunicações ou aquisições.
<b>Fase 2. Projeto da Unidade Instrucional (UI)</b>
A estratégia instrucional definida é um jogo educacional adaptando um <i>quiz game</i> para revisar o aprendizado em gerenciamento de projetos. Este <i>quiz game</i> será aplicado na unidade instrucional assim que os conteúdos tenham sido introduzidos. No caso das disciplinas Planejamento e Gestão de Projetos o jogo será aplicado após o conteúdo da área de conhecimento de escopo de gerenciamento de projetos tenha sido exposto. Através do uso do laboratório, notebooks ou <i>smartphones</i> pessoais, o jogo pode ser aplicado e os alunos poderão revisar efetivamente o conteúdo, respondendo às perguntas.

<sup>6</sup> <https://getkahoot.com/>

<b>Fase 3. Desenvolvimento do Jogo Educacional</b>
<b>Fase 3.1. Análise do Jogo</b>
<p>A seguir são apresentados alguns requisitos levantados para o jogo são:</p> <ol style="list-style-type: none"> <li>1. Conter perguntas referentes às áreas de conhecimento e grupos de processos alinhados ao PMBOK.</li> <li>2. Após o acontecimento de uma rodada, o professor pode manter o jogo parado para falar sobre a pergunta que acabou de ser respondida pelos alunos.</li> <li>3. Após responder a sequência de perguntas do jogo, o jogador terá seu relatório final (<i>feedback</i>). Poderá saber sua taxa de acertos, taxa de acerto dos outros alunos, classificação.</li> <li>4. O jogador poderá acompanhar estatísticas dos outros jogadores, como classificação, acertos, e respostas mais dadas.</li> <li>5. O jogo não deve ter custo com licença.</li> <li>6. O jogo deve ser compatível com os sistemas operacionais: Windows 7 e versões superiores; Ubuntu 12.04 e versões superiores; Android; IOS;</li> <li>7. O jogo será web e terá suporte completo para os <i>browsers</i> a partir da seguinte versão: Chrome 39+, Firefox 30+, Internet Explorer 10+, Safari 7+ e Android Browser 4+.</li> </ol> <p>Para versões dos browsers Chrome 37-8, Firefox 29-30, Internet Explorer 9 e Safari 5-6, o jogo está com suas características limitadas.</p>
<b>Fase 3.2. Concepção do Jogo</b>
<p>O jogador deve responder cada questão dentro de um tempo limite. O jogador que fizer mais pontos respondendo mais perguntas corretas no menor tempo é o vencedor. O gênero do jogo é <i>quiz</i>, onde o jogador precisa responder perguntas para uma determinada área de conhecimento. A plataforma utilizada é computador online e smartphone. As perguntas definidas no jogo estão apresentadas no Apêndice I.</p>
<b>Fase 3.3. Design do Jogo</b>
<p>Inicialmente foram pesquisadas diversas ferramentas que serviam como modelos de <i>quiz games</i>. Entre as ferramentas pesquisadas, foi selecionado o <i>quiz game Kahoot</i>. O <i>Kahoot</i> é uma plataforma de <i>quiz games</i> web, onde é possível cadastrar grupos de perguntas e respostas, lançar sessões de jogos para os jogadores jogarem e convidar jogadores para estas sessões, onde o jogador poderá responder às perguntas do <i>quiz game</i>. Esta ferramenta se destacou por sua facilidade de manutenção, <i>layout</i> agradável, boa usabilidade, possibilidade de uso online, adaptabilidade para dispositivos móveis, presença dos relatórios dos <i>quiz</i> aplicados e interatividade.</p>
<b>Fase 3.4. Implementação do Jogo</b>
<p>A partir das perguntas criadas na fase de concepção do jogo, elas foram inseridas no <i>Kahoot</i>, como apresentado na figura abaixo:</p>
<b>Fase 3.5. Testes do Jogo</b>
<p>Foram realizados testes no jogo, inicialmente sendo realizado pelo aluno de TCC que desenvolveu o jogo. Em seguida, foram realizadas simulações do jogo com alunos de</p>



<p>graduação e pós-graduação do Grupo de Qualidade de Software (GQS)/UFSC. Os erros encontrados estavam relacionados apenas ao conteúdo do jogo e foram corrigidos antes da aplicação em sala de aula.</p>
<p><b>Fase 4. Execução da Unidade Instrucional (UI)</b></p>
<p>O jogo foi aplicado em uma disciplina de Gerência de Projetos na Universidade Federal de Santa Catarina (UFSC). Teve a duração de 40 minutos e contou com a participação de 29 alunos de graduação. Os sistemas operacionais dos computadores utilizados pelos alunos eram Windows 7 e Ubuntu 12.04. Os alunos também jogaram com seus <i>smartphones</i> com os sistemas operacionais Android e iOS.</p>
<p><b>Fase 5. Avaliação da Unidade Instrucional (UI)</b></p>
<p>O jogo foi avaliado por meio do questionário de avaliação do MEEGA (SAVI; WANGENHEIM; BORGATTO, 2011). De modo geral, o jogo apresentou <i>feedback</i> positivo dos alunos em relação a motivação, experiência do usuário e aprendizagem. No Apêndice I apresenta-se os resultados da avaliação do jogo.</p>

Nesta aplicação do processo ENgAGED foi utilizado para customizar uma ferramenta de *quiz game* denominada *Kahoot*. Mesmo sem a necessidade de executar as atividades de design dos elementos do jogo ou implementação do jogo, o processo demonstrou ser flexível sem exigir a execução de todas as atividades do processo.

### Jogo Fuga de Gambix

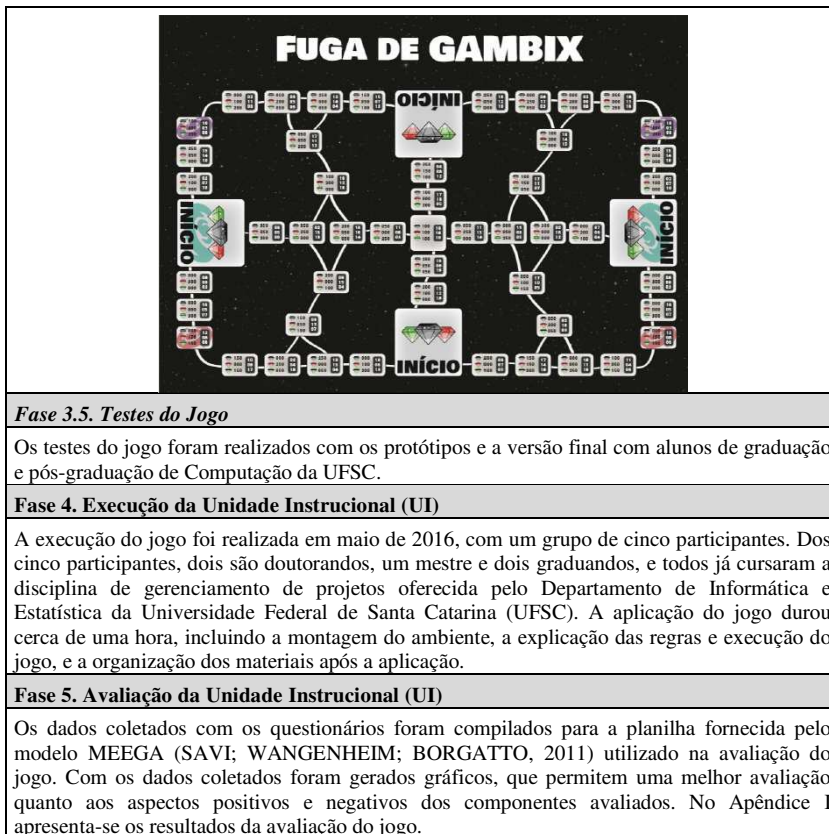
O jogo *Fuga de Gambix* (JUNCKES, 2016) foi desenvolvido utilizando a versão completado processo ENgAGED. Nesta versão utilizada para desenvolver o jogo as fases e atividades eram detalhadas, tendo como base o desenvolvimento dos três jogos educacionais. No entanto, na fase de *desenvolvimento do jogo* foi necessário detalhar as descrições das atividades que tratavam do design de jogos não-digitais.

Diferentemente dos outros 3 jogos desenvolvidos, este jogo é não-digital. Na tabela 19 apresenta-se uma visão geral do processo de desenvolvimento do jogo *Fuga de Gambix*.

Tabela 19 – Desenvolvimento do jogo *Fuga de Gambix*.

<p><b>Fase 1. Análise da Unidade Instrucional (UI)</b></p>
<p>A unidade instrucional será aplicada na disciplina Planejamento e Gestão de Projetos do curso de Ciência da Computação e Sistemas de Informação da Universidade Federal de Santa Catarina (UFSC). A faixa etária do público-alvo é geralmente entre 19 e 30 anos. Tipicamente, este tipo de público-alvo já possui contato com jogos em geral. Além disso, assume-se que o mesmo tenha recebido uma introdução com relação a parte de planejamento em gerenciamento de projetos antes de ser apresentado o jogo, na própria disciplina no qual o jogo deverá ser aplicado. O objetivo de aprendizado do jogo desenvolvido não abrange nenhum aprendizado de conteúdo ou habilidade, mas uma mudança de atitude e em nível afetivo.</p>
<p><b>Fase 2. Projeto da Unidade Instrucional (UI)</b></p>

<p>A estratégia instrucional escolhida foi a de aprendizado por experiência, através de um jogo educacional. Serão abordados no jogo, como conteúdo a análise de objetivos e o planejamento de ações. Com base no objetivo a ser alcançado, deve-se realizar um planejamento, de modo a conseguir encontrar o melhor meio de alcançar sucesso (vencer). Os alunos irão interagir principalmente competindo com os outros jogadores, mas realizando a jogada de forma individual, com feedback e troca de experiências com os outros participantes ao longo do jogo.</p>
<p><b>Fase 3. Desenvolvimento do Jogo Educacional</b></p>
<p><b>Fase 3.1. Análise do Jogo</b></p>
<p>Seguindo a atividade levantar requisitos do jogo do processo, são apresentados abaixo os requisitos do jogo:</p> <ul style="list-style-type: none"> <li>• Como requisito do jogo, foi definido que ele deve ser jogável entre 4-6 pessoas;</li> <li>• O jogo deve também ser aplicável em um tempo aproximado de 50 min., o equivalente há uma hora/aula em média;</li> <li>• Deve ser um jogo não-digital para possibilitar a jogada sem uso de computadores, estimulando também a interação social entre os alunos;</li> <li>• O custo por jogo não deve ser mais do que 150 reais;</li> <li>• Possuir em sua mecânica, regras ou elementos que façam com que quem planeje, chegue ao final com uma chance de vitória maior do que quem não planejou.</li> </ul>
<p><b>Fase 3.2. Concepção do Jogo</b></p>
<p>O jogo tem por objetivo levar o jogador a completar a missão de coletar a quantidade de recursos atribuída a ele. O jogo é do gênero de estratégia, sendo desenvolvido em uma plataforma não-digital e de tabuleiro. O modo de interação é <i>multiplayers</i>. Destaca-se como regras do jogo a movimentação do jogador pelo tabuleiro coletando os recursos indicados nas casas que ele ocupar. As cartas obtidas alteram opções de coleta ou movimentação, e valem apenas para a rodada que for utilizada. Ao final do jogo, ganha quem alcançar o objetivo antes de 14 rodadas. Caso isso não aconteça, um critério de desempate baseado na quantidade de recursos inúteis coletados definirá um vencedor.</p>
<p><b>Fase 3.3. Design do Jogo</b></p>
<p>O jogo segue desde o início do seu desenvolvimento a ideia de que o jogador deve coletar recursos para vencer. Na primeira versão, o jogo tinha a ideia de caminhos fixos, aonde o jogador ia caminhando, e a cada rodada coletava os recursos. Aplicações desta versão detectaram que esta possuía caminhos muito longos, que não ajudavam no planejamento, e era pouco divertido. Uma nova versão, ainda seguindo a ideia de ter caminhos, que uma vez escolhidos, percorria-se nele coletando recursos a cada rodada, e ao fim do caminho, escolhia-se o próximo.</p>
<p><b>Fase 3.4. Implementação do Jogo</b></p>
<p>O jogo possui como elementos que devem ser comprados os peões e dado de 20 números. Os demais elementos (cartas e tabuleiro) devem ser impressos. O tabuleiro deve ser impresso em papel A3. A seguir apresenta a imagem do tabuleiro do jogo:</p>



Nesta aplicação do processo ENgAGED destaca-se como forte sua flexibilidade para desenvolver um jogo não-digital, sendo que os processos encontrados na revisão sistemática da literatura limitam-se ao desenvolvimento de jogos digitais.

### 5.3 ANÁLISE DA APLICAÇÃO DO PROCESSO

A partir do processo ENgAGED, foram desenvolvidos 4 jogos educacionais para o ensino de Computação. No desenvolvimento do primeiro jogo (*SCRUM-Scape*) o processo estava em sua primeira versão, simplificada, contendo fases e atividades descritas em alto nível. Ao longo do desenvolvimento do jogo, algumas fases e atividades repetitivas e ambíguas foram retiradas. Nesta primeira versão, o processo foi estruturado principalmente com fases e atividades do processo ADDIE e dos processos encontrados na literatura. Destaca-se

que o jogo desenvolvido a partir desta primeira versão alcançou os objetivos esperados como estratégia instrucional. A utilização de uma *game engine* contribuiu para tornar o desenvolvimento do jogo mais rápido, uma vez que os cenários e personagens utilizados eram da própria ferramenta, não sendo necessário realizar o design dos elementos do jogo. Por outro lado, a utilização da *game engine* limitou o desenvolvimento às funcionalidades disponíveis nela.

A partir do segundo jogo desenvolvido (*SCRUM'ed*) foi utilizada a versão completa do processo ENgAGED. Nesta versão, as fases e atividades estão alinhadas ao design instrucional ADDIE (BRANCH, 2009) e ao design de jogos apresentados por Fullerton (2008) e Schell (2008). Esta versão do processo permite descrever de modo detalhado os objetivos do jogo, gênero, plataforma, narrativa, regras, mecânica e *feedback* educacional. Além disso, a última versão permite a prototipação dos elementos do jogo (personagens, cenários e artefatos). Enquanto o jogo *SCRUM'ed* foi desenvolvido, foram realizadas reuniões de acompanhamento com o aluno de graduação. Com isso, as fases e atividades do processo foram descritas de forma mais clara e detalhada.

No desenvolvimento do jogo *PM Quiz*, o processo se apresentou flexível e permitiu a customização do jogo a partir da plataforma *Kahoot*. No caso do desenvolvimento do jogo *Fuga de Gambix*, o processo se apresentou útil e flexível para o desenvolvimento de jogos não-digitais.

Portanto, a partir do desenvolvimento dos 4 jogos educacionais observa-se que o processo ENgAGED contribui para o desenvolvimento de jogos alinhados aos aspectos instrucionais e do design de jogos, sem apresentar as lacunas como os processos encontrados na revisão da literatura.



## 6 AVALIAÇÃO DO PROCESSO ENGAGED

Este capítulo apresenta a avaliação do processo ENgAGED. A avaliação é dividida em duas fases: avaliação da qualidade do processo e avaliação da qualidade do produto desenvolvido pelo processo.

### 6.1 DEFINIÇÃO DE QUALIDADE

Em um processo de desenvolvimento de jogos, espera-se características e propriedades fundamentais para garantir a qualidade do produto que está sendo desenvolvido (CROSBY, 1979; JURAN, 1979; ACUÑA e FERRÉ, 2001; OREHOVACKI e BABIC, 2015). Existem diversos conceitos para definir a qualidade de um produto. O conceito utilizado pela Organização Internacional de Normalização (ISO) é: “*o total de propriedades ou características de um produto ou serviço que são relevantes, satisfazendo requisitos específicos ou necessidades óbvias*” (NBRISO/IEC9126-1, 2003).

Um conceito para qualidade que pode ser aplicado ao desenvolvimento de jogos é a “*conformidade a requisitos funcionais e de desempenho explicitamente declarados, a padrões de desenvolvimento claramente documentados e a características implícitas que são esperadas de todo software profissionalmente desenvolvido*” (PRESSMAN, 2009). Outro conceito também utilizado para qualidade é a “*rigorosa especificação dos processos que serão utilizados na produção de um bem ou serviço, incluindo as faixas de tolerância desejada dos resultados*” (OLIVEIRA, 2004). Portanto, é fundamental que o processo de desenvolvimento seja formado por características específicas que garantam a qualidade do que está sendo desenvolvido. Assim, é possível identificar se as atividades do processo foram seguidas, assegurando o resultado conforme o planejado (SOMMERVILLE, 2007).

Segundo Rittgen (2010) para um processo ter qualidade é necessário as seguintes características:

- **Autenticidade:** é o grau que indica se o processo consegue representar realisticamente o domínio a qual foi modelado.
- **Corretude:** é o grau que indica se as instruções do processo estão corretas.
- **Compleitude:** é o grau que indica se as informações das fases e atividades estão completas.

- **Relevância:** é o grau que indica se as instruções do processo são relevantes para resolução do problema.

Para Matook e Induska (2009) um processo que apresente qualidade precisa conter as seguintes características:

- **Completude:** é o grau em que todos os componentes estão presentes no escopo do processo, sem faltar informações.
- **Compreensibilidade:** é o grau em que os conceitos e estrutura do processo são claros aos usuários.
- **Flexibilidade:** é o grau em que o processo se adapta às mudanças, permitindo ser aplicado em contextos distintos ao que foi modelado.
- **Generalidade:** é o grau em que o processo realiza uma ampla gama de funções e é utilizável em casos distintos.
- **Usabilidade:** é o grau de facilidade de usar ou aplicar o processo modelado.

Para Porter, Votta e Basili (1995) é necessário utilizar procedimentos sistemáticos para a detecção de falhas em um produto, aplicando-se também ao contexto do desenvolvimento de jogos. Os autores apresentam características necessárias para inspecionar softwares, tornando possível a detecção de falhas em sistemas.

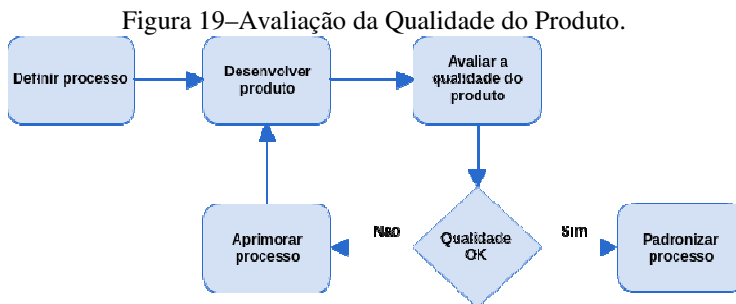
Contudo, identificar se o processo está sendo utilizado conforme orienta-se suas fases e atividades é um problema que influencia na qualidade de um processo. Muitas vezes afirma-se utilizar o processo de desenvolvimento, mas na prática utiliza-se de modo *ad-hoc*, desconsiderando o conjunto de fases e atividades sistematicamente modeladas. Este tipo de comportamento é estudado amplamente na Teoria da Ação Racional (*Theory of Reasoned Action - TRA*). Esta teoria busca avaliar a intenção comportamental por meio da avaliação das *crenças, atitudes em relação ao comportamento, motivação para obedecer a uma determinada crença, a intenção comportamental e o comportamento real* (DAVIS, BAGOZZI e WARSHAW, 1989).

Para a área tecnológica foi desenvolvido o modelo TAM (*Technology Acceptance Model*) que é uma adaptação do TRA, especificamente modelado para avaliar a aceitação em sistemas de informação. O TAM fornece uma explicação sobre determinados comportamentos demonstrando o grau de aceitação do uso de tecnologias. Ele apresenta uma base para análise do impacto de fatores

externos e crenças pessoais na utilização de tecnologias (DAVIS, 1989). O modelo apresenta dois pontos de avaliação de comportamento dos usuários: utilidade e usabilidade (DAVIS, 1989, DAVIS, BAGOZZI e WARSHAW, 1989).

Com base nos autores citados, a avaliação da qualidade do processo ENgAGED leva em consideração as seguintes características: ambiguidade, consistência, completude, compreensibilidade, correteza, flexibilidade, usabilidade e utilidade.

Além da avaliação da qualidade do processo, também é necessário avaliar a qualidade do produto desenvolvido pelo processo. A partir da avaliação da qualidade do produto, identificam-se as inconsistências levando a revisão do processo, como apresentado na figura 19 (SOMMERVILLE, 2007).



Fonte: SOMMERVILLE, 2007.

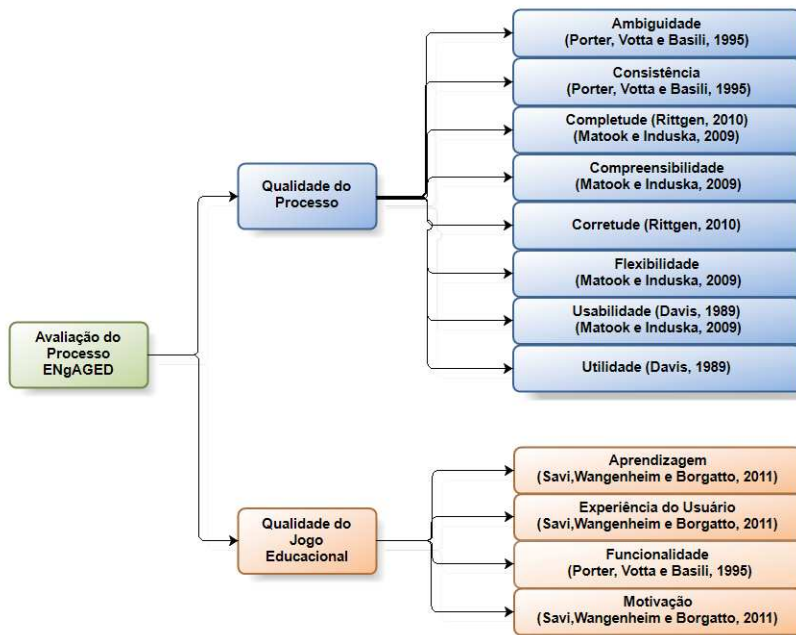
Com objetivo de avaliar também a qualidade do produto criado pelo processo ENgAGED, utiliza-se o modelo de avaliação de jogos educacionais MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011). O MEEGA é um modelo reconhecido na área de Engenharia de Software, sendo um dos principais modelos para avaliar a qualidade dos jogos educacionais (CALDERÓN e RUIZ, 2015). O modelo avalia o impacto de um jogo educacional em termos de motivação dos alunos, experiências dos usuários e aprendizagem. Nesta pesquisa a característica de funcionalidade também será utilizada na avaliação do da qualidade produto.

A avaliação da qualidade do processo (ENgAGED) é realizada por professores ou alunos que desenvolvem jogos. A avaliação da qualidade do produto (jogos educacionais) é realizada pelos alunos que jogaram os jogos em sala de aula. Na figura 20, apresenta-se a composição da avaliação do processo ENgAGED com base em Rittgen



(2010), Matook e Induska (2009), Porter, Votta e Basili (1995), Davis (1989) e Savi, Wangenheim e Borgatto (2011).

Figura 20– Composição da Avaliação.



Fonte: Elaborada pelo autor.

## 6.2 DEFINIÇÃO DA AVALIAÇÃO

Nesta seção, apresenta-se as definições da avaliação do processo ENgAGED, a qual utiliza o método de avaliação GQM (*Goal, Question, Metric*) (BASILI, CALDIERA e ROMBACH, 1994). A estrutura hierárquica do GQM é composta por três níveis. O primeiro é o nível conceitual que identifica os objetivos que se deseja alcançar com a avaliação do processo. O segundo é o nível operacional, que apresenta questões de análise para caracterizar o objeto estudado em relação aos objetivos definidos para avaliação. O terceiro é o nível quantitativo, que apresenta métricas para mediar cada questão. Portanto, ao utilizar o método GQM é necessário definir os objetivos da avaliação, questões de análise para avaliar se o objeto estudado alcançou os objetivos.

A avaliação apresenta dois objetivos, o primeiro é *avaliar a qualidade do processo ENgAGED do ponto de vista do criador de jogos*. Considera-se como criador de jogos, o professor ou aluno (orientado por um professor) que desenvolve o jogo educacional.

O segundo objetivo é *avaliar a qualidade dos jogos educacionais por meio da percepção dos alunos a respeito dos níveis de motivação, experiência do usuário e aprendizagem promovidos por um jogo*. As questões de análise e suas respectivas métricas são criadas com base nas características definidas por Rittgen (2010), Matook e Induska (2009), Porter, Votta e Basili (1995), Davis (1989) e Savi, Wangenheim e Borgatto (2011).

As perguntas e métricas para avaliação do jogo educacional utilizou como base o MEEGA, na qual avalia a motivação dos alunos por meio da atenção, relevância, confiança e satisfação (ARCS - *Attention, Relevance, Confidence, Satisfaction*). Ele também avalia a experiência dos usuários em jogos por meio da imersão, desafio, competência, divertimento, controle e imersão social. Ele avalia a aprendizagem dos alunos por meio do conhecimento, compreensão, aplicação, aprendizado de curto prazo e aprendizado de longo prazo (SAVI, WANGENHEIM e BORBATTO, 2011).

A tabela 20 apresenta os objetivos das avaliações da qualidade do processo e da qualidade do produto, decompondo as questões de análise e métricas para avaliação do processo ENgAGED.

Tabela 20– Objetivo, questões de análise e métricas para avaliação do processo ENgAGED.

<b>Objetivo 1</b>	<i>Avaliar a qualidade do processo ENgAGED, sob o ponto de vista do criador de jogo.</i>
<b>Ambiguidade</b>	
<b>Pergunta</b>	<b>P1. A descrição do processo é ambígua?</b>
Métrica	MP1.1 Quantidade de descrições ambíguas.
<b>Consistência</b>	
<b>Pergunta</b>	<b>P2. O processo é consistente?</b>
Métrica	MP2.1 Quantidade de inconsistência por fases, atividades e produtos de trabalho.
<b>Completo</b>	
<b>Pergunta</b>	<b>P3. O processo é completo?</b>
Métrica	MP3.1 Quantidade de fases, atividades e produtos de trabalho faltando.
<b>Compreensibilidade</b>	
<b>Pergunta</b>	<b>P4. O processo é compreensível?</b>
Métrica	MP4.1 Quantidade de descrições incompreensíveis por fases, atividade e produtos de trabalho.
<b>Corretude</b>	

<b>Pergunta</b>	<b>P5. O processo é correto?</b>
Métrica	MP5.1 Quantidade de erros no processo por fases, atividade e produtos de trabalho. MP5.2 Erros corrigidos.
<b>Flexibilidade</b>	
<b>Pergunta</b>	<b>P6. O processo é aplicável para o desenvolvimento de jogos digitais e não-digitais?</b>
Métrica	MP6.1 Quantidade de jogos digitais e não-digitais desenvolvidos pelo processo. MP6.2 Quantidade de gêneros de jogos desenvolvidos pelo processo. MP6.3 Quantidade de descrições das fases, atividades e produtos de trabalho que foram modificados para cada tipo específico de jogo. MP6.4 Grau de Facilidade para adaptação do processo para um determinado jogo.
<b>Usabilidade</b>	
<b>Pergunta</b>	<b>P7. O processo é fácil de usar?</b>
Métrica	MP7.1 Grau de usabilidade do processo.
<b>Utilidade</b>	
<b>Pergunta</b>	<b>P8. O processo é útil?</b>
Métrica	MP8.1 Grau de utilidade do processo.
<b>Objetivo 2</b>	<i>Avaliar a qualidade dos jogos educacionais por meio da percepção dos alunos a respeito dos níveis de motivação, experiência do usuário e aprendizagem promovidos por um jogo.</i>
<b>Funcionalidade</b>	
<b>Pergunta</b>	<b>P9. O processo é funcional?</b>
Métrica	MP9.1 Grau de suporte de aspectos educacionais no desenvolvimento do jogo.
<b>Motivação</b>	
<b>Pergunta</b>	<b>P10. Os alunos ficaram satisfeitos após a aplicação do jogo?</b>
Métrica	MP10.1 Grau de satisfação dos alunos.
<b>Pergunta</b>	<b>P11. Os alunos sentiram-se confiantes ao jogar o jogo?</b>
Métrica	MP11.1 Grau de confiança dos alunos.
<b>Pergunta</b>	<b>P12. Os alunos consideram o jogo relevante?</b>
Métrica	MP12.1 Grau de relevância do jogo.
<b>Pergunta</b>	<b>P13. Os alunos mantiveram atenção durante a aplicação do jogo?</b>
Métrica	MP13.1 Grau de atenção dos alunos ao jogar o jogo.
<b>Experiência do usuário</b>	
<b>Pergunta</b>	<b>P14. Os alunos apresentam as competências necessárias para jogar o jogo?</b>
Métrica	MP14.1 Grau de competência dos alunos.
<b>Pergunta</b>	<b>P15. Os alunos consideram o jogo divertido?</b>
Métrica	MP15.1 Grau de diversão dos alunos.
<b>Pergunta</b>	<b>P16. Os alunos sentiram-se desafiados a vencer o jogo?</b>
Métrica	MP16.1 Grau em que os alunos sentiram-se desafiados.
<b>Pergunta</b>	<b>P17. O jogo apresentou interação social aos alunos?</b>
Métrica	MP17.1 Grau de interação social do jogo
<b>Pergunta</b>	<b>P18. O jogo permitiu a imersão dos alunos no jogo?</b>
Métrica	MP18.1 Grau de imersão dos alunos.
<b>Aprendizagem</b>	
<b>Pergunta</b>	<b>P19. Os alunos consideram que o jogo auxiliou no seu aprendizado?</b>

Métrica	MP19.1 Grau de aprendizado dos alunos.
Pergunta	<b>P20. Os alunos consideram que alcançaram os objetivos de aprendizagem do jogo?</b>
Métrica	MP20.1 Grau em que os alunos alcançaram os objetivos de aprendizagem.

A partir das questões de análise e métricas levantadas, identifica-se a necessidade de coletar dados para avaliar a qualidade do processo do ponto de vista do criador de jogos (objetivo 1) e avaliar a qualidade dos jogos educacionais por meio da percepção dos alunos a respeito dos níveis de motivação, experiência do usuário e aprendizagem promovidos por um jogo (objetivo 2). Para levantar os dados do objetivo 1, foi realizada uma inspeção por um painel de especialistas analisando o processo em relação as métricas definidas (BRUIN e ROSEMAN, 2007). O questionário para levantar os dados referente a inspeção está no Apêndice F.

Para levantar os dados referentes ao objetivo 2, foi utilizado o modelo MEEGA, sendo possível realizar um estudo de caso, levantando os dados em forma de auto-avaliação dos alunos após a aplicação dos jogos educacionais desenvolvidos pelo processo ENgAGED. Antes das avaliações dos jogos, foram realizadas aulas-expositivas para explicar o conteúdo aos alunos. Depois de todo conteúdo ser apresentado, aplicou-se o jogo educacional oferecendo também a oportunidade dos alunos revisarem o conteúdo abordado nas aulas-expositivas, em seguida os alunos realizaram a auto-avaliação do jogo aplicado em sala de aula. O questionário utilizado para avaliação do segundo objetivo está no Apêndice G.

A tabela 21 detalha a relação entre as métricas apresentadas na tabela 20 com as perguntas do questionário apresentados nos Apêndices F e G. O questionário referente ao objetivo 1 é chamado de Q1 e o questionário referente ao objetivo 2 é chamado de Q2.

Tabela 21—Relação das métricas e questões.

Métrica	Perguntas para o Objetivo 1	Escala de Resposta
MP1.1 Quantidade de descrições ambíguas.	Q1- pergunta: 6) Você identificou algo ambíguo no processo?	Sim ou Não
MP1.1 Quantidade de descrições ambíguas.	Q1- pergunta: 7) Quais fases, atividades e produtos de trabalho você considera ambíguos?	Fases, atividades e produtos de trabalho com problema.
MP2.1 Quantidade de inconsistência por fases, atividades e produtos de	Q1- pergunta: 8) Você identificou fases, atividades e produtos de trabalho inconsistentes?	Sim ou Não

trabalho.		
MP2.1 Quantidade de inconsistência por fases, atividades e produtos de trabalho.	Q1- pergunta: 9) Quais fases, atividades e produtos de trabalho são considerados inconsistentes?	Fases, atividades e produtos de trabalho com problema.
MP3.1 Quantidade de fases, atividades e produtos de trabalho faltando.	Q1-pergunta: 10) Existem fases, atividades e produtos de trabalho faltantes que tornam o processo incompleto?	Sim ou Não
MP3.1 Quantidade de fases, atividades e produtos de trabalho faltando.	Q1-pergunta: 11) Quais fases, atividades e produtos de trabalho são incompletos?	Fases, fases, atividades e produtos de trabalho incompletos.
MP4.1 Quantidade de descrições incompreensíveis por fases, atividade e produtos de trabalho.	Q1- pergunta: 12) Você identifica fases, atividades e produtos de trabalho incompreensíveis?	Sim ou Não
MP4.1 Quantidade de descrições incompreensíveis por fases, atividade e produtos de trabalho.	Q1- pergunta: 13) Quais fases, atividades e produtos de trabalho são considerados incompreensíveis?	Fases, atividades e produtos de trabalho com problema.
MP5.1 Quantidade de erros no processo por fases, atividade e produtos de trabalho.	Q1- pergunta: 14) Você identifica erros no processo?	Sim ou Não
MP5.1 Quantidade de erros no processo por fases, atividade e produtos de trabalho.	Q1- pergunta: 15) Quais fases, atividades e produtos de trabalho você identifica erros?	Fases, atividades e produtos de trabalho com problema.
MP5.2 Erros corrigidos.	Q1- pergunta: 15) Quais fases, atividades e produtos de trabalho você identifica erros?	Fases, atividades e produtos de trabalho com problema.
MP6.1 Quantidade de jogos digitais e não-digitais desenvolvidos pelo processo.	Q1-perguntas: 16) Você desenvolveu jogo digital com o processo ENgAGED?  17) Você desenvolveu jogo não-digital com o processo ENgAGED?	Sim ou Não para as perguntas 16 e 17.
MP6.2 Quantidade de gêneros de jogos desenvolvidos pelo processo.	Q1- pergunta: 18) Caso tenha desenvolvido um jogo com ENgAGED, qual foi gênero?	Opções: <b>Ação-aventura:</b> Um jogo que exigem que os jogadores utilizem reflexos rápidos, com precisão para superar obstáculos, resolver desafios ou para responder perguntas.

		<p><b>Adivinhação:</b> Um jogo em que o objetivo é identificar algum tipo de informação, como uma palavra, a partir de desenhos ou imitação de outros jogadores.</p> <p><b>Corrida:</b> Um jogo em que o jogador comanda um veículo ou participa de uma corrida, tentando se mover mais rápido do que os oponentes para alcançar um objetivo específico ou ser mais rápido que um determinado tempo.</p> <p><b>Estratégia:</b> Um jogo que é focado na jogabilidade e exige pensamentos táticos e cuidados, a fim de conseguir a vitória.</p> <p><b>Puzzle:</b> Um jogo que envolve personagens que controlam um conjunto de comandos, para navegar através de labirintos ou para organizar os objetos de maneira correta.</p> <p><b>Quiz:</b> Um jogo onde o jogador precisa responder perguntas para uma determinada área de conhecimento.</p> <p><b>Role-playing game (RPG):</b> Um jogo onde o jogador controla ações de um protagonista e com este personagem vive imerso em um mundo fictício. Em jogo de role-playing os personagens interagem com este mundo e ficam mais fortes.</p> <p><b>Roll-and-move:</b> Jogos de tabuleiro em que as fichas são movidas com base nos resultados mostrados em um ou mais dados (por exemplo: <i>Snake and Ladders</i> e <i>Monopoly</i>).</p> <p><b>Simulação:</b> Um jogo desenvolvido para colocar o jogador no controle de um determinado ambiente ou atividade, o qual busca ser o</p>
--	--	--

		mais realista possível.			
MP6.3 Quantidade de descrições das fases, atividades e produtos de trabalho que foram modificados para cada tipo específico de jogo.	Q1- pergunta: 19) Quais fases, atividades e produtos de trabalho foram modificados para cada tipo de jogo?	Fases, atividades e produtos de trabalho com problema.			
MP6.4 Grau de facilidade para adaptação do processo para um determinado jogo.	Q1- pergunta: 20) Acho que consigo facilmente adaptar o processo para o desenvolvimento de um determinado tipo de jogo?	Sim ou Não			
MP7.1 Grau de usabilidade do processo.	Q1- perguntas: 21) Acho que consigo desenvolver um jogo mais rapidamente quando utilizo o processo ENgAGED? 22) Acho que consigo melhorar o meu desempenho no desenvolvimento de jogos, utilizando o processo ENgAGED? 23) Acho que consigo aumentar a minha produtividade no desenvolvimento do jogo, utilizando o processo ENgAGED? 24) Acho que consigo aumentar a eficácia no desenvolvimento de jogos, utilizando o processo ENgAGED? 25) Acho que consigo tornar mais fácil o desenvolvimento de jogos, utilizando o processo ENgAGED?	Escala para as perguntas de 21 à 31:			
		Fortemente Concordo	Concordo	Discordo	Discordo Fortemente
MP8.1 Grau de utilidade do processo.	Q1- pergunta: 26) Acho que é fácil aprender a usar o processo ENgAGED? 27) Acho que é fácil obter informações do processo ENgAGED para fazer o que desejo? 28) Acho que consigo interagir com o processo ENgAGED de modo claro e compreensível? 29) Acho que o processo ENgAGED é flexível? 30) Acho que poderia me tornar habilidoso usando o processo ENgAGED? 31) Acho que o processo ENgAGED é realmente fácil de usar para o desenvolvimento de jogos?				

Métrica	Perguntas para o Objetivo 2	Escala de Resposta					
MP9.1 Grau de suporte de aspectos educacionais no desenvolvimento do jogo.	Q2-perguntas: 25) O jogo contribuiu para a minha aprendizagem na disciplina? 26) O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina? 27) A experiência com o jogo vai contribuir para meu desempenho na vida profissional?	Escala para as perguntas de 25 à 27: <table border="1" data-bbox="664 252 938 400"> <tr> <td data-bbox="664 252 785 300">Discordo Fortemente</td> <td data-bbox="790 252 841 400">-2 -1 0 +1 +2</td> <td data-bbox="846 252 938 323">Concordo Fortemente</td> </tr> </table>			Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente
Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente					
MP10.1 Grau de satisfação dos alunos.	Q2-perguntas: 9) Estou satisfeito porque sei que terei oportunidades de utilizar na prática coisas que aprendi com o jogo? 10) É por causa do meu esforço pessoal que consigo avançar no jogo?						
MP11.1 Grau de confiança dos alunos.	Q2-perguntas: 7) Foi fácil entender o jogo e começar a utilizá-lo como material de estudo? 8) Ao passar pelas etapas do jogo senti confiança de que estava aprendendo?						
MP12.1 Grau de relevância do jogo.	Q2-perguntas: 4) O conteúdo do jogo é relevante para os meus interesses? 5) O funcionamento deste jogo está adequado ao meu jeito de aprender? 6) O conteúdo do jogo está conectado com outros conhecimentos que eu já possuía?						
MP13.1 Grau de atenção dos alunos ao jogar o jogo.	Q2-perguntas: 1) O design do jogo é atraente (interface ou objetos, como cartas ou tabuleiros)? 2) Houve algo interessante no início do jogo que capturou minha atenção? 3) A variação (de forma, conteúdo ou de atividades) ajudou a me manter atento ao jogo?						
MP14.1 Grau de competência dos alunos.	Q2-perguntas: 23) Consegui atingir os objetivos do jogo por meio das minhas habilidades? 24) Tive sentimentos positivos de eficiência no desenrolar do jogo?						
MP15.1 Grau de diversão dos alunos.	Q2-perguntas: 20) Quando interrompido, fiquei desapontado que o jogo tinha acabado (gostaria de jogar mais)?						



	<p>21) Eu recomendaria este jogo para meus colegas?</p> <p>22) Gostaria de utilizar este jogo novamente?</p>	
MP16.1 Grau em que os alunos sentiram-se desafiados.	<p>Q2-perguntas:</p> <p>17) Este jogo é adequadamente desafiador para mim, as tarefas não são muito fáceis nem muito difíceis?</p> <p>18) O jogo evolui num ritmo adequado e não fica monótono – oferece novos obstáculos, situações ou variações de atividades?</p>	
MP17.1 Grau de interação social do jogo	<p>Q2-perguntas:</p> <p>14) Pude interagir com outras pessoas durante o jogo?</p> <p>15) Me diverti junto com outras pessoas?</p> <p>16) O jogo promove momentos de cooperação e/ou competição entre as pessoas que participam?</p>	
MP18.1 Grau de imersão dos alunos.	<p>Q2-perguntas: 11, 12, 13</p> <p>11) Temporariamente esqueci- das minhas preocupações do dia-a-dia, fiquei totalmente concentrado no jogo?</p> <p>12) Eu não percebi o tempo passar enquanto jogava, quando vi o jogo acabou?</p> <p>13) Me senti mais no ambiente do jogo do que no mundo real, esquecendo do que estava ao meu redor?</p>	
MP19.1 Grau de aprendizado dos alunos.	<p>Q2-perguntas:</p> <p>25) O jogo contribuiu para a minha aprendizagem na disciplina?</p> <p>26) O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina?</p> <p>27) A experiência com o jogo vai contribuir para meu desempenho na vida profissional?</p>	
MP20.1 Grau em que os alunos alcançaram os objetivos de aprendizagem.	<p>Q2-perguntas:</p> <p>29) Lembra-se dos conceitos/métodos (antes da aplicação do jogo)?</p> <p>30) Lembra-se dos conceitos/métodos (depois da aplicação do jogo)?</p> <p>31) Compreensão ao funcionamento dos conceitos / métodos (antes da aplicação do jogo)?</p> <p>32) Compreensão ao funcionamento</p>	

	dos conceitos / métodos (depois da aplicação do jogo)? 33) Compreensão de como aplicar os conceitos / métodos (antes da aplicação do jogo)? 34) Compreensão de como aplicar os conceitos / métodos (depois da aplicação do jogo)?	
--	---	--

As avaliações realizadas pelos alunos após as aplicações dos jogos desenvolvidos foram aprovadas pelo CEPESH – Comissão de Ética em Pesquisa com Seres Humanos da UFSC sob o parecer de número 1.021.539. O apêndice H apresenta o TCLE – Termo de Consentimento Livre e Esclarecido.

### 6.3 EXECUÇÃO DA AVALIAÇÃO DA QUALIDADE DO PROCESSO ENGAGED

Para avaliação da qualidade do processo foi realizada uma avaliação com 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. 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 realizar a avaliação, foram convidados 50 professores e alunos que desenvolvem ou utilizam jogos educacionais no Brasil. Para selecionar estes especialistas foram pesquisados artigos em congressos e periódicos nacionais, envolvidos em pesquisas na área da informática na educação, mas principalmente para o desenvolvimento ou aplicação de jogos educacionais. Também foram selecionados para a avaliação, os alunos que desenvolveram jogos educacionais utilizando o ENgAGED.

A avaliação ficou disponível on-line entre outubro de 2014 a março de 2015, sendo utilizada a ferramenta *LimeSurvey* para disponibilizar o questionário on-line. Para realizar a avaliação foi entregue aos especialistas o relatório técnico do ENgAGED (BATTISTELLA e WANGENHEIM, 2015) apresentando o processo em detalhe, juntamente com o questionário de avaliação (Apêndice F). 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.

#### 6.4 ANÁLISE DA AVALIAÇÃO DA QUALIDADE DO PROCESSO ENGAGED

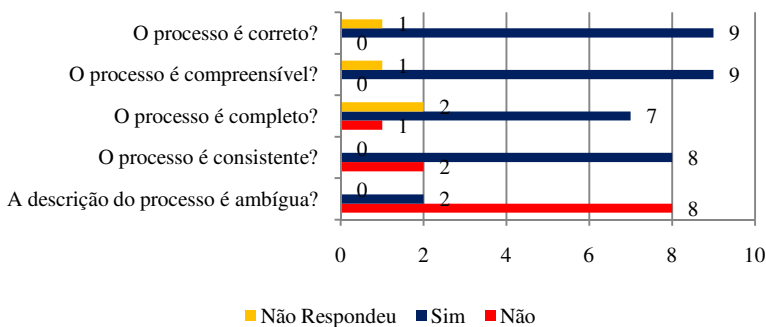
De modo geral, o resultado da avaliação indica que o processo apresenta boa qualidade, do ponto de vista da não-ambiguidade, consistência, completude, compreensibilidade, corretude, flexibilidade, utilidade e usabilidade.

Observa-se com base na figura 21 que o processo é correto e compreensível para o desenvolvimento de jogos educacionais. Em geral, os especialistas consideraram o processo completo, mas 1 especialista considerou incompleto. O especialista afirmou que o processo não permite o desenvolvimento de jogos que motivem ou reforcem algum conteúdo de Computação. Esta afirmação é uma inverdade porque na fase de análise é possível definir o nível de aprendizagem conforme a Taxonomia de Bloom. E também, entre os jogos desenvolvidos com o ENgAGED, o jogo *Fuga de Gambix* tem como objetivo motivar os alunos para o planejamento de projetos.

Em relação a consistência, grande parte dos especialistas consideraram o processo consistente. Apenas 2 especialistas consideraram inconsistente. Um especialistas sugeriu que as atividades de jogos digitais e não-digitais fossem separadas. No capítulo 4 apresenta-se a tabela 14 com uma visão geral do processo tendo as atividades para jogos digitais e não-digitais separadas. Outro especialista considerou inconsistente porque as fases do design instrucional não são iterativas. No entanto, com base no design instrucional ADDIE todas as fases podem ser revisadas ao longo da criação da unidade instrucional.

Em relação a ambiguidade, grande parte dos especialistas consideraram o processo não-ambíguo. Porém, 2 especialistas consideraram ambíguo. Um deles considerou ambíguo porque não compreendeu a tarefa de contextualização da unidade instrucional na fase de análise da unidade instrucional, na qual é necessária a criação de um fluxograma contextualizando o conteúdo dentro da disciplina. No entanto, esta tarefa é importante para que outros professores utilizem a unidade instrucional e consigam contextualizar o jogo na disciplina que será ministrada. O outro especialista considerou ambíguo porque não compreendeu a diferença entre as atividades de levantamento de requisitos e concepção do jogo. No entanto, não apresentou comentários claros do que poderia ser melhorado.

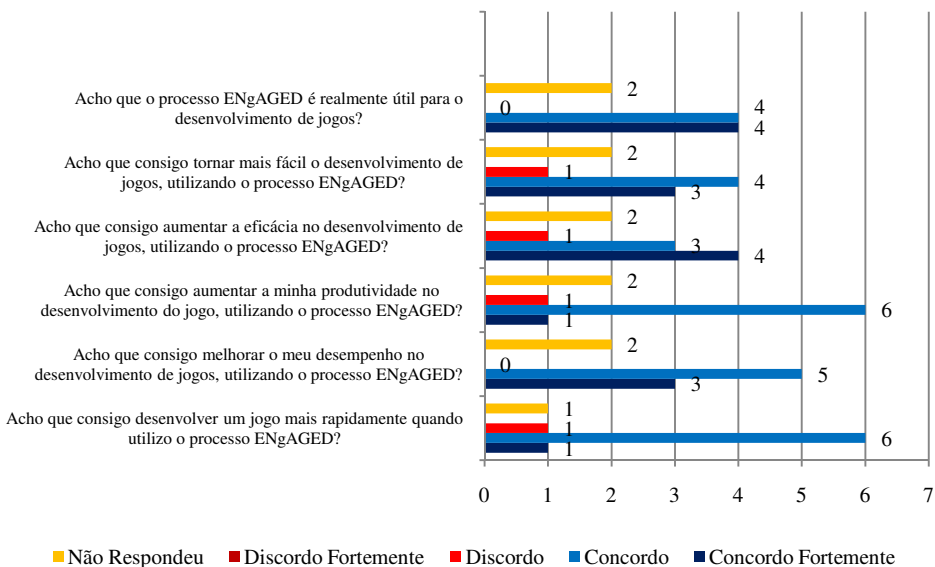
Figura 21 – Gráfico da avaliação da qualidade do processo. Características: corretude, compreensibilidade, completude, consistência e ambiguidade.



Fonte: Elaborada pelo autor.

Em relação a utilidade, observa-se na figura 22 que o processo é útil. Principalmente para aumentar a produtividade, melhorar o desempenho e produzir jogos mais rapidamente.

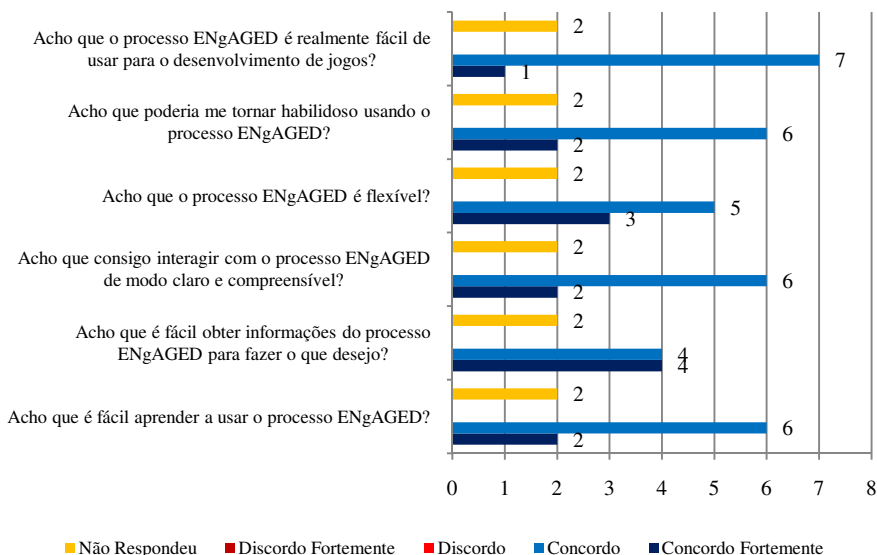
Figura 22–Gráfico da avaliação da qualidade do processo. Característica: utilidade.



Fonte: Elaborada pelo autor.

Em relação a usabilidade, observa-se com base na figura 23 que o processo apresenta boa usabilidade, com destaque para a facilidade de uso, habilidade no uso e interação clara e compreensível.

Figura 23–Gráfico da avaliação da qualidade do processo. Características: usabilidade.



Fonte: Elaborada pelo autor.

Em relação a flexibilidade, 4 especialistas desenvolveram jogos educacionais utilizando o processo ENgAGED, sendo 3 digitais e 1 não-digital. Os gêneros desenvolvidos foram RPG, Estratégia e *Quiz Game*. Entre os 8 especialistas que desenvolver e não desenvolveram jogos utilizando o ENgAGED apenas 1 afirmou que não seria fácil adaptar o processo ao desenvolvimento de um determinado jogo. Este especialista não descreveu comentários que poderiam melhorar o processo.

O resultado desta avaliação inicial realizada por um painel de especialistas fornece uma primeira indicação positiva da qualidade do processo ENgAGED. Na avaliação, os especialistas foram questionados, por meio de uma pergunta descritiva e subjetiva, quais foram os aspectos que mais gostaram do processo. A seguir são apresentados os aspectos principais relatados pelos especialistas:

- Apresentar bom nível de detalhamento das atividades.

- Apresentar definições claras do que deve ser feito em cada atividade.
- Apresentar equilíbrio entre os aspectos educacionais, técnicos e de diversão.
- Apresentar exemplos para facilitar o desenvolvimento dos jogos.
- Apresentar *templates* e exemplos disponíveis no relatório técnico.
- Estar alinhado ao design instrucional.
- Possuir flexibilidade nas fases de desenvolvimento do jogo.
- Ser abrangente para o desenvolvimento de diferentes tipos de jogos.
- Ser fácil de compreender o que deve ser feito em cada atividade.

Como principal ponto fraco do processo, destaca-se que os especialistas consideraram o processo extenso, recomendando a criação de uma versão resumida. Assim, no capítulo que define o processo ENgAGED (capítulo 4) apresenta a tabela *Visão geral do processo ENgAGED* tendo como objetivo resumir as fases e atividades. Uma versão resumida também é apresentada no artigo *ENgAGED: Um Processo de Desenvolvimento de Jogos para Ensinar Computação* (Battistella e Wangenheim, 2016a).

## 6.5 EXECUÇÃO DA AVALIAÇÃO DA QUALIDADE DO JOGO EDUCACIONAL

A execução da avaliação levou em consideração os 4 jogos educacionais desenvolvidos a partir do processo ENgAGED: *SCRUM-Scape*, *SCRUM'ed*, *PM Quiz*, e o *Fuga de Gambix*. A avaliação foi realizada após as aplicações dos jogos em sala de aula (pós-teste) e utilizou o questionário definido no modelo MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011) como instrumento para coleta de dados. As avaliações foram realizadas nas disciplinas de Gerência de Projetos dos cursos de graduação em Ciência da Computação e Sistemas de Informação da Universidade Federal de Santa Catarina (UFSC). Além dos alunos das disciplinas, algumas avaliações também foram realizadas com ex-alunos da disciplina no Grupo de Qualidade de Software (GQS)/UFSC. Levando em consideração as aplicações dos

jogos e em seguida as avaliações, a média de duração da unidade instrucional foi de 50 minutos.

A avaliação do jogo *SCRUM-Scape* foi realizada no segundo semestre de 2013 e contou com a participação de 17 alunos. A avaliação do jogo *SCRUM'ed* foi realizada no primeiro semestre de 2015 e estiveram presentes 23 alunos. A avaliação do jogo *PM Quiz* foi realizada no primeiro semestre de 2015 e contou com a participação de 33 alunos. A avaliação do jogo *Fuga de Gambix* foi realizada no primeiro semestre de 2016 e estiveram presentes 5 alunos.

## 6.6 ANÁLISE DA AVALIAÇÃO DA QUALIDADE DO JOGO EDUCACIONAL

A avaliação dos jogos educacionais foi realizada com base em 78 questionários (Apêndice G) preenchidos pelos alunos após as aplicações dos jogos. Com base nas avaliações observa-se que os jogos foram avaliados positivamente em relação a motivação, experiência do usuário e aprendizagem. As avaliações apresentaram resultados positivos, mesmo levando em consideração as diferentes versões do ENgAGED utilizadas no desenvolvimento dos jogos. A maior diferente entre as versões ocorreu entre a versão utilizada no desenvolvimento do *SCRUM-Scape* na qual não havia detalhamento das fases e atividades. Um fator que pode ter influenciado neste resultado positivo está relacionado a utilização da *game engine RPG Maker*. Com a utilização desta *game engine* não foi necessário implementar os elementos do jogo (cenários, personagens e objetivos), porque a ferramenta disponibiliza gratuitamente estes elementos.

Em relação a motivação, os jogos foram avaliados positivamente na satisfação, confiança, relevância e atenção (figura 24). Como aspectos fortes da avaliação destaca-se que os alunos conseguiram avançar nos jogos a partir dos esforços pessoais, não sendo necessárias interrupções durante as aplicações dos jogos para explicar um determinado conteúdo, ou as mecânicas e regras dos jogos. Por isso, os alunos consideraram que os jogos eram fáceis de entender. Os alunos consideraram os jogos relevantes para seus interesses pessoais e afirmaram que os designs são atraentes. Como aspecto negativo, destaca-se o jogo *Fuga de Gambix* que apresentou mediana negativa em relação ao sentimento de confiança de estar aprendendo. No entanto, este jogo não apresenta como objetivo o aprendizado de conteúdos ou habilidades, mas uma mudança de atitude e em nível afetivo (taxonomia de Bloom (1956)).

Figura 24– Medianas da avaliação da motivação.

		Mediana			
		SCRUM-Scape	SCRUMed	PM-Quiz	Fuga de Gambix
Satisfação	É por causa do meu esforço pessoal que consigo avançar no jogo	1	1	1	2
	Estou satisfeito porque sei que terei oportunidades de utilizar na prática coisas que aprendi com o jogo	1	1	1	0
Confiança	Ao passar pelas etapas do jogo senti confiança de que estava aprendendo.	1	1	1	-1
	Foi fácil entender o jogo e começar a utilizá-lo como material de estudo	2	1	2	1
Relevância	O conteúdo do jogo está conectado com outros conhecimentos que eu já possuía.	2	2	1	1
	O funcionamento deste jogo está adequado ao meu jeito de aprender	2	1	1	1
	O conteúdo do jogo é relevante para os meus interesses	1	1	2	0
Atenção	A variação (forma, conteúdo ou de atividades) ajudou a me manter atento ao jogo.	1	1	1	2
	Houve algo interessante no início do jogo que capturou minha atenção.	1	1	1	1
	O design do jogo é atraente (interface ou objetos, como cartas ou tabuleiros).	1	1	2	1

Fonte: Elaborada pelo autor.

Em geral, na avaliação da experiência do usuário, os jogos foram avaliados positivamente em relação ao desenvolvimento de competências, interação social e imersão (figura 25). Os jogos foram avaliados moderadamente, tendo medianas iguais a 0 em relação ao divertimento e desafios. No entanto, os jogos *multiplayers* (*PM-Quiz* e *Fuga de Gambix*) alcançaram maior divertimento que os jogos *singleplayer* (*SCRUM-Scape* e *SCRUM'ed*).

O jogo *SCRUM'ed* apresentou avaliações negativas em relação ao divertimento. Este resultado pode estar relacionado a narrativa do segundo nível do jogo, na qual os jogadores realizam atividades relativamente simples e que não exige muito esforço do ponto de vista lógico e de aprendizagem. Portanto, mostrando uma necessidade de melhoria no jogo, possivelmente tendo que inserir tarefas mais complexas aos jogadores. Outros aspectos de melhorias são identificados, como no jogo *SCRUM-Scape* que precisa ser melhorado



em relação ao aumento dos desafios e novos obstáculos tornando-o menos monótono. Um aspecto positivo em relação aos jogos é relacionado à imersão dos jogos, permitindo que os alunos se sentissem no ambiente do jogo, não percebendo o tempo passar. Este aspecto positivo é necessário para que os jogos educacionais sejam atrativos aos alunos e seja uma estratégia instrucional motivadora que levem os alunos a um maior engajamento na disciplina.

Figura 25– Medianas da avaliação da experiência dos usuários.

		Mediana			
		SCRUM-Scape	SCRUMed	PM-Quiz	Fuga de Gambix
Competência	Tive sentimentos positivos de eficiência no desenrolar do jogo	1	1	1	2
	Conseguir atingir os objetivos do jogo por meio das minhas habilidades.	1	1	1	1
Divertimento	Gostaria de utilizar este jogo novamente	0	0	1	1
	Eu recomendaria este jogo para meus colegas.	0	-1	1	2
	Quando interrompido, fiquei desapontado que o jogo tinha acabado (gostaria de jogar mais).	0	0	1	2
	Me diverti com o jogo.	0	-1	2	2
Desafio	O jogo evolui num ritmo adequado e não fica monótono – oferece novos obstáculos, situações ou variações de atividades.	0	0	1	1
	Este jogo é adequadamente desafiador para mim, as tarefas não são muito fáceis nem muito difíceis.	1	0	0	2
Interação Social	O jogo promove momentos de cooperação e/ou competição entre as pessoas que participam.	1	1	2	1
	Me diverti junto com outras pessoas.	1	1	1	0
	Pude interagir com outras pessoas durante o jogo.	1	1	1	1
Imersão	Me senti mais no ambiente do jogo do que no mundo real, esquecendo do que estava ao meu redor.	1	1	2	1
	Eu não percebi o tempo passar enquanto jogava, quando vi o jogo acabou.	1	1	1	1
	Temporariamente esqueci as minhas preocupações do dia-a-dia, fiquei totalmente concentrado no jogo.	1	1	1	1

Fonte: Elaborada pelo autor.

Em relação a aprendizagem, os jogos foram avaliados positivamente em relação a eficiência na aprendizagem em comparação a outras atividades e na contribuição para a aprendizagem na disciplina. No entanto, o jogo *Fuga de Gambix* foi avaliado moderadamente em relação a aprendizagem, principalmente porque o jogo apresenta como objetivo a avaliação de nível afetivo (figura 26).

Na avaliação da qualidade do produto também foi avaliada a funcionalidade com base no modelo MEEGA. É possível observar na tabela 21 as perguntas do questionário do MEEGA relacionadas com a métrica *MP9.1 Grau de suporte de aspectos educacionais no desenvolvimento do jogo*. A métrica *MP9.1* é referente a avaliação da funcionalidade. Na tabela 26 observa-se que o processo é funcional, porém apenas o jogo *Fuga de Gambix* apresenta avaliação neutra.

Figura 26– Medianas da avaliação da aprendizagem.

	Mediana			
	SCRUM-Scape	SCRUMed	PM-Quiz	Fuga de Gambix
A experiência com o jogo vai contribuir para meu desempenho na vida profissional.	0	1	0	-1
O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina.	1	1	1	0
O jogo contribuiu para a minha aprendizagem na disciplina	1	1	1	0

Fonte: Elaborada pelo autor.

## 6.7 AMEAÇAS A VALIDADE DA PESQUISA

Os resultados obtidos nesta pesquisa precisam ser interpretados com cuidado, levando em consideração ameaças à sua validade. Em relação à avaliação da qualidade do produto, destaca-se como ameaça o tamanho relativamente pequeno da amostra, podendo comprometer a generalização dos resultados. 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 sem que alguns especialistas aplicassem o processo. Para minimizar esta ameaça, foram realizados 4 estudos de caso em paralelo à modelagem do processo ENgAGED, sendo desenvolvido os jogos *SCRUM-Scape* (CAMARGO, 2013), *SCRUM'ed* (SCHNEIDER, 2015), *PM Quiz* (CASSETTARI, 2015), e o *Fuga de Gambix* (JUNCKES, 2016). Entre os 10 especialistas, 4 desenvolveram jogos educacionais utilizando o ENgAGED.

O design de pesquisa utilizado nos estudos de caso (*one-shot post-test only*) podem causar diversos problemas no que diz respeito à validade dos resultados. Um dos problemas está relacionado à falta de base para avaliar realmente o conhecimento dos alunos, porque não foi realizado um pré-teste antes da unidade instrucional ser aplicada. Este problema é mais agravado pelo fato de não haver um grupo de controle para comparar os efeitos identificados. Na aplicação dos jogos, optou-se por realizar estudos de caso, em vez de um experimento formal, a fim de minimizar as interrupções nos fluxos normais da disciplina. Isso, por um lado, reduz a validade dos resultados obtidos, mas por outro lado era a única maneira possível de executar a pesquisa.

Na pesquisa, também podem ocorrer ameaças à validade do constructo, ou seja, ameaças em relação à teoria utilizada na pesquisa. Para minimizar esta ameaça, a avaliação do processo utiliza o método de avaliação GQM (Goal, Question, Metric) (BASILI, CALDIERA e ROMBACH, 1994), para permitir a decomposição sistemática das questões de análise e métricas com base nos objetivos da avaliação. As questões de análise e as métricas foram criadas com base em características que estão relacionadas à qualidade de um processo (ambiguidade, consistência, completude, compreensibilidade, corretude, flexibilidade, funcionalidade, usabilidade e utilidade) (RITTGEN 2010, MATOOK e INDUSKA 2009, PORTER, VOTTA e BASILI 1995 e DAVIS 1989) e na qualidade dos jogos educacionais (motivação, experiência do usuário e aprendizagem) (SAVI, WANGENHEIM e BORGATTO, 2011).



## 7 CONCLUSÕES

Os jogos educacionais são considerados relevantes métodos de ensino, utilizados amplamente na área da Computação, como observado no capítulo do Estado da Arte. Tipicamente, são estratégias instrucionais usadas no apoio ao ensino, apresentando vários benefícios, como atratividade, imersão, competição, diversão e ilustram princípios abstratos.

Atualmente existem diversos jogos educacionais para o ensino de Computação, como mostra o resultado da revisão sistemática da literatura identificando 107 jogos utilizados para o ensino de Computação. Observa-se que grande parte destes jogos se concentra na área de Engenharia de Software (41 jogos). Os jogos de Engenharia de Software são normalmente do gênero de simulação e permitem que o jogador interprete o papel de um gerente de projetos que deve planejar, executar, monitorar e controlar um projeto de software. Outras áreas de conhecimento também apresentam uma quantidade considerável de jogos, como Fundamentos de Programação, Redes de Computadores, Algoritmos e Segurança. No caso de jogos para a área de Fundamentos de Computação, objetiva-se auxiliar os alunos na aprendizagem básica de programação exigindo que o jogador execute comandos de programação, como estruturas de repetição, de decisão e funções. No caso da área de Redes de Computadores, normalmente o jogador deve simular a manutenção em redes e sub-redes. No caso da área de Algoritmos e Complexidade, tradicionalmente o jogador deve executar diferentes algoritmos, ordenar um conjunto de dados ou então classificar os algoritmos conforme sua complexidade. Nas áreas como Gestão de Informação (TI), Interação Homem-Computador e Computação Distribuída, foram encontrados poucos jogos ( $\leq 3$ ). Portanto, observa-se uma oportunidade para o desenvolvimento de novos jogos, também para estas áreas pouco exploradas.

Em geral, os jogos encontrados foram desenvolvidos para diversos tipos de plataformas e gêneros. Destaca-se que a maior parte (68) são jogos de computadores (*stand alone* ou online). Por outro lado, a revisão da literatura identificou 40 jogos não-digitais de diferentes plataformas (papel & lápis, tabuleiro, cartas).

Analisando os jogos em relação aos objetivos de aprendizagem, observa-se que a grande maioria tem como objetivo, alcançar o conhecimento no nível cognitivo da taxonomia de Bloom (1956). Apenas um pequeno número de jogos visa o ensino de habilidades, como a comunicação ou trabalho em equipes, ou uma mudança de

atitude como reconhecer as dificuldades existentes na gerência de equipes de desenvolvimento de software. Analisando cada nível de aprendizagem separadamente, observa-se que a maioria dos jogos visam níveis cognitivos mais baixos da taxonomia de Bloom (1956), de acordo com as expectativas de aprendizagem em níveis de graduação (ACM/IEEE-SE, 2004). Neste contexto, os jogos foram mais usados para reforçar os conhecimentos previamente ensinados utilizando diferentes métodos de ensino.

Com base na revisão da literatura dos jogos, observou-se que os jogos normalmente não são avaliados, o que torna difícil identificar se os alunos alcançaram os objetivos de aprendizagem propostos pela unidade instrucional.

Identificou-se também que mesmo existindo alguns processos de desenvolvimento de jogos disponíveis na literatura, há diversas lacunas tanto em relação ao design instrucional, quanto ao design de jogos. Inicialmente foram encontrados em uma revisão sistemática de literatura 47 artigos sobre processos de desenvolvimento de jogos. No entanto, apenas 6 artigos descreveram em detalhes ou de modo claro suas fases e atividades. A grande parte dos artigos não descrevia detalhadamente as fases e atividades ou então descrevia características necessárias aos jogos, como diversão e aprendizagem. Entre os 6 processos encontrados, observa-se que apenas o processo apresentado por Marcos e Zagalo (2011) não é voltado ao design instrucional. Os demais processos, mesmo estando alinhados ao design instrucional, apresentaram lacunas, como não identificar as metas instrucionais (STAALDUINEN e FREITAS, 2011, SOMMEREGGER e KELLNER, 2012, LOH, 2009, KICKMEIER-RUST et al., 2006), ou então, não apresentam atividades para caracterização do público-alvo (STAALDUINEN e FREITAS, 2011, KICKMEIER-RUST et al., 2006). Outra lacuna identificada está relacionada a limitação dos processos para orientar o criador de jogos na avaliação da unidade instrucional. Tipicamente orienta-se a realização da avaliação, contudo, não são apresentadas orientações de como a avaliação deve ser conduzida ou então quais instrumentos podem ser utilizados para coleta de dados.

Em relação ao design de jogos, observa-se que o processo apresentado por Sommeregger e Kellner (2012) é mais completo em relação as atividades da concepção do jogo. O processo apresentado por Marcos e Zagalo (2011) apresenta como ponto forte a modelagem das atividades com base em processos de artes digitais. Os processos apresentados por Kirkley, Tomblin e Kirkley (2005), Loh (2009) e Kickmeier-Rust et al. (2006) são relativamente similares e contemplam

grande parte das atividades do design de jogos. Por outro lado, Staalduinen e Freitas (2011) apresenta maior limitação em relação do design de jogos em comparação aos demais processos.

Entretanto, mesmo contendo fases e atividades do design de jogos, os processos apresentam diversas lacunas, como apresentado no comparativo entre os processos encontrados na literatura e o processo ENgAGED (capítulo 4). Para suprir as lacunas dos processos encontrados, o processo ENgAGED foi proposto levando em consideração as fases e atividades dos processos encontrados na literatura, mas também complementado com fases e atividades específicas do design instrucional e design de jogos. Neste contexto, destaca-se como pontos forte do processo ENgAGED os seguintes aspectos relativos ao design instrucional: caracterização do público-alvo, definição dos objetivos de aprendizagem, revisão cíclica da estratégia instrucional, e realização da avaliação dos jogos. Na avaliação, destaca-se como ponto forte a utilização do modelo de avaliação sistemático de jogos educacionais (MEEGA (SAVI, WANGENHEIM e BORBATTO, 2011)).

Em relação ao design de jogos, o processo ENgAGED também apresenta pontos fortes, como levantamento dos requisitos do jogo, definição de aspectos fundamentais do design de jogos como definição do gênero do jogo, plataforma, modo de interação, narrativas, regras, mecânica e pontuação. O processo também permite a realização da prototipação dos elementos do jogo (como, personagens, cenários e objetos) e a realização de testes ao longo do processo de desenvolvimento.

Outro ponto forte do processo é a integração entre as atividades do design de jogos como o design instrucional. Por exemplo, definir o gênero do jogo tendo como base os níveis de aprendizagem da taxonomia de Bloom (1956). Ou então, definir o *feedback* educacional na mesma atividade em que as narrativas e regras do jogo são concebidas, permitindo a integração dos aspectos de diversão do jogo com os aspectos instrucionais.

Outro aspecto criado no processo que busca alinhar o design instrucional ao design de jogos é a atividade de modelagem do jogo. Nesta atividade cria-se fluxogramas contendo os *feedbacks* educacionais, relacionando-os com os níveis do jogo e também cria-se diagramas de sequência contendo os diálogos entre os elementos do jogo. Além de permitir a documentação, os diálogos permitem aos professores que estão orientando alunos no desenvolvimento a revisar os conteúdos dos diálogos, por exemplo, permitindo identificar algum

conceito abordado de modo errado ou então diálogo com conteúdo antiético.

Outro aspecto que pode ser considerado como ponto forte do processo, sendo desconsiderado pelos demais processos encontrados na literatura, é possibilidade de tornar os jogos acessíveis a pessoas com deficiências moderadas (acessibilidade). No ENgAGED, durante a fase de design, o criador de jogos pode optar por características em seus jogos que incluam alunos portadores de algum tipo de deficiência visual, auditiva e motora.

Outro ponto forte do processo está relacionado ao detalhamento das fases e atividades, diferentemente dos processos encontrados na literatura que apresentam apenas descrições gerais das fases e atividades. Além disso, os processos encontrados orientam apenas o desenvolvimento jogos digitais, e o ENgAGED apresenta suporte para o desenvolvimento de jogos digitais e não-digitais.

O processo ENgAGED foi avaliado com objetivo de identificar a qualidade do processo. Para avaliar o processo, foi avaliada a qualidade do processo e a qualidade do produto. A avaliação da qualidade do processo foi realizada por um painel de especialistas, professores e alunos de Computação que desenvolvem ou utilizam jogos educacionais. Com base nesta primeira avaliação do processo, destaca-se que o processo ENgAGED é não-ambíguo, consistente, completo, compreensível, correto, flexível, com boa usabilidade e útil. No entanto, os avaliadores sugeriram algumas modificações no processo para torná-lo mais claro, como apresentado no capítulo 6 de Avaliação do Processo ENgAGED. Como ponto a melhorar, os especialistas sugeriram realizar um resumo das fases e atividades. Para resumir as fases e atividades do processo, foi apresentado no capítulo 4 (Processo ENgAGED) a Tabela 14 – *Visão geral do processo ENgAGED*.

A avaliação da qualidade do produto levou em consideração os resultados das avaliações dos jogos educacionais realizadas por meio do modelo MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011). Em geral, os jogos desenvolvidos foram bem avaliados em relação a motivação, experiência do usuário e aprendizagem. Contudo, o jogo *Fuga de Gambix* foi avaliado moderadamente em relação a aprendizagem, principalmente porque o jogo apresenta como objetivo a avaliação de nível afetivo.

Como trabalhos futuros que poderão ser realizados para expandir os resultados obtidos na pesquisa, destaca-se o desenvolvimento de novos jogos educacionais para diversas áreas da Computação, utilizando o processo ENgAGED. E também, dar continuidade do



desenvolvimento dos jogos já desenvolvidos criando novos níveis e desafios, e dar continuidade a aplicação e avaliação dos jogos em sala de aula.

## REFERÊNCIAS

ABES. **Mercado Brasileiro de TI fatura US\$ 61,6 bilhões em 2013.**

Associação Brasileira das Empresas de Software (Brasil) (Org.).

Disponível em: <<http://www.abes.org.br>>. Acesso em: 24 abr. 2015.

ABES. **Investimentos em TI somam US\$ 60 bilhões em 2014.**

Associação Brasileira das Empresas de Software (Brasil) (Org.).

Disponível em: <<http://www.abes.org.br>>. Acesso em: 24 abr. 2015.

ABLUQUERQUE, João P. et al. **Educação em Sistemas de Informação no Brasil: Uma Análise da Abordagem Curricular em Instituições de Ensino Superior Brasileiras.** Revista Brasileira de Informática na Educação, v. 22, n. 1, 2014.

ABT, Clarck C. **Serious games.** University Press of America.2002.

ACM/IEEE-CS. **Computing Curricula 2005.** Estados Unidos da

América: ACM e IEEE Computer Society, 56 p., 2005.

ACM/IEEE-SE. **The Joint Task Force on Computing Curricula**

**IEEE/ACM.** Software engineering: curriculum guidelines for undergraduate degree programs in software engineering, IEEE e ACM Computer Society, 2004.

ACM/IEEE-CS. **Computer Science Curricula 2013.** Estados Unidos

da América: ACM e IEEE Computer Society, 514 p., 2013.

ACUÑA, Silva T. et al. **The Software process: modeling, evaluation and improvement.** Handbook of Software Engineering and Knowledge Engineering. World Scientific Publishing Company, 2000.

ACUÑA, Silvia T. e FERRÉ, Xavier. **Software Process Modelling.**

Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics. Orlando Florida, EUA, p. 1-6, 2001.

ADAMS, Ernest e ROLLINGS Andrew. **Fundamentals of game design, prentice hall.** Prentice Hall, 2006.

AFZAL, Wasif; TORRAR, Richard e FELDT, Robert. **A Systematic Mapping Study on Non-Functional Search-Based Software Testing**. Inf. Software Technol. v. 51, n. 6, p. 957-976, 2009.

AREVALILLO-HERRÁEZ, Miguel; MORÁN-GÓMEZ, Raul e CLAVER, José M. **Conquer the net: an educational computer game to learn the basic configuration of networking components**. Computer Applications in Engineering Education, v. 20, n. 1, p. 72-77, 2009.

BAKER, Alex; NAVARRO, Emily O. e HOEK, André v. d. **Problems and Programmers: an educational software engineering card game**. Proceedings of the 25th International Conference on Software Engineering, Portland, OR, 2003.

BAKER, Alex; NAVARRO, Emily O. e HOEK, André van der. **An experimental card game for teaching software engineering processes**. Journal of Systems and Software, v. 75, n. 1-2, p. 3-16, 2005.

BARNES, Tiffany et al. **Game2Learn: A study of games as tools for learning introductory programming concepts**. Proceedings of the 38th Technical Symposium on Computer Science Education, Covington, KY, v. 39, n. 3, p.121-125, 2007.

BARNES, Tiffany et al. **Game2Learn: improving the motivation of CS1 students**. Proceeding of the 3rd International Conference on Game Development in Computer Science Education, Miami, FL, 2008.

BARROS, Márcio de O. et al. **Model-driven game development: experience and model enhancements in software project management education**. Software Process: Improvement and Practice, v. 11, n. 4, p. 411-421, 2006.

BASIL, V.; CALDIERA, G. e ROMBACH, H.D. **Goal Question Metric Approach**. Encyclopedia of Software Engineering, p. 528-532, John Wiley & Sons, Inc., 1994.

BATTLESHIPS. **Computer Science Unplugged**. 2002. Disponível em: <http://csunplugged.org/searching-algorithms>. Acesso em: 20 jan. 2012.

BATTISTELLA, Paulo E., WANGENHEIM, Christiane G. von e FERNANDES, João. **Como jogos educacionais são desenvolvidos? Uma revisão sistemática da literatura.** Workshop sobre Educação em Computação, Brasília, 2014.

BATTISTELLA, Paulo E., SCHNEIDER, Marcelo F. e WANGENHEIM, Christiane G. von. **Aplicação do Processo ENgAGED: Desenvolvimento do Jogo Educacional SCRUM'ed.** Florianópolis: INCoD, 39 p., 2015.

BATTISTELLA, Paulo E. e WANGENHEIM, Christiane G. von. **ENgAGED: Processo de Desenvolvimento de Jogos para Ensino em Computação.** Florianópolis: INCoD, 72 p., 2015.

BATTISTELLA, Paulo E. e WANGENHEIM, Christiane G. von. **ENgAGED: Um Processo de Desenvolvimento de Jogos para Ensinar Computação.** SBIE - Simpósio Brasileiro de Informática na Educação, Uberlândia, Outubro, 2016a.

BATTISTELLA, Paulo E. e WANGENHEIM, Christiane G. von. **Caracterização do Público-Alvo de Jogos Educacionais na área da Computação.** 24º WEI - Workshop sobre Educação em Computação, Porto Alegre, Julho, 2016b.

BATTISTELLA, Paulo E. e WANGENHEIM, Christiane G. von. **Games for Teaching Computing in Higher Education – A Systematic Review.** IEEE Technology and Engineering Education, v.1, n. 3, p.8-30, 2016c.

BATTISTELLA, Paulo E. e WANGENHEIM, Christiane G. von. **Caracterização do Público-Alvo de Jogos Educacionais na área da Computação.** 24º WEI - Workshop sobre Educação em Computação, Porto Alegre, Julho, 2016b.

BECKER-KORNSTAEDT, Ulrike. **Towards systematic Knowledge Elicitation for Descriptive Software Process Modeling.** IESE Report n° 036.01/E, version 1.0, 2001.

BELFORE-II, Lee A.; ADCOCK, Amy B. e WATSON, Ginger S. **Introducing digital logic and electronics concepts in a game-like**

**environment.** Proceeding of the Spring Simulation Multiconference, San Diego, CA, 2009.

BELL, Jonathan S.; SHETH, Swapneel e KAISER, Gail E. **Secret ninja testing with HALO software engineering.** Proceeding of the 4th Int. Workshop on Social Software Engineering, Szeged, Hungary, 2011.

BENALI, Khakid e DERNIAME, Jean-Claude. **Software processes modeling: what, who, and when.** Proceedings of the Second European Workshop on Software Process Technology, 1992.

BINARY GAME. **The Cisco Learning Network.** 2008. Disponível em: <https://learningnetwork.cisco.com/docs/DOC-1803>. Acesso em: 15 jan. 2012.

BLOOM, Benjamin S. **Taxonomy of educational objectives: the classification of educational goals.** Handbook I, cognitive domain, 1956.

BONWELL, Charles C. e EISON James A. **Active Learning: Creating Excitement in the Classroom.** Eric Digests, Washington, EUA, 1991.

BOYLE, E. A. et al. **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, 2016.

BROWN, Nanette et al. **The Hard Choices Game explained white paper.** Software Engineering Institute/Carnegie Mellon University, Pennsylvania, PA, 2010.

BRANCH, Robert M. **Instructional Design: The ADDIE Approach.** New York: Springer, EUA, 203 p., 2009.

BRATHWAITE, Brenda e SCHREIBER, Ian. **Challenges for game designers.** Course Technology, CENGAGE Learning, Boston, EUA, 2009.

BRUIN, Tonia e ROSEMANN, Michael. **Using the Delphi Technique to Identify BPM Capability Areas.** 18th Australasian Conference on Information Systems, 42, 2007.

BUDGEN, David et al. **Using mapping studies in Software Engineering**. Evidence-based Software Engineering. Lancaster University, p. 195-204, 2008.

CALDERÓN, Alejandro e RUIZ, Mercedes. **A systematic literature review on serious games evaluation: An application to software project management**. Computer & Education, v. 87, 2015.

CAMARGO, André. **Jogo de RPG para ensinar SCRUM**. 62 f. TCC (Graduação) - Curso de Ciências da Computação, Departamento de Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, 2013.

CARGOBOT, 2012. Disponível em: <http://twolivesleft.com/CargoBot>. Acesso em: 15 dez. 2012.

CARVALHO, Osório P. **Educational game to support teaching interpersonal skills for software project management**. Project Thesis, Bachelor Course on Information Systems, Federal University of Santa Catarina, Brazil, 2012.

CASSETTARI, Fernando T. **Customização de um Jogo Educacional para Revisão de Conhecimento em Gerenciamento de Projetos**. 108 f. TCC (Graduação) - Curso de Sistemas de Informação, Departamento de Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, 2015.

CAULFIELD, Craig; VEAL, Davide MAJ, Stanislaw P. **Teaching Software Engineering Project Management – a novel approach for Software Engineering programs**. Modern Applied Science, v. 5, n. 5, p. 28-43, 2011.

CAULFIELD, Craig et al. **A systematic survey of games used for Software Engineering education**. Modern Applied Science, v. 5, n. 6, p. 28-43, 2011b.

CEEBOT, 2008. Disponível em: <http://www.cebobot.com/cebobot/index-e.php>. Acesso em: 18 dez. 2012.

CHAFFIN, Amanda et al. **Experimental evaluation of teaching recursion in a video game.** Proceeding of the Symposium on Video Games, Nova Orleans, LA, 2009.

CHAVES, Rafael O. et al. **Intelligent Behavior Simulation Module for Software Process Elements.** Proceeding of the Brazilian Symposium on Games and Digital Entertainment, Salvador, Brazil, 2010.

CHAVES, Rafael O. et al. **DESIGMPS: A game to support the teaching models of quality of software processes, based on conceptual maps.** Proceeding of the 17th Workshop on Informatics for Education, Aracaju, Brazil, 2011.

CLARK, D. R. **ADDIE Model.** 2009. Disponível em: <<http://www.sos.net/~donclark/hrd/isd/ADDIE.html>>. Acesso em: 7 abr. 2015.

CNIB. **Clear Print Accessibility Guidelines.** 2015. Disponível em:<<http://www.cnib.ca/accessibility>>. Acesso em: 8 jul. 2016.

COLOBOT, 2001. Disponível em: <http://www.ccebot.com/colobot/index-e.php>. Acesso em: 18 dez.2012.

CONNOLLY, Thomas M. et al. **A systematic literature review of empirical evidence on computer games and serious games.** Computers & Education, v. 59, n. 2, p. 661-686, 2012.

CONNOLLY, Thomas M.; STANSFIELD, Mark e HAINEY, Thomas. **An application of games-based learning within software engineering.** British Journal of Educational Technology, v. 38, n. 3, p. 416-428, 2007.

CORE, Kevin. **Dice and digital - rehabilitating the board game geek.** BBC News, 2013. Disponível em: <http://www.bbc.co.uk/news/business-21615083>. Acessa em: 24 jan. 2013.

CHOI, Jeomg-Im e HANNAFIN, Michael. **Situated Cognition and Learning Environments: Roles, Structures and Implications for Design.** Educational Technology Research and Development. v. 43, n. 2, p.53-69, 1995.

CRESWELL, Jhon W. **Projeto de pesquisa: métodos qualitativo, quantitativo e misto**. 2 ed. Califórnia: Artmed e Bookman, 2007. 248 p.

CROSBY, Philip B. **Quality is Free**. McGraw-Hill, London, 1979.

CYBERCIEGE, 2011. Disponível em: [cizr.nps.edu/cyberciege/](http://cizr.nps.edu/cyberciege/). Acesso em: 25 ato. 2014.

DANTAS, Alexandre; BARROS, Márcio e WERNER, Cláudia. **A simulation-based game for project management experiential learning**. Proceedings of the 16th International Conference on Software Engineering and Knowledge Engineering, Banff, Canada, 2004.

DAVIS, Fred D. **Perceived usefulness, perceived ease of use, and user acceptance of information technology**. MIS Quarterly, v. 13, n. 3, p. 319-340, 1989.

DAVIS, Fred D.; BAGOZZI, Richard P. e WARSHAW, Paul R. **User acceptance of computer technology: a comparison of two theoretical models**. Management Science, v. 35, n. 8, 982-1003 p., 1989.

DEMPSEY, John V.; RASMUSSEN, Karen e LUCASSEN, Barbara. **The Instructional Gaming Literature: Implications and 99 Sources**. Alabama: College Of Education University Of South Alabama, 1996.

DENNING, Tamara; KOHNO, Tadayoshi e SHOSTACK, Adam. **Control-Alt-Hack: a card game for computer security outreach, education, and fun**. Technical report UW-CSE-12-07-01, University of Washington, Seattle, WA, p. 1-7, 2012.

DESPIAN, Wendy. **100 Principles of Game Design**. 1ª ed. EUA: Nova Riders, 240 p., 2012.

DICK, Walter e CAREY, Lou. **The systematic design of instruction**. 4th ed., Nova York, NY: Harper Collin, 1996.

DO, Trien V. e LEE, Jong-Weon. **A multiple-level 3D-LEGO game in augmented reality for improving spatial ability**. Proceeding of the 13th International Conference on Human-Computer Interaction. San Diego, CA, 2009.



DRAPPA, Anke e LUDEWIG, Jochen. **Simulation in software engineering training**. Proceeding of the Int. Conference on Software Engineering, Limerick, Ireland, 2000.

EAGLE, Michael e BARNES, Tiffany. **Experimental evaluation of an educational game for improved learning in introductory computing**. Proceeding of the 40th Technical Symposium on Computer Science Education, Chattanooga, TN, 2009.

FERNANDES, João M. e SOUSA, Sónia M. **PlayScrum: A card game to learn the scrum agile method**. Proceeding of the 2nd Int. Conference on Games and Virtual Worlds for Serious Applications, Braga, Portugal, 2010.

FINCHER, Cameron. **Learning theory and research**. Quotations on Teaching, Learning, and Education, 1985.

FILATRO, Andrea. **Design Instrucional na Prática**. São Paulo: Pearson Education do Brasil, 2008.

FINKELSTEIN, Anthony, KRAMER, Jeff e NUSEIBEH, Bashar. **Software Process Modelling and Technology**. Research Studies Press, 1994.

FORD, Charles W. e MINSKER, Steven. **TREEZ: An educational data structures game**. Journal of Computing Sciences in Colleges, v. 18, n. 6, p. 180-185, 2003.

FREEMAN, Scott et al. **Active learning increases student performance in science, engineering, and mathematics**. Proc. of the National Academy of Sciences, v. 111, n. 23, 2014, 8410-8415 p.

FREITAS, Sara I. de. **Using games and simulations for supporting learning**. Learning, Media and Technology, Digital Games and Learning, v. 31, n. 4, p. 343-358, 2006.

FULLERTON, Tracy. **Game design workshop: A play centric approach to creating innovates games**. 3. ed. Burlington: Morgan Kaufmann, p. 470. 2008.

GAGNE, Robert M. **Some reflections on thinking skills.** Instructional Science, 17, p. 387-390, 1988.

GERAB, Fábio, GERAB, Irani F. da S. e BUENO, Ivander A. M. **Análise das Interações Curriculares em um Curso de Ciência da Computação: buscando subsídios para aprimoramento curricular.** Revista Brasileira de Informática na Educação, v. 22, n. 1, 2014.

GIL, Antonio C. **Como elaborar projetos de pesquisa.** 5ª ed. São Paulo: Atlas, 184 p., 2010.

GKRITSI, Aikaterini. **Scrum Game: an agile software management game.** Master Thesis, University of Southampton, UK.2011.

GÓMEZ-MARTÍN, Marco A. **Not yet another visualization tool: learning compilers for fun.** Proceeding of the 8th International Symposium on Computers in Education, León, Spain, 2006.

GRAVEN, Olaf H.; HANSEN, Hallstein A. e MACKINNON, Lachlan. **A computer game modeling routing in computer networks as abstract learning material in a blended learning environment.** International Journal of Emerging Technologies in Learning, 4, 2009.

HAINES, Thomas et al. **A systematic literature review of games-based learning empirical evidence in primary education.** Computers e Education. v. 9, 2016.

HAINES, Thomas et al. **Evaluation of a game to teach requirements collection and analysis in software engineering at tertiary education level.** Computers e Education, 56, p. 21-35, 2011.

HAKULINEN, Lasse. **Card games for teaching data structures and algorithms.** Proceeding of the 11th Koli Calling International Conference on Computing Education Research, Koli National Park, Finland, 2011.

HAKULINEN, Lasse. **DSasketch: Data Structures and Algorithms drawing game.** Proceeding of the 12th Koli Calling International Conference on Computing Education Research, Tahko, Finland, 2012.

HAMEY, Leonard G. C. **Using the security protocol game to teach computer network security.** Proceeding of the Symposium on UniServe Science Improving Learning Outcomes, Sydney, Australia, 2003.

HARPER, Michael, MILLER, Joseph e SHEN, Yuzhorng. **Digi Island: a serious game for teaching and learning digital circuit optimization.** NASA, Langley Research Center, EUA, p. 869-889, 2011.

HAYS, Robert T. **The effectiveness of instructional games: A literature review and discussion.** Technical report 2005-004, Naval Air Warfare Center Training Systems Division, Orlando, FL, 2005.

HAYS, Robert T.; SINGER, Michael J. **Simulation fidelity in training system design: Bridging the gap between reality and training.** Recent Research in Psychology, Springer, 415 p., 1988.

HARPER, Michael; MILLER, Joseph e SHEN, Yuzhong. **Digi Island: a serious game for teaching and learning digital circuit optimization.** NASA, Langley Research Center, EUA, p. 869-889, 2011.

HERZ, Jessie C. **Joystick nation: how videogames ate our quarters, won our hearts, and rewired our minds.** Little, Brown and Company, Boston, MA, 1997.

HICKS, Andrew. **Towards social gaming methods for improving game-based computer science education.** Proceeding of the 5th International Conference on the Foundations of Digital Games, Monterey, CA, p. 259-261, 2010.

HILL, John M.D.et al. **Puzzles and Games: addressing different learning styles in teaching operating systems concepts.** Proceeding of the 34th Technical Symposium on Computer Science Education, Reno, NV, 2003.

HURWITZ, A. e DAY, M. **Children and their art: Methods for the elementary school.** Eighth Edition. Belmont: Thomson Wadsworth, CA, EUA, 2007.

IEEE. **Reference Guide for Instructional Design**. IEEE Advancing Technology for Humanity. 2014. Disponível em: [http://www.ieee.org/education\\_careers/education/reference\\_guide/index.html](http://www.ieee.org/education_careers/education/reference_guide/index.html) Acesso em: 13 out. 14.

INNOV8, 2008. Disponível em: <http://www-01.ibm.com/software/solutions/soa/innov8/index.html>. Acesso em: 18 dez. 2012.

IT MANAGER, 2007. Disponível em: <http://itmanagerduels.intel.com>. Acesso em: 18 dez. 2012.

ITIN, Christian M. **Reasserting the philosophy of experiential education as a vehicle for change in the 21st Century**. The Journal of Experiential Education, v. 22, n. 2, p. 91-98, 1999.

ITZHAKI, Sarah e MATTHEWS, Patrick. **For Homeschoolers e Parents: Shyness in Children**. 2013. Disponível em: <http://g4ed.com/index.php/for-homeschoolers/427-shyness-in-children>. Acesso em: 20 jan. 2012.

JAIN, Apurva e BOEHM, Barry. **SimVBSE: Developing a Game for Value-Based Software Engineering**. Proceeding of the 19th Conference on Software Engineering Education e Training, Oahu, HI, 2006.

JIMÉNEZ-DÍAZ, Guillermo et al. **Pass the Ball: Game-based Learning of Software Design**. Entertainment Computing, Lecture Notes in Computer Science, Entertainment Computing, 4740, p. 49-54, 2007.

JORDAN, Craig et al. **Counter Measures: a game for teaching computer security**. Proceeding of the 10th Annual Workshop on Network and Systems Support for Games, Ottawa, Canada, 2011.

JORDAN, Hyun L. **Saving Princess Sera Game Design, Usability, and Interface**. Department of Computer Science Lynchburg College, Lynchburg, VA, 2006.

JURAN, Joseph M. **Quality Control Handbook**. McGraw-Hill, London, 1979.

JUNCKES FILHO, Paulo Jair. **GAMBIX - um jogo não-digital para motivar o planejamento de projetos**. 100 f. TCC (Graduação) - Curso de Sistemas de Informação, Departamento de Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, 2016.

KAFAI, Yasmin B. **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, IL, 2001.

KAIN, Erik. **Are board games better than video games?** Forbes Business, 2012. Disponível em: <http://www.forbes.com/sites/erikkain/2012/04/19/ar>. Acesso em: 24 jan. 2013.

KE, Fengfeng. **A qualitative meta-analysis of computer games as learning tools**. Handbook of Research on Effective Electronic Gaming in Education, IGIglobal, EUA, 2009.

KERN, Vinícius M.; STOTZ, Maria do R. e BENTO, Merilyn. **A game-playing experience in the learning of database concurrency control**. Merlot - Multimedia Educational Resource for Learning and Online Teaching, 2008.

KICKMEIER-RUST, Michael D. et al. **The ELEKTRA project: Towards a new learning experience**. M. Pohl, A. Holzinger, R. Motschnig, & C. Swertz: M3 – Interdisciplinary aspects on digital media & education, Vienna, p. 19-48, 2006.

KID KRYPTO. **Computer Science Unplugged**. 2007. Disponível em: <http://csunplugged.org/public-key-encryption>. Acesso em: 18 dez. 2012.

KIRKLEY, Sonny E.; TOMBLIN, Steve e KIRKLEY, Jamie. **Instructional Design Authoring Support for the Development of Serious Games and Mixed Reality Training**. Interservice/Industry Training, Simulation, and Education Conference, p. 1-11, 2005.

KIRKPATRICK, Donald L. e KIRKPATRICK, James D. **Evaluating training programs: the four levels**. 3ed., Berrett-Koehler Publishers, 2006.

- KIRRIEMUIR, John. **The relevance of video games and gaming consoles to the higher and further education learning experience.** Techwatch report TSW, 2003. Disponível em: [www.jisc.ac.uk/uploaded.../tsw\\_02-01.rtf](http://www.jisc.ac.uk/uploaded.../tsw_02-01.rtf). Acesso em: 26 jan.2012.
- KIRRIEMUIR, John; MCFARLANE, Ceangal A. **Literature Review in Games and Learning Literature Review in Games and Learning.** Bristol: FutureLab Series, 2004.
- KITCHENHAM, Barbara. **Procedures for performing systematic reviews.** Tech. Report TR/SE-0401, Keel University, Keel, UK. 2004.
- KOLB, David A. **Experiential learning: experience as the source of learning and development.** Prentice Hall, Englewood Cliffs, 1984.
- KOHWALTER, Troy C.; CLUA, Esteban W.G. e MURTA, Leonardo G.P. **SDM - An educational game for Software Engineering.** Proceeding of the Brazilian Symposium on Games and Digital Entertainment, Salvador, Brazil, p. 222-231, 2011.
- KRATHWOHL, David R.; BLOOM, Benjamin S.; MASIA, Bertram B. **Taxonomy of Educational Objectives, the Classification of Educational Goals.** Handbook II: Affective Domain. Nova York, EUA, 1973.
- KREMERS, Rudolf. **Level Design: Concept, theory, and practice.** A. K. Peters, EUA, 2009.
- KRUCHTEN, Philippee KING, James. **Mission to Mars: an agile release planning game.** Proceeding of the 24th Conference on Software Engineering Education and Training, Honolulu, HI, 2011.
- KUK, Kristijan et al. **Use game based interactive multimedia modules to learning basic concepts on courses for computing science.** Electrical Review, p. 150-153, 2012.
- LARMAN, Craig. **Agile and Iterative Development: A Manager's Guide.** Addison-Wesley Professional, 2003.

LABURÚ, C. E. e CARVALHO, M. **Educação científica: Controvérsias construtivistas e pluralismo metodológico.** Eduel, editora da Universidade Estadual de Londrina, Londrina, PR, BR, 2013.

LÉGER, Pierre-Majorique et al. **Business simulation training in information technology education: guidelines for new approaches in IT training.** Journal of Information Technology Education, 10, p. 39-53, 2011.

LEONG, Ben; KOH, Zi H. e RAZEEN, Ali. **Teaching introductory programming as an online game.** Department of Computer Science, National University of Singapore, p. 2-7, 2011.

LEVERINGTON, Michale; YÜKSEL, Murate ROBINSON, Michale. **Using role-play for an upper level CS course.** Journal of Computing Sciences in Colleges, v. 24, n. 4, p. 259-266, 2009.

LI, Frederick W.B.; WATSON, Christopher. **Game-based concept visualization for learning programming.** Proceedings of the 3rd Int. Workshop on Multimedia Technologies for Distance Learning. Scottsdale, AZ, 2011.

LIGHT-BOT, 2010. Disponível em:  
<http://armorgames.com/play/6061/light-bot-20>. Acesso em: 18 dez. 2012.

LINEK, Stephanie B. et al. **When playing meets learning: methodological framework for designing educational games.** Web Information Systems and Technologies, v. 45, p. 73-85, 2010.

LOH, Christian S. **Researching and Developing Serious Games as Interactive Learning Instructions.** International Journal of Gaming and Computer-Mediated Simulations, p. 1-19, 2009.

LONGSTREET, C. Shaun e COOPER, Kendra. **Using games in Software Engineering education to increase student.** Proceeding of the 24th Conference on Software Engineering Education and Training, Honolulu, HI, 2011.

LUDI, Stephanie. **Software Engineering Unplugged**. Department of Software Engineering Rochester Institute of Technology, Rochester, Nova York, EUA, 2010.

MAGNO, Eduardo et al. **SimulES: A game for teaching Software Engineering**. Project Thesis, Bachelor Course on Computer Science, PUC, Rio de Janeiro, Brazil, 2006.

MARCONI, Marina de A. e LAKATOS, Eva M. **Fundamentos de metodologia científica**. 5 ed., São Paulo: Atlas, 311 p., 2003.

MARCOS, Adérito e ZAGALO, Nelson. **Instantiating the creation process in digital art for serious games design**. Entertainment Computing, v. 2, n. 2, p. 143-148, 2011.

MARTIN, Andrew. **The design and evolution of a simulation/game for teaching information systems development**. Simulation e Gaming, v. 31, n. 4, p. 445-463, 2000.

MARZANO, Robert J. **The Art and Science of Teaching: A Comprehensive Framework for Effective Instruction**. 1 ed., Association for Supervision and Curriculum Development, Virginia, EUA, 2007.

MASCARENHAS, André O. **Gestão Estratégica de Pessoas**. São Paulo: Cengage Learning, 2008.

MATOOK, Sabine e INDULSKA, Marta. **Improving the quality of process reference models: A quality function deployment-based approach**. Decis. Support Syst. v. 47, n. 1, p. 60-71, 2009.

MCCLELLAND, David C. **Testing for competence rather than for intelligence**. American Psychologist, 1973.

MELERO, Javier; HERNÁNDEZ-LEO, Davinia e BLAT, Josep. **Considerations for the design of mini-games integrating hints for puzzle solving ICT-related concepts**. Proceeding of the 12th International Conference on Advanced Learning Technologies, Rome, Italy, 2012.



MERRIL, M. David et al. **Reclaiming instructional design.** Educational Technology, v. 36, n. 5, p. 5-7, 1996.

MICHAEL, David e CHEN, Sande. **Serious games: games that educate, train, and inform.** Thomson Course Technology, Boston, EUA, 2006.

MINISTÉRIO DA EDUCAÇÃO. **Diretrizes Curriculares Nacionais para Cursos de Graduação em Computação.** Parecer CNE/CNS 136/2012. Disponível em: <<http://portal.mec.gov.br>>. Acesso em: 13 maio. 2015.

MITCHELL, Alice; SAVILL-SMITH, Carol. **The use of computer and video games for learning: a review of the literature.** Learning and Skills Development Agency, Ultralab, London, UK, 2004.

MITTERMEIR, Roland et al. **AMEISE – A media education initiative for Software Engineering.** Proceeding of the Interactive Computer Aided Learning, International Workshop, Villach, Austria, 2003.

MOLENDÁ, Michael. **In Search of the Elusive ADDIE Model.** Performance improvement, v. 42, n. 5, p. 34-37, 2003.

MONSALVE, Elizabeth S., WERNECK, Vera M.B. e LEITE, Julio C.S. do P. **Teaching software engineering with SimulES-W.** Proceeding of the 24th Conference on Software Engineering Education and Training, Honolulu, HI, 2011.

MORSI, Rasha e RICHARDS, Chad. **BINX: an XNA/XBox 360 educational game for electrical and computer engineers.** Consumer Electronics Times, v. 1, n. 3, p. 33-42, 2012.

MORRISON, Briana B. e PRESTON, Jon A. **Engagement: gaming throughout the curriculum.** Proceedings of the 40th technical symposium on computer science education. ACM, Nova York, NY, EUA, p. 342-346, 2009.

MYPLANNET. **The Cisco Learning Network.** 2010. Disponível em: <https://learningnetwork.cisco.com/docs/DOC-7635>. Acesso em: 18 dez. 2012.

NATVIG, Lasse; LINE, Steinar e DJUPDAL, Asbjom. **Age of computers: an innovative combination of history and computer game elements for teaching computer fundamentals.** 34th Annual Conference on Frontiers in Education, Savannah, GA, 2004.

NATVIG, Lasse e LINE, Steinar. **Age of Computers – game-based teaching of computer fundamentals.** Proceeding of the 9th Annual Conference on Innovation and Technology in Computer Science Education, Leeds, UK, 2004.

NAVARRO, Emily O. **SimSE: a software engineering simulation environment for software process education.** Doctoral Dissertation, University of California, Irvine, CA, 2006.

NAVARRO, Emily O. e HOEK, André van der. **Comprehensive evaluation of an educational software engineering simulation environment.** Proceeding of the 20th Conference on Software Engineering Education e Training, Dublin, Ireland, 2007.

NAVARRO, Emily O. e HOEK, André van der. **SimSE: an interactive simulation game for software engineering education.** Proceeding of the 7th Int. Conference on Computers and Advanced Technology in Education, Kauai, HI, 2004.

NBR ISO/IEC 9126. **Engenharia de Software – Qualidade de Produto: Modelo de Qualidade.** ABNT – Associação Brasileiras de Normas Técnicas, Rio de Janeiro, RJ, 2003.

NERBRATEN, Oyvind e ROSTAD, Lilian. **hACME game: A Tool for Teaching Software Security.** Proceeding of the Int. Conference on Availability, Reliability and Security, Regensburg, Germany, 2009.

NET TERRARIUM. **NET Terrarium 2.0 project!** 2008. Disponível em: <http://terrarium2.codeplex.com/>. Acesso em 18 dez. 2012.

OLIVEIRA, Otávio J. (Org.) **Gestão da Qualidade: Tópicos Avançados.** Cengage Learning Nacional, 2004.

OLIVEIRA, Luiza M. B. **Cartilha do Censo 2010 – Pessoas com Deficiência.** Secretaria de Direitos Humanos da Presidência da República (SDH/PR) / Secretaria Nacional de Promoção dos Direitos da

Pessoa com Deficiência (SNPD) / Coordenação-Geral do Sistema de Informações sobre a Pessoa com Deficiência, Brasília, Brasil, 2012.

OMERINCK, Matthew. **Creating the art of the game**. New Riders, California, EUA, 2004.

ORANGE GAME, 2002. **Computer Science Unplugged: The orange game: routing and deadlock in networks**. Disponível em: <http://csunplugged.org/routing-and-deadlock>. Acesso em: 15 dez. 2012.

OREHOVACKI, Tihomir e BABIC, Snjezana. **Evaluating the quality of games designed for learning programming by students with different educational background: An empirical study**. Information and Communication Technology, Electronics and Microelectronics (MIPRO), p.963-968, 2015.

PAPER TOWER COMPETITION, 2009. Disponível em: <http://www.gqs.ufsc.br/paper-tower-competition/>. Acesso em: 19 dez. 2012.

PARSONS, Paul. **Preparing Computer Science Graduates for the 21st Century**. Teaching Innovation Projects, v. 1, n. 1, 2011.

PEIXOTO, Daniela C. C. et al. **An overview of the main design characteristics of simulation games in Software Engineering education**. Proceeding of the 24th Conference on Software Engineering Education and Training, Pittsburgh, PA, 2011.

PETERSEN, Kaiet al. **Systematic Mapping Studies in Software Engineering**. Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering. British Computer Society, Swinton, UK, p. 68-77, 2008.

PM MASTER, 2012. Disponível em: <http://www.gqs.ufsc.br/pm-master/>. Acesso em: 19 dez. 2012.

POPESCU, Maria et al. **Serious games in education: linking pedagogy and game design**. Proc. of 5th European Conference on Games Based Learning, Athens, Greece, 2011.

PORTER, Adam A.; VOTTA, Lawrence G., Jr. e BASILI, Victor R. **Comparing detection methods for software requirements inspections: a replicated experiment.** Software Engineering, IEEE Transactions, v.21, n.6, p. 563-575, 1995.

Poulova, Petra e Klimova, Blanka. **Education in Computational Sciences. International Conference on Computational Science.** Procedia Computer Science, v. 51, 2015, 1996–2005 p.

PPGCC. **Programa de Pós-Graduação em Ciência da Computação. Linhas de Pesquisa.** Disponível em: <  
<http://ppgcc.posgrad.ufsc.br/linhas-de-pesquisa-2/>>. Acesso em: 06Set. 2016.

PRATT, Mary K. **10 hot IT skills for 2013: Want to snag a pay premium?** Check out the IT skills that will be in high demand in 2013. 2012. Compute world. Disponível em:  
[http://www.computerworld.com/s/article/9231486/10\\_](http://www.computerworld.com/s/article/9231486/10_). Acesso em: 24 jan. 2012.

PRENSKY, Marc. **Digital game-based learning.** Paragon House, 464 p., 2007.

PRESSMAN, Roger S. **Engenharia de Software.** 7ª. Ed. São Paulo: McGraw-Hill, 2009.

PROJECT RISK, 2001. Disponível em:  
<https://www.successfulprojects.com/Store/ProjectRiskBoardGame/tabid/68/Default.aspx>. Acesso em: 19 dez. 2012.

PROJECT-O-POLY (POP), 2007. Disponível em:  
<http://www.semq.eu/leng/proimplog1.htm>. Acesso em: 19 dez. 2012.

QIAN, Meihua e CLARK, Karen R. **Game-based Learning and 21st century skills: A review of recent research.** Journal Computers in Human Behavior, v. 63, p. 50-58, 2016.

RITTGEN, Peter. **Quality and perceived usefulness of process models.** Proceedings of the ACM Symposium on Applied Computing. ACM, Nova York, EUA, p. 65-72, 2010.

ROBOT TROUBLE. **Djco-robot-trouble: XNA serious game teaching procedural programming**, 2011. Disponível em: <https://code.google.com/p/djco-robot-trouble/>. Acesso em: 20 jan. 2012.

ROBOZZLE, 2011. **RoboZZle a social puzzle game**. Disponível em: <http://robozzle.com/>. Acesso em: 20 jan. 2012.

ROSSI, Simo. **Moving Towards Modeling Oriented Software Process Engineering: A Shift from Descriptive to Prescriptive Process Modeling**. Proceedings of Product Focused Improvement of Embedded Software Processes, p. 508- 522, 1999.

ROSSIOU, Eleni e PAPADAKIS, Spyros. **Educational games in higher education: a case study in teaching**. Proceeding of the Int. Conference on Education in a Changing Environment, Salford, UK, 2007.

RUAS, Roberto. **Gestão por Competências – Uma contribuição à estratégia das organizações**. In: RUAS, R. et al. Os novos horizontes da gestão – Aprendizagem organizacional e competências. Porto Alegre: Bookman, 2005.

RYOO, Jungwoo et al. **Game-based InfoSec Education using OpenSim**. Proceeding of the 15th Colloquium for Information Systems Security Education. Fairborn, OH, 2011.

SALEN, Katie e ZIMMERMAN, Eric. **Rules of Play: game design fundamentals**. Cambridge, MIT Press, 2004.

SASKATCHEWAN EDUCATION. **Instructional approaches: a framework for professional practice table of contents**. 1991.

SAUNDERS, Mark; LEWIS, Philip e THORNHILL, Adrian. **Research Methods for Business Students**. 5 ed. Prentice Hall, 635 p., 2009.

SAVI, Rafael; WANGENHEIM Christiane G. von e BORGATTO, André. **Um Modelo de Avaliação de Jogos Educacionais na Engenharia de Software**. 25th Brazilian Symposium on Software Engineering /São Paulo, Brasil, 2011.

SAWYER, Ben e SMITH, Peter. **Serious games taxonomy**. 2008. Disponível em: [www.seriousgames.org/presentations/serious-games-taxonomy-2008\\_web.pdf](http://www.seriousgames.org/presentations/serious-games-taxonomy-2008_web.pdf). Acesso: 15 fev. 2012.

SCHELL, Jesse. **The art of game design: A book of lenses**. 1ª ed. Morgan Kaufmann: CRC Press, 489 p., 2008.

SCHIFFLER, Andreas. **A heuristic taxonomy of computer games**. Ferzkopp's Philosophy Work, 1–13, 2006.

SCHNEIDER, Marcelo F. **SCRUM'ed: um jogo de RPG para ensinar Scrum**. 66 f. TCC (Graduação) - Curso de Sistemas de Informação, Departamento de Departamento de Informática e Estatística, Universidade Federal de Santa Catarina, Florianópolis, 2015.

SCHWABERE, Ken e BEEDLE, Mike. **Agile Software Development with Scrum**. 1ª ed. Prentice Hall PTR, Upper Saddle River, EUA, 2001.

SCHWABER, Ken e SUTHERLAND, Jeff. **SCRUM guide, 2013**. Disponível em: <https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-Portuguese-BR.pdf>. Acesso em: 15 nov. 2015.

SCRUM LEGO CITY, 2009. Disponível em: <http://www.agile42.com/en/training/scrum-lego-city/>. Acesso: 15 fev. 2012.

SCRUMIA. **Welcome to SCRUMIA**. 2009. Disponível em: <http://www.gqs.ufsc.br/applying-scrum-welcome-to-s>. Acesso em: 20 jan. 2012.

SECURE VOLUNTEER GAME. **The Cisco Learning Network**. 2010. Disponível em: <https://learningnetwork.cisco.com/docs/DOC-7559>. Acesso em: 19 dez. 2012.

SEMESP. **Por que estudantes largam o curso de computação?** Sindicato das Entidades Mantenedoras de Estabelecimentos de Ensino Superior no Estado de São Paulo, 2013. Disponível em: [http://semesp1.tempsite.ws/semesp\\_beta/por-que-estudantes-largam-o-curso-de-computacao/](http://semesp1.tempsite.ws/semesp_beta/por-que-estudantes-largam-o-curso-de-computacao/). Acesso em: 13 mai. 2015.

SHARKWORLD. **A project management game**. 2008. Disponível em: <http://www.sharkworldgame.com/>. Acesso em: 19 dez. 2012.

SHAW, Katherine e DERMOUDY, Julian. **Engendering an empathy for Software Engineering**. Proceedings of the Australasian Conference on Computing Education, Darlinghurst, Australia, 2005.

SHENG, S. et al. **Anti-Phishing Phil: the design and evaluation of a game that teaches people not to fall for phish**. Proceeding of the 3rd Symposium On Usable Privacy and Security, Pittsburgh, PA, 2007.

SILVEIRA NETO, Paulo A. da M. **A systematic mapping study of software product lines testing**. Information and Software Technology v. 53, p. 407-423, 2011.

SIMPSON, Elizabeth J. **The classification of educational objectives, psychomotor domain**. Washington, Gryphon House, 1972.

SIMULATE COMPUTER GAME. **Children and technology**. 2008. Disponível em: <https://sites.google.com/site/childrenandtechnology/presentation-3-simulate-computer>. Acesso em: 19 dez. 2012.

SINDRE, Guttorm; NATVIG, Lasse e JAHRE, Magnus. **Experimental validation of the learning effect for a pedagogical game on computer fundamentals**. IEEE Transactions on Education, v. 52, n. 1, p. 10-18, 2009.

SINGH, Jaspaljeet; DORAIRAJ, Siva K. e WOODS, Peter. **Learning computer programming using a board game – case study on C-Jump**. Proceeding of the International Symposium on Information and Communications Technologies, Kuala Lumpur, Malaysia, 2007.

SOARES, Gustavo M. e RAUSIS, Bruno Z. **Development of an educational game for teaching project management in undergraduate computing programs**. Project Thesis, Bachelor Course on Information Systems, Federal University of Santa Catarina, Brazil, 2011.

SOMMEREGGER, Paul e KELLNER, Gudrun. **Brief Guidelines for Educational Adventure Games Creation (EAGC)**. Digital Game and Intelligent Toy Enhanced Learning, IEEE Fourth International Conference, Takamatsu, Japão, p.120-122, 2012.

SOMMERVILLE, Ian. **Software Engineering**. Pearson Education, v. 8. 2007.

SORTING ALGORITHMS. **Digital Game Based Learning**. 2012.Disponível em: <http://sortingalgorithmgame.webs.com/>. Acesso em: 19 dez. 2012.

SRINIVASAN, Jayakanth e LUNDQVIST, Kristina. **A Constructivist approach to Teaching Software Processes**. Proceeding of the 29th International Conference on Software Engineering, Minneapolis, MI, 2007.

SRINIVASAN, Vinoud; BUTLER-PURRY, Karen e PEDERSEN, Susan. **Using video games to enhance learning in digital systems**. Proceeding of the Conference on Future Play, Toronto, Canada, 2008.

STAALDUINEN, Jan-Paul van e FREITAS, Sara de. **A Game-Based Learning Framework: Linking Game Design and Learning Outcomes**. Nova York: Peter Lang, p. 29-45, 2011.

SUBNET GAME. **The Cisco Learning Network**. 2008.Disponível em:<https://learningnetwork.cisco.com/docs/DOC-1802>. Acesso em: 19 dez. 2012.

SUBNET TROUBLESHOOTING GAME.**The Cisco Learning Game**. 2010.Disponível em: <https://learningnetwork.cisco.com/docs/DOC-7558>. Acesso em: 19 dez. 2012.

SUSI, Tarja; JOHANNESSON, Mikael e BACKLUND, Per. **Serious Games – an overview**. Technical Report HS-IKI-TR-07-001, University of Skövde, Sweden, 2007.

SWEBOK. **Guide to the Software Engineering Body of Knowledge**.2004 Version. Disponível em: <http://www.swebok.org>. Acesso em: 15 nov. 2015.



SYBERFELDT, Anna e SYBERFELDT, Sanny. **A serious game for understanding artificial intelligence in production optimization.** Proceeding of the IEEE Conference on Computational Intelligence and Games, Copenhagen, Denmark, 2010.

TAN, Binge SENG, Jet L.K. **Game-based learning for Data Structures: a case study.** 2nd International Conference on Computer Engineering and Technology, Chengdu, China, 2010.

TANG, Ying; SHETTY, Sachin e CHEN, Xiufang. **Educational effectiveness of virtual reality games promoting metacognition and problem-solving.** 119th Annual Conference and Exposition of the American Society for Engineering Education, San Antonio, Texas, EUA, 2012.

TARAN, Gil. **Using games in Software Engineering education to teach Risk Management.** Proceeding of the 20th Conference on Software Engineering Education e Training, Dublin, Irlanda, p. 211-220, 2007.

THIRY, Marcello et al. **Uma Abordagem para a Modelagem Colaborativa de Processos de Software em Micro e Pequenas Empresas.** SBQS - Simpósio Brasileiro de Qualidade de Software, Vitória, 2006.

THIRY, Marcello; ZOUCCAS, Alessandra e SILVA, Antônio C. da. **Empirical study upon software testing learning with support from educational game.** Proceeding of the 23rd International Conference on Software Engineering and Knowledge Engineering, Miami Beach, EUA, 2011.

THOMPSON, Jim; BARNABY-GREEN, Barnaby e CUS-WORTH, Nic. **Game Design: Principles practice, and techniques- The ultimate guide for the aspiring game designer.** 1ª ed. Nova Jersey: Wiley, 192 p., 2007.

TREASURE HUNT. 2011. Disponível em: <https://www.nonamesite.com/web/cs-stem/treasure-hunter>. Acesso em: 20 jan. 2012.

VOGEL, Jennifer J. et al. **Computer gaming and interactive simulations for learning: A meta-analysis.** Journal of Educational Computing Research, v. 34, n. 3, p. 229-243, 2006.

WANGENHEIM, Christiane G. von et al. **Creating Software Process Capability/ Maturity Models.** IEEE Software, v. 27, n. 4, 2010.

WANGENHEIM, Christiane G. von e SHULL, Forrest. **To game or not to game?** IEEE Software, v. 26, n. 2, p. 92-94, 2009.

WANGENHEIM, Christiane G. von; THIRY, Marcello e KOCHANSKI, Djone. **Empirical evaluation of an educational game on software measurement.** Empirical Software Engineering, v. 14, n. 4, p. 418-452, 2009.

WANGENHEIM, Christiane G. von; SAVI, Rafael e BORGATTO, Adriano F. **DELIVER! – An educational game for teaching Earned Value Management in computing courses.** Information and Software Technology, v. 54, n. 3, p. 286-298, 2012.

WANGENHEIM, Christiane G. von e WANGENHEIM, Aldo von. **Ensinando Computação com Jogos.** Bookes, 2013.

WATERFALL GAME. **The Waterfall Game: games for Software Testing and QA.** 2009. Disponível em: <http://www.the-software-tester.com/games.html>. Acesso em: 19 dez. 2012.

WEB POSTER WIZARD. **7 Helpful Instructional Strategies: Summary and Reflections.** 2009. Disponível em: [http://poster.4teachers.org/view/poster.php?poster\\_id=424539](http://poster.4teachers.org/view/poster.php?poster_id=424539). Acesso em: 13 set. 2016.

WEIN, Joel et al. **Virtualized games for teaching about distributed systems.** Proceeding of the ACM Technical Symposium on Computer Science Education, ACM Nova York, EUA, p. 246-250, 2009.

WIRELESS EXPLORER. **The Cisco Learning Network.** 2010. Disponível em: <https://learningnetwork.cisco.com/docs/DOC-7560>. Acesso em: 19 dez. 2012.

WOLF, Mark J. **The medium of the video game.** Austin: University of Texas Press, p. 113-134, 2001.

WONG, Wai-Tak e CHOU, Yu-Min. **An interactive bomberman game-based teaching /learning tool for introductory C programming.** Technologies for E-Learning and Digital Entertainment, Lecture Notes in Computer Science, Springer-Verlag Berlin, p. 433-444, 2007.

YE, En; CHANG, Liu e POLACK-WAHL, Jennifer A. **Enhancing software engineering education using teaching aids in 3-D online virtual worlds.** Proceeding of 37th Frontiers in Education Conference Milwaukee, WI, 2007.




YIN, Robert K. **Applications of case study research.** Applied social research methods series, v. 34, Sage Publications, 231 p., 2012.

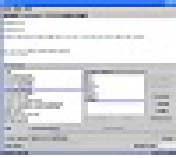


ZAPATA, Carlos M. e AWAD-AUBAD, Gabriel. **Requirements Game: teaching software project management.** CLEI Electronic Journal, v. 10, n. 1, 2007.






ZAPATA, Carlos M. **Teaching software development by means of a classroom game: The software development game.** Developments in Business Simulation and Experiential Learning, 36, p. 156-165, 2009.






ZWIKAEL, Ofer e GONEN, Ammon. **Project execution game (PEG): training towards managing unexpected events.** Journal of European Industrial Training, v. 31, n. 6, p. 495-512, 2007.

## APÊNDICE A –Jogos encontrados na revisão sistemática da literatura.

No.	Name	Computing Knowledge Area	Screenshot/Photo	Gameplay	Reference(s)
1	3DAR Lego Game	Human Computer Interaction		<p>Players have to re-construct a 3D model using its decomposed pieces. The game provides 3 game levels with increasing difficulty. At level 1, all pieces are precise components of the model, at level 2 each piece is a primitive geometry (cube, cone) that has to be customized and at level 3 no pieces are provided and the players have to create the model from primitive geometries.</p>	(Do e Lee,2009)
2	Age of Computers	Computer Science		<p>Player travel through computer history from early mechanical computers, transistor age to embedded systems of the present age. They have to solve different kind of problems (multiple-choice, numbering, ALU control signals) to earn points. The availability of content in the game depends on the player's progress. A chat window for each historical period is used for communication with other students and teaching assistants.</p>	<p>(Sindre et al. 2009) (Natvig e Line 2004) (Natvig et al. 2004)</p>
3	Algorithms Recursive Game	Algorithms and Complexity		<p>During the game, the players have to answer questions on different levels of difficulty covering various aspects of recursive algorithms. Players advance their tokens by rolling a dice. If a player arrives at the bottom of a ladder or at the head of a snake, a question is displayed. If the player answers correctly, her/his token moves up to the top of the ladder or stays at the head of the snake.</p>	(Rossiou e Papadakis, 2007)




4	AMEISE	Software Engineering		<p>AMEISE is based on SESAM. In the game, students assume the role of a technical project manager. They can hire and fire personnel, structure the project, and allocate tasks. Students are challenged to manage a project according to a particular model of the problem structure, selected by the instructor. It is up to the instructor to select the number of trials (simulation runs) to solve given tasks within specified constraints. Students can learn from previous simulation runs, change their strategies and measure their own success using the AMEISE self-assessment feature.</p>	(Mittermeir et al., 2003)
5	Anti-Phishing Phil	Security		<p>The main character of the game is Phil, a young fish living in the Interweb Bay. Phil wants to eat worms so he can grow up to be a big fish, but has to be careful of phishers that try to trick him with fake worms (representing phishing attacks). Each worm is associated with a URL, and Phil's job is to eat all the real worms (which have URLs of legitimate web sites) and reject all the bait (which have phishing URLs) before running out of time. The other character is Phil's father, who is an experienced fish in the sea. He occasionally helps Phil out by giving Phil some tips on how to identify bad worms (and hence, phishing web sites). The game is split into four rounds, each one more difficult than the previous and focuses on a different type of deceptive URLs. Players have to correctly recognize at least six out of eight URLs within two minutes to move on to the next round. If a player loses all three lives, the game is over.</p>	(Sheng et al., 2007)
6	Battleships	Algorithms and Complexity		<p>Using sheets with sequences of battleships, each player circles one of the ships. Then, the players have to guess where his/her partner's ship is. There are several variations of the game for different search algorithms (binary, hashing)</p>	(Battleships, 2002)





7	BattleThreats	Operating Systems		<p>The class is broken down into one controller and two teams. The players are responsible for the placement of one ship each and for firing a shot from that ship each turn until their own ship is destroyed (or their side wins by sinking all of their opponents' ships). The controller gets the enemy team's ship layout and announces the effect (hit or miss) of each shot. At the end of a turn, the controller compares battle damage and reports the results. No other means of communications are allowed.</p>	(Hill et al., 2003)
8	Binary Game	Net Centric		<p>The game requires players to either set bits in a byte correctly to equal a given number or to insert the number represented by a given set of bits.</p>	(Binary Game, 2008)
9	BINX	Digital Logic		<p>The game takes place in the context of a computer being attacked by a virus. The main character is Chip, an integrated circuit designed with the purpose to wipe all traces of the malicious virus from the computer. Inside the computer, information is flowing from the motherboard to all output devices attached to the computer except the monitor. The adventure takes place inside the CPU. Players must navigate a path through the bus to find the viral infection plaguing the graphics processor and resolving several missions on different levels.</p>	(Morsi and Richards, 2012)
10	Bomberman Game	Programming fundamentals		<p>In this game, players have to write C code to control the movement of the Bomberman.</p>	(Wong e Chou, 2007)
11	BOTS	Programming Fundamentals		<p>Bots is a game that contains a series of programming challenges that students must solve.</p>	(Hicks, 2010)





12	CARGO-BOT	Programming fundamentals		Players must program a crane to properly situate boxes using loops and conditional variables. It rewards players for discovering the most efficient way.	(CargoBot, 2012)
13	CEEBOT	Programming fundamentals		In CeeBot-A (for teenager and adults) the player has to program robots in outside environments on various planets. This version, deals, mainly with repetitions, conditional options, variables, arrays, functions and classes.	(CeeBot, 2008)
14	C-Jump	Programming Fundamentals		The goal of the game is to find the most efficient way to “ski” down a mountain. Therefore, players must make decisions based on common programming syntax to go down a certain path. By moving around the board, entering loops, branching under conditional and switch statements, the players gain physical experience of a complete program. First player to move all skiers past the finish line is the winner.	(Singh et al., 2007)
15	COLOBOT	Programming fundamentals		The player heads a space expedition and is assisted only by some robots. Her/his mission consists in successive attempts at the exploration and colonization of various planets. During the game, the player has to build and program new types of robots in order to complete the mission.	(Colobot, 2001)
16	Computer Architecture Mini-Game	Computer Architecture and Organization		The player has to solve several problems on different levels of difficulty on logic gates and elements of computer architecture. The game starts with an initial score (1000 points) and each time a player makes a wrong decision the score is decreased. Textual hints are provided.	(Melero et al., 2012)





17	Computer Theory Jeopardy Game	Theory of Programming Languages		Similar to the Jeopardy game, players respond questions to win the game. Here, the questions are related to context free grammar and pushdown automata.	(Hill et al., 2003)
18	Computing Networks Mini-Game	Net Centric		The game encourages players to solve three different situations (one per level): how to connect different elements to have ADSL in home, how the information travels through the network, and how information is sent from a personal computer to another. Routers, IPs, Computer ports, browsers or frames are examples of puzzle pieces in the game. Correct actions lead to an increase of the player's score, which is reduced when making mistakes.	(Melero et al., 2012)
19	Conquer the Net	Net Centric		This risk-game based game takes place at a scenario composed of a number of PCs, switches, and routers, placed on a map. At the start of the game, a PC and an objective are assigned to each player. The game performs in turns. At each turn, a player is allowed to modify the configuration of any of the devices displayed on the board (e.g., change its IP address, mask, and gateway). The number of modifications a player may perform is determined by rolling a dice. The winner is the player who achieves his/her objective first.	(Arevalillo-Herráez et al., 2009)
20	Control-Alt-Hack	Net Centric		The players act as white hat hackers in a security consulting company. Each player is given a Hacker card. Gameplay is centered on missions—a variety of audit jobs and pro bono work that require the selective application of hacker skills: Hardware Hacking, Network Ninja, Cryptanalysis, Forensics. The character's skill levels and player's dice rolls determine whether the player succeeds or fails at a mission. Players can increase their skill levels by purchasing useful items (Bag of Tricks); opponents can hinder player's efforts to complete a mission by playing Lightning Strikes on them. Mission successes and failures lead to the gain and loss of Hacker Cred. Players win the game by accruing enough Hacker Cred and becoming the CEO of their own consulting company.	(Denning et al., 2012)











21	CounterMeasures	Security		<p>The player is guided through several missions, each teaching a new aspect of security. Each mission has a title, a description, a score for completing the mission, a skill as the focus of the mission, objectives required to complete the mission, help given to guide the user, and a list of commands learned during the mission. The missions build upon each other, allowing the player to utilize previously learned skills in each new mission. Players are given a fully functional shell that runs commands while working on a mission.</p>	(Jordan et al., 2011)
22	CyberCIEGE	Net Centric		<p>Players of this video game purchase and configure workstations, servers, operating systems, applications and network devices. They make trade-offs as they struggle to maintain a balance between budget, productivity, and security. In its longer scenarios, users advance through a series of stages and must protect increasingly valuable corporate assets against escalating attacks.</p>	(CyberCIEGE, 2011)
23	Database concurrency control card game	Information Management		<p>In this game, students play cards and build a schedule for a given set of simultaneous transactions. They use their knowledge about transactions and concurrency control protocols to simulate the work of a transaction processing system. The sequence of activities is simulated through a control card or kanban that is passed from student to student. The card exchange is similar to a procedure call. Each student has to decide what to do at her/his turn, based on the concepts s/he learned and the concurrency control protocol being used.</p>	(Kern et al., 2008)
24	Databases: Lots of Data and Getting Quick Results	Information Management	N/I	<p>Players execute database operations by representing different roles, such as, user, timekeeper, DBMS, or data. The Data Record players stand in line and, then, simulating a query the DBMS goes to each record to check, if they are in the query. If yes, they move to the designated results area. The timekeeper states the time to conduct the search.</p>	(Ludi, 2010)





25	Dealing with difficult people	Software Engineering		<p>During this game, groups of learners realize a project kick-off meeting. One of the group members represents the project manager, who moderates the meeting and has to make sure that in the end all members confirm their commitment. The problem is that each of the other group members is assigned a role of a difficult person being instructed to “act”, e.g., as a whiner, no-person. Once the project manager reacts in an adequate way, the person turns into a cooperative member and gives her commitment to the project.</p>	(Carvalho, 2012)
26	Deliver!	Software Engineering		<p>DELIVER! is a board game to teach Earned Value Management for monitoring &amp; controlling a software project. Once the game starts, player pairs advance in the project execution rolling a dice - in addition risk events may happen. Each completed round all player pairs have to pay the weekly salary for their project teams. When finishing a project phase, the player pairs have to monitor the project progress calculating Earned Value indicators and may change the project plan (e.g., by firing or contracting new team members). Winner of the game it the player pair, who first delivers the product to the customer arriving at the field product delivery without running out of financial resources during execution.</p>	(Wangenheim et al., 2012)
27	DesigMPS	Software Engineering		<p>The player assumes the role of a process engineer and has to model a process based on given process descriptions (aligned to the Brazilian software process improvement model MPS.BR). The game offers 4 levels with increasing degree of difficulty in terms of given elements. The created process models are compared to pre-defined solutions and, based on their degree of similarity, a score is assigned. Goal of the game is to achieve maximum score.</p>	(Chaves et al., 2010) (Chaves et al., 2011)
28	Detective Game – what killed the project?	Software Engineering		<p>The game takes place in the context of a fictitious company that completed a software project for the development of an online pizza web site. The project failed and now the company is contracting the players as consultants in order to identify what went wrong. Therefore, the players receive a set of project documents. The players have to analyze the project documentation and track project progress by revising the weekly status reports applying Earned Value Management. For each correct calculation</p>	(Soares e Rausis, 2011)





				and correctly identified time or cost overrun, the players receive a point. The winner is the group of players that obtained the largest number of points.	
29	Digi Island	Digital Logic		The game takes place on an exotic island, where 1s and 0s in digital circuits are represented through usable and unusable spaces on the island. The player is an adventurer on the Digi Island to be transformed into a tourist attraction by developing real estates, such as, amusement parks. The goal of the player is to construct a minimum number of buildings as large as possible covering all the usable spaces, while satisfying some further regulations.	(Harper et al., 2011)
30	Digital Logic and Electronics Concepts	Digital Logic		The game is designed in levels progressing from basic to advanced topics on digital logic. The levels are separated into three groups. The first group focuses primarily on binary information, the second group focuses on logic functions, and the third group focuses on circuits with memory.	(Belfore-II et al., 2009)
31	Digital System Game	Digital Logic		The game starts with the player in one corner of an imaginary 3D world similar to those found in first person-shooter games. The player's goal is to reach the exit, which can be accomplished by unlocking several doors and obtaining two skill upgrades. At each locked door, the player is presented with a sum-of-products combinational circuit problem. The game switches to a 2D environment for the digital circuit design problems. The game updates the external outputs automatically to indicate the values of the outputs of the current circuit for the specified input values.	(Srinivasan et al., 2008)
32	DSAsketch	Algorithms and Complexity		Players are divided into two teams. The main idea of the game is to draw concepts related to data structures and algorithms on the whiteboard while other team members are trying to guess the concept in the picture. The winner of the game is the team with the highest score once all players had a turn to draw.	(Hakulinen, 2012)

33	EleMental: The Recurrence	Programming Fundamentals		EleMental is a death match pitting players against one another in a winner-takes-all battle. Players can manipulate terrain, wind, water, and temperatures as they fight for supremacy against their opponents applying programming. 3D FPS style game that runs on both PC and Xbox 360.	(Chaffin et al., 2009)
34	ERPsim	Information Systems		Using a continuous-time simulation, students are put in a situation in which they have to run their business with a real-life ERP system. Students, thus operate a company; be it a bottled water distributor, a make-to-stock cereal manufacturer, or a dairy company, and must interact with suppliers and customers by sending and receiving orders, delivering their products and completing the whole cash-to-cash cycle.	(Léger et al., 2011)
35	Groupthink /Second Life	Software Engineering		Multi-player Second Life version of the groupthink exercise game focusing on requirements engineering. The objective of the game is to test the ability of a group of learners to reach consensus on software specifications. After discussing the specification within the group, players individually answer a set of questions on the specifications and the system evaluates the number of corresponding answers and presents performance statistics. Winner of the game is the group with the highest score at the end.	(Ye et al., 2007)
36	hACME Game	Security		The game is organized as a series of levels where the player must overcome a set of challenges in order to unlock access to the next level. Each level focuses on a set of well-known security vulnerabilities.	(Nerbraten e Rostad, 2009)


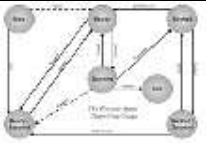

37	Immersive Security Education Environment (I-SEE)	Security		The learning module simulates a scenario in which two teams are competing against each other: one acting as a group of attackers and the other acting as a group of defenders. Each team has a base consisting of a packet assembly board, a building containing a money counter and a console, and a pipeline representing a network connection from the base to the gateway router. The game begins, when each team successfully creates a data packet according to the TCP/IP protocol on the packet assembly board. As soon as frames start moving, the other team can launch an attack, while the team has to defend the attacks.	(Ryoo et al., 2011)
38	INNOV8	Information Systems		The player interacts with other virtual employees, participating in their daily activities in a fictitious company. During the game, the player will be involved in three types of Business Process Modeling activities: process discovery and process modeling, collaboration-driven simulation and iterative process improvement and real-time business management.	(INNOV8, 2008)
39	IT Manager	Information Systems		Using a collection of technologies (cooling, security, mobility, multi-core machines) the player has to keep the servers running and the people working. The problems users face range from virus attacks or slow computers to broken disks or strange beeps. The problems the player has to resolve range from simultaneous support requests to spam in their inbox.	(IT Manager, 2007)
40	JV2M	Programming Fundamentals		The game takes the form of a 3D videogame in which the player must compete to provide the right machine instructions, collect resources needed by the instructions and use her/his knowledge about Java compilation to find the best strategy to win the game. This means that a player has to compile and execute byte code quicker than the opponent.	(Gómez-Martín et al., 2006)
41	Kid Krypto	Security	N/I	Players send secret messages using encryption to pass information on a treasure map.	(Kid Krypto, 2007)





42	Lego Factory	Intelligent Systems (AI)		<p>The simulation shows a physical production process in the form of a miniature factory for producing "bubble gums" using Lego bricks and Mindstorms NXT. As the factory production manager, the player's task is to find the best configuration (wrt. production flow and buffer sizes) possible for the production process in order to maximize profit. The player wins the game, if the optimal configuration of the factory is found. In order to find optimal solutions, players have to develop AI algorithms.</p>	(Syberfeldt e Syberfeldt, 2010)
43	Lego Tower Team Activity: Managing Change	Software Engineering	N/I	<p>Players are instructed to construct Lego towers. Winner of the game is the group that constructs the largest tower. During the construction, changes are requested (e.g., only one person can construct - other team members can only give instructions, tower has to be put on wheels to be mobile).</p>	(Ludi, 2010)
44	Light-Bot	Programming Fundamentals		<p>In this game, the player has to program a robot to light up blue tiles. Each time the objective is achieved, the player advances to the next level.</p>	(Light-Bot, 2010)
45	MIS Project Manager (Formerly known as the Information Systems Project Manager Game)	Software Engineering		<p>The player takes the role of project manager in an IS development project. Once selected a strategy for focus, the player works through the systems development life-cycle. In addition, s/he has to deal with events occurring during the project execution.</p>	(Martin, 2000)
46	Mission to mars - Release or iteration planning	Software Engineering		<p>The game is a Monopoly-style board game, where some factors such as uncertainty in estimation, actual velocity, and occurrence of defects are simulated by rolling a dice. Hard constraints and dependencies between stories are added to stimulate discussion on the strategy to pursue and how to mitigate risks.</p>	(Kruchten e King, 2011)






47	myPlanNet Simulation	Net Centric		During the game, the player represents the CEO of a service provider, who must manage the company as it evolves. The player has to connect citizens with the next-generation IP network and guide them into the Connected Life with the wonders of visual networking.	(MyPlanNet, 2010)
48	Next Generation_7 helicopter game	Software Engineering		The game is played for a minimum of three iterations by project teams that are each composed of sub-teams covering the functional areas of requirements, design, implementation, integration, and reserve funds. Each member of the project team receives the problem statement, the cockpit avionics architecture diagram and a detailed set of instructions about their own role. The first iteration of the game, for example, is set up as a classical waterfall model, wherein the different teams work in a sequential fashion with discrete handoffs between the phases.	(Srinivasan e Lundqvist, 2007)
49	.NET Terrarium 2.0	Programming fundamentals		Applying the .NET programming model and languages, the player can create herbivores, carnivores, or plants and then introduce them into a peer-to-peer, networked ecosystem where they compete for survival. Once the creature is loaded into Terrarium, it acts on the instructions supplied by its code.	(NET Terrarium, 2008)
50	Operating systems role plays	Operating Systems	N/I	In the game, students assume the roles of processes, while the instructor represents the processor. The game itself involves running sample sets of program code in a step-wise fashion so that students can see the actions and consequences of each segment of code.	(Leverington et al., 2009)
51	Paper Tower Competition	Software Engineering		During the game, students have to build a paper tower for a given set of requirements while executing a systematic project management process, including the application of Earned Value Management for monitoring & controlling the project. Winner of the game is the group that manages to build the tower and obtained the largest SPI and CPI with minor total project cost taking into consideration also the beauty of the design of the tower.	(Paper Tower Competition, 2009)



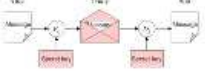


52	Path	Programming Fundamentals		<p>Paths string together a series of questions that students must complete in sequence so as to complete a path. They may either be multiple choice questions or more general coding questions where automated public and private test cases are run to verify the students' answers.</p>	(Leong et al., 2011)
53	PDCConsole	Distributed Systems		<p>The basis of the game is a barebones, but fully functional integrated website that combines features of social networking and video sharing. The site displays (mock) advertisements when different pages are viewed. The game generates artificial traffic against the site, and the overall system metric of health is the number of advertisements served and resulting revenue. The game is a sort of fire fighting exercise that begins when the game administrator breaks parts of the system in some way. Players notice that the system's performance has degraded because ads and revenue drop. Then, they have to identify the problem and fix it.</p>	(Wein et al., 2009)
54	PlayScrum	Software Engineering		<p>PlayScrum is derived from Problems and Developers. It is a card game in which each student plays the role of a Scrum Master in a software development project adopting SCRUM practices. The game is divided into sprints that differ from project to project and during which each player must develop a number of tasks. The winner is the player, who first performs all tasks without errors or who completed the highest percentage of tasks without errors at the end of the last iteration.</p>	(Fernandes e Sousa, 2010)
55	PM Master	Software Engineering		<p>Trivia-style board game with multiple-choice questions about Project Management on different knowledge areas, such as, scope, time and quality management in alignment with PMBOK (4.Ed.). The player, who first responds correctly one question of each of the nine knowledge areas, wins the game.</p>	(PM Master, 2012)











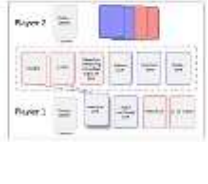
56	Problems and Programmers	Software Engineering		<p>A card game that simulates the software process from requirements specification to product delivery based on the waterfall life cycle. Players take the role of the project leader in the same project and compete to be the first to complete the project. They pass through the phases of the software process and draw cards and take actions to continue the development as well as to react to problems. The winner of the game is the player who first achieves a sufficient number of integrated code cards without bugs.</p>	<p>(Baker et al., 2003) (Baker et al., 2005)</p>
57	Process State Transition	Operating Systems		<p>Each group of players is given a game board representing the seven-state process transition model. One of the students is selected to be the operating system (OS), one the timekeeper (TK), and the others become programs, each keeping track of some number of processes as they are managed by the operating system. When a process moves into a state in which it must be present in memory (Ready, Running), the player who owns that process places its memory markers on a grid representing available memory. When the process is suspended, the memory markers are lifted, indicating that the process has been moved out of main memory.</p>	<p>(Hill et al., 2003)</p>
58	Programming Fundamentals Mini-Game	Programming fundamentals		<p>The player has to solve several problems on different levels of difficulty. The game provides the player with a skeleton of the code where s/he then has to place the different code pieces. The game starts with an initial score (1000 points) and each time a player makes a wrong decision the score is decreased. Visual color-coded hints are provided.</p>	<p>(Melero et al., 2012)</p>
59	Project Execution Game (PEG)	Software Engineering		<p>In this game, the players work in groups in order to manage a project. They receive a detailed project plan and the success evaluation criterion, which is to complete the project within the minimum possible budget. Project overruns due to players' decisions, cause penalties and overhead cost, which negatively affect the team's success.</p>	<p>(Zwikael e Gonen, 2007)</p>





60	Project Risk	Software Engineering		The board game is based on a path through the project management lifecycle. A project manager pawn progresses through the path over 12-rounds of project play (representing project reporting periods). Progressing steps may cost or return chips. Six game pawns represent team members (that can be lost through certain project risks). When playing competitively, the winner is the one who finishes with most chips and team members left.	(Project Risk, 2001)
61	Project-o-poly (PoP)	Software Engineering		The goal of the game is to achieve the highest possible profit by renting, buying and selling the projects located around the game board, until becoming the richest Project Manager (PM) and, possibly, the monopolist. Starting from "Go!", every PM moves his/her token on the game board by rolling the dice.	(Project-o-poly, 2007)
62	Requirements Activity: Lego House	Software Engineering	N/I	In the beginning, a project is described by the instructor (client) for building a house. The players have 30 minutes to provide a house that meets the client's needs. Throughout the game, the players can elicit requirements three times with the client asking three questions each time. At the end, the groups present the constructed houses in order to check, if they satisfy the client's requirements.	(Ludi, 2010)
63	Requirements collection and analysis game (RCAG)	Software Engineering		The basic idea of the game is for a team to manage and deliver a number of software development projects. Each player has a specific role such as project manager, systems analyst, systems designer or team leader. Several project scenarios are available with underlying business models that define budget, schedule and resources. The player(s) assigned to the system analyst role has to identify the requirements for the project.	(Hainey et al., 2011) (Connolly et al., 2007)
64	Requirements Game	Software Engineering		The players assume different roles (director, analyst, designers, and programmers). The instructor coordinates the game execution and acts as customer. During the game, the groups have to develop a requirements specification, including, e.g., a model of the proposed solution, entity-relationship diagram and relational model. Raw materials, computer time and team member work time are restricted. The game can be played in 1- or 2-cycles depending on the available time.	(Zapata e Awad-Aubad, 2007)



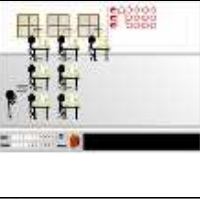
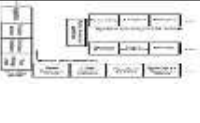

65	Robot Trouble	Programming Fundamentals		The player is supposed to navigate a robot through a maze of traps to reach the destination by programming its movements.	(Robot Trouble, 2011)
66	Robozzle	Programming Fundamentals		RoboZZle is a programming puzzle game. The player has to program a robot to collect all items from 2D tri-colored field.	(Robozzle, 2011)
67	Saving Princess Sera	Programming Fundamentals		Saving Sera is a 2D exploratory game, where the player has to rescue the kidnapped princess Sera. Therefore, the player has to create “machine” (which are programs) to solve problems. In Saving Sera, the player fixes the machine by unscrambling a while loop to track a fisherman’s catch; debugging a nested for loop placing eggs in crates; and visually piecing together a flowchart for quicksort. When the player makes a mistake, s/he must fight a script bug by answering various computer science questions.	(Eagle e Barnes, 2009)
68	Scrum Lego	Software Engineering		Players execute sprints building LEGO houses and vehicles from user stories following the SCRUM process and performing SCRUM ceremonies.	(Scrum Lego City, 2009)
69	SDM - Software Development Manager	Software Engineering		In this game, the player has a team of employees to develop software. The game presents possibilities to the player to decide on development strategies and to define roles for each staff member. When the software is completed and delivered to the customer, there is a quality assessment of the software and a project completion payment.	(Kohwalter et al., 2011)

70	Secret Ninja Testing	Software Engineering		HALO presents a series of quests of software engineering tasks. Quests can either be individual, requiring a developer to work alone, or in groups, requiring a developer to form a team and work collaboratively towards their objective. Completing each quest gives the students experience points and achievements. Through a global leader board students can track their own and other students' achievements and experience points.	(Bell et al., 2011)
71	Secure Volunteer Game	Net Centric		In the game, the player joins as "volunteer" an organization with the task to enable connectivity. The player sets up VPN connectivity to enable relief workers in the field to securely share environmental, health and medical information with the headquarters, while moving through virtual rooms in the game and interacts with fictitious characters to configure a VPN concentrator.	(Secure Volunteer Game, 2010)
72	Security Protocol Game	Security		Within each group, one player is selected to play Alice or Bob, the two communicating parties. Another player is selected to play Gavin. The same player may also take the role of Colin. The remaining player(s) take the role of Trudy the intruder. The game starts with the players seated around a table. The students select a game scenario to play, e.g., Alice wishes to purchase computer software from Bob over the Internet using her credit card for payment, and a protocol to use such as the Transport Layer Security protocol.	(Hamey, 2003)
73	SESAM	Software Engineering		Students get a project of a given size assigned. They take the role of a project manager, aiming to complete the project within given time and budget by a team of simulated, virtual software engineers. Students can hire personnel from a pool of persons with different qualifications and different expected salary. The player controls the simulator using a purely textual interface.	(Drappa e Ludewig, 2000)
74	SharkWord	Software Engineering		The game creates a virtual environment in which projects develop in (accelerated) real time. The player is forced to act to problems immediately and intervene properly. The game is propelled by an underlying suspense story in the context of building a shark aquarium. The game covers not only economic aspects, but also social aspects, conflict management and diplomatic skills.	(Sharkworld, 2008)


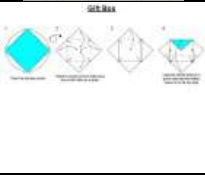
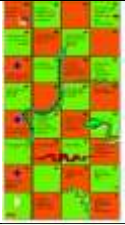

75	SimjavaSP	Software Engineering		The student, acting as project manager, has to develop a simulated software project within the required time and budget, and of acceptable quality.	(Shaw e Dermoudy, 2005)
76	SimSE	Software Engineering		A customizable simulation environment for educating students in software processes/management. The environment supports the creation and simulation of game-based software process simulation models (e.g., waterfall, incremental, XP). In the game, learners take on the role of the project manager and must manage a team of developers in order to successfully complete a software project or task in conformance to the respective software process model.	(Navarro e Hoek, 2007) (Navarro, 2006)
77	Simsoft Game	Software Engineering		Simsoft players are formed into teams of 2 or more and they are given a scenario that describes the requirements for a small software development project. Taking the role of project manager, the team must manage the project from start-up to final delivery. The players gather around a printed game board to discuss the current state of the project and to decide on their next move. The board shows the flow of the game while plastic counters are used to represent the staff of the project.	(Caulfield et al., 2011)
78	SimSYS	Software Engineering		The game world represents a software development organization (offices, meeting rooms). The player starts play as Scrum Master for an agile development process. S/he is presented with a product description that needs to be developed for a product owner. The player must select a diverse, adequately skilled scrum team, elicit requirements and create a release backlog with the product owner and, then, manage development sprints (short iterations).	(Longstreet e Cooper, 2011)
79	Simulate Computer Game	Computer Architecture and Organization	N/I	Every player assumes a role (processor, mouse) identified by props (big nerd glasses, helmet etc). Then, simple functions are executed by players.	(Simulate Computer Game, 2008)





80	SimulES	Software Engineering		SimulES is based on the Problems and Programmers game introducing a game board to organize the cards. Similar to Problems and Programmers, the players' objective is to be the first to complete a pre-defined software project. A player performs different roles such as software engineer, technical coordinator, quality controller and project manager.	(Monsalve, 2011) (Magno et al., 2006)
81	SimulES-W	Software Engineering		SimulES-W is a digital version of the game SimulES. The game allows a student to take on different roles (project manager, auditor, software engineer) in a project to build software, allowing him/her to experience common tasks and decisions in the context of software development.	(Monsalve et al., 2011)
82	SimVBSE	Software Engineering		The game starts with a visit to the CEO's briefing room, where the student through animated videos is asked to take on the role of a project manager, and briefed on the current organizational situation and the student's overall objective in the game. Making a move in the game involves visiting different rooms (board room, lounge) and choosing one or more of the available options.	(Jain e Boehm, 2006)
83	Sorting Algorithms	Algorithms and Complexity		In this game, the student has to collect boxes and sort them according their values by using sorting algorithms (Bubble and Bucket Sorting).	(Sorting Algorithms, 2012)
84	SortingCasino	Algorithms and Complexity		SortingCasino resembles the card game Casino. On a turn, player can capture one or more cards from the table by using one of his or her hand cards. With an algorithm card, player can capture all special cards that are valid for that algorithm. Respectively, with a special card, player can capture all algorithm cards that are valid for the criteria in the special card and put them on his/her victory stack. The game ends when either algorithm stack or special card stack is empty. The winner is the player with most cards in her/his victory stack.	(Hakulinen, 2011)





85	SortingGame	Algorithms and Complexity		<p>SortingGame is a card game that includes two decks of cards: algorithm cards and special cards. Each algorithm card has a name of one sorting algorithm on it. Special cards contain criteria that either apply or do not apply for a given algorithm. In the beginning, the dealer deals 3 algorithm cards and 2 special cards to each player. The actual game round consists of two phases: special card phase and algorithm phase. During the special card phase, each player can place one special card to the table. In algorithm phase, each player places one algorithm card to the table. The algorithm should be valid for the active special cards on the table. The winner of the round is the player whose algorithm's asymptotical time complexity is the best. The winner collects all the cards played during the round and adds them to his/her victory stack.</p>	(Hakulinen, 2011)
86	Starter MMO	Net Centric		<p>The game combines routing and forwarding. Students start out in an area, marked start on a map representing a maze. In this area there is a Non Player Character (NPC) that gives them quests to perform. These quests all consist of delivering a package (representing a 'data packet') to another NPC (host) somewhere. The students are neither told where the destination is located nor about the layout of the maze. The aim is to run to the NPC's location (simulating transmission of the data packet), and deliver the package to them.</p>	(Graven et al., 2009)
87	Subnet Game	Net Centric		<p>This game allows the player to solve a number of IP subnetting problems in top secret Area 51. Each level must be solved in an allotted amount of time.</p>	(Subnet Game, 2008)
88	Subnet Troubleshooting Game	Net Centric		<p>The job of the player is to identify problems in otherwise correct subnets and fix them so the network works properly. When all problems in a level have been solved, the player moves to the next level. The objective of the game is to complete the problems in all the levels in the game and get the highest score possible.</p>	(Subnet Troubleshooting Game, 2010)



89	The Catacombs	Programming Fundamentals		In this 3D fantasy game, the user is an apprentice wizard who must cast three increasingly complex spells (programs) to save two children who are trapped in the catacombs. The first spell uses IF statements to magically unlock a door, the second uses nested FOR loops to construct a bridge, and the third spell uses nested FOR loops to solve a cryptogram.	(Barnes et al., 2007)
90	The Hard Choices	Software Engineering		The Hard Choices game board represents activities of a software development release. In their quest to become the market leader, players are competing against each other to release their product to the market place. Players earn points for landing on a square with a tool (representing rewards for investing in technical infrastructure) or by not finishing last (representing rewards for speed to market). When a player crosses a hard choices square, s/he must decide whether to go over the shortcut bridge or to go the long way and try to collect one or more tool cards.	(Brown et al., 2010)
91	The Incredible Manager	Software Engineering		In the game, the learner acts as a project manager being responsible for planning, executing, and controlling a software project. The goal is to complete a project, whose cost and schedule are established during a planning phase and approved by stakeholders. Project execution occurs in continuous turns, consuming the planned resources. The learner must monitor the project execution and take corrective actions, when necessary. Visual effects and reports provide feedback, showing exhausted developers, late tasks.	(Dantas et al., 2004) (Barros et al., 2006)
92	The MIS Game	Software Engineering		Key elements are a board with tokens representing progress; artifacts the player can “own” (e.g., money); “events” that happen and the dice, which “drives” progress. Players are given a certain budget at the start and are required to acquire and deploy resources to develop a portfolio of systems. Then the player focus on the IS development.	(Martin, 2000)
93	The Mystery of Traffic Lights	Digital Logic		Designed from a first-person perspective, the game starts at the major intersection of a small town, where an engineer character, Jack, is standing frustrated by the busy and messy traffic due to malfunctioning traffic lights. He then invites students to help him redesign the controller with the right logic for the current traffic flow.	(Tang et al., 2012)



94	The Orange game	Net Centric		<p>Players are labeled with a letter of the alphabet and for each player two oranges are marked with the player's letter. The oranges are distributed equally to the players (except one who receives only one orange). Players pass the oranges around until each child gets the oranges labeled with their letter of the alphabet.</p>	(Orange Game, 2002)
95	The Software Development Game	Software Engineering		<p>Players must build origami boxes with one of the following four groups of letters, SO, FT, WA or RE. Every box represents a software module. One group of four modules forms one software piece (a complete word, SOFTWARE). Every module must accomplish a set of pre-defined requirements, which can be discussed with the director of the game. The goal, therefore, is that the players must compete in groups to gain profits from an imaginary software company that makes software modules.</p>	(Zapata, 2009)
96	The Waterfall Game	Software Engineering		<p>The game is a snake and ladders-style game in which players take a turn to throw the dice and advance on the game board.</p>	(Waterfall Game, 2009)
97	Tower of Cubes	Algorithms and Complexity		<p>A list of cubes in the tower represents the data in a stack or queue depending on the selected mode ( Stack or Queue). When the game starts, 5 cubes with two random colors are dropped into the tower. Subsequently, a new cube with a random color is dropped into the tower automatically either to the top/rear of the tower depending on the mode. When two consecutive cubes with the same color meet, both cubes will be disappeared. The player needs to move the cube in and out in order to clear the cubes in the tower and the score will be increased. When all cubes are cleared in the tower, the player wins.</p>	(Tan e Seng, 2010)

98	Treasure Hunt	Programming fundamentals		<p>The goal of the game is to get the pirate to the treasure by defining his treasure map. Therefore, the player has to drag commands from the panel on the right to the main command area and then execute the program to have the pirate follow the created "treasure map".</p>	(Treasure Hunt, 2011)
99	TREEZ	Algorithms and Complexity		<p>In the single-player version, a player is given a randomly generated binary tree with labeled nodes. Then, s/he must traverse the tree as if he/she were an ant crawling on the tree's surface with the object to move through the tree in the correct order. Time restrictions are imposed on individual moves, and the player gets to compete against his/her previous fastest time. In the two-player version, each player is given a different tree. Then, players alternate turns.</p>	(Ford e Minsker, 2003)
100	Using Games in Software Engineering Education to Teach Risk Management	Software Engineering		<p>In the game, each player assumes the role of a project manager. The game's objective is to develop a product, sell it in the market and win by having more money at the end of the game than all the other players. The game has 5 stages: planning, requirements, architecture &amp; design, implementation and testing. Yet, unlike traditional board games, players are free to roam around, choose where to go and what to do, without the need for throwing a die to advance.</p>	(Taran, 2007)
101	U-Test	Software Engineering		<p>The game is based on a case in which the player is considered a candidate for a tester job in a software company. After a job interview, the player must solve challenges by preparing unit test cases. The game presents the player's score based on his/her performance.</p>	(Thiry et al., 2011)

102	ViRPlay3D2	Programming Fundamentals		<p>Students are immersed using a first-person view that simulates the point of view of the objects. An aiming point in the center of the screen serves to interact with other entities in the virtual world and to represent message passing. The user interface also displays a score that contains information about the current state of the game session. In the single-player mode, the user does not perform the role of any object, but s/he is an external avatar that observes and partially controls the game. In multi-user mode, ViRPlay3D2 represents a virtual space where users discuss the interactions among objects needed to carry out a scenario execution.</p>	(Jiménez-Díaz et al., 2007)
103	Welcome to SCRUMIA	Software Engineering		<p>The game deals with the planning and execution of a sprint applying SCRUM where the students have to produce paper boats, hats and planes. Each group member takes on a specific role (SCRUM master). During the game, the students execute the SCRUM process, including sprint planning, daily meetings. Winner of the game is the group that obtained the highest profit and business value.</p>	(SCRUMIA, 2009)
104	Wireless Explorer	Net Centric		<p>Players compete by ensuring friendly space aliens remain in wireless communication throughout their galactic voyages. Welcoming an envoy of alien scientists, players have to configure open wireless access to a space ship mainframe correctly for each alien. Individual and team scores are based on successful site survey strategies, deployment skills and network maintenance.</p>	(Wireless Explorer, 2010)
105	Wu's Castle	Programming Fundamentals		<p>Players interact in two ways: by manipulating arrays changing loop parameters and by physically walking the game character through loop execution. After an introduction to the game story and interface, the player walks through a visual representation of for-, while-, and do-while-loops. The player then manipulates a one-dimensional array by setting the parameters in a for loop. The player repeats this process with a nested loop walkthrough and two-dimensional array manipulation.</p>	(Eagle e Barnes, 2009)

106	X-MED	Software Engineering		<p>In this game, the learner takes the role of a measurement consultant in a software organization. During the game session, the player passes sequentially through all steps of a GQM-based measurement program creating measurement artifacts based on a series of constrained selections with pre-defined alternatives. For each decision, the game gives a feedback and score. At the end a total score and feedback report is provided.</p>	(Wangenheim et al., 2009)
107	Z-Buffer	Computer Architecture and Organization		<p>In the game a series of image fields is given on which a student should click to determine the value of each bit, i.e. contents of the registry in various situations presented in the task. The correct answer to fill the content of one bit is one of the proposed answers presented to students in the form of squares to be selected.</p>	(Kuk et al., 2012)

## APÊNDICE B – Classificação dos jogos com relação aos aspectos instrucionais.

No	Name	Learning objective/purpose	Level(s) of learning objects	Feedback
1	3DAR Lego Game	To improve spatial competencies, such as, analyzing or assembling a 3D model structure for augmented reality.	Cognitive: Knowledge, Comprehension	N/I
2	Age of Computers	To learn computer fundamentals.	Cognitive: Knowledge, Comprehension	Immediate response is given when the player enters an answer.
3	Algorithms Recursive Game	To learn recursive algorithms.	Cognitive: Knowledge	N/I
4	AMEISE	To experience software project management.	Cognitive: Knowledge, Comprehension, Application	The game provides a consultant and the friendly peer that can be seen as parametric agents, which support the students while managing a project, whereas an evaluation component gives the students the possibility to analyze their management performance at the end of the game.
5	Anti-Phishing Phil	To learn how to identify phishing URLs, where to look for cues for trustworthy or untrustworthy sites in web browsers, and how to use search engines to find legitimate sites in order to avoid security attacks.	Cognitive: Knowledge, Comprehension	Provides feedback immediately after each action and provides a feedback summary after each round, indicating errors with brief explanations.
6	Battleships	To learn about search algorithms.	Cognitive: Knowledge, Comprehension	Debriefing discussion
7	BattleThreads	To learn about threads and the distinctions between threads and processes, advantages of a multithreaded organization in structuring applications and in performance.	Cognitive: Knowledge, Comprehension	N/I
8	Binary Game	To learn binary numbers for networking.	Cognitive: Knowledge, Comprehension, Application	N/I
9	BINX	To learn about number systems and their arithmetic operations.	Cognitive: Knowledge, Comprehension	The game displays an assessment screen that provides the player with an assessment of his/her comprehension.
10	Bombberman Game	To learn basics of C programming.	Cognitive: Knowledge, Comprehension, Application	By running the code, the results are visualized.

11	BOTS	To learn basic programming concepts.	Cognitive: Knowledge, Comprehension, Application	N/I
12	CARGO-BOT	To learn basics of software programming.	Cognitive: Knowledge, Comprehension, Application	N/I
13	CEEBOT	To learn the basics of an object oriented programming language similar to C++, C# or Java.	Cognitive: Knowledge, Comprehension, Application	By running the code, the results are visualized.
14	C-Jump	To learn basics of programming languages, such as C, C++ and Java, such as "if", "else", "switch", and variable "x" concept.	Cognitive: Knowledge, Comprehension, Application	N/I
15	COLOBOT	To learn the basics of an object oriented programming language similar to C++, C# or Java.	Cognitive: Knowledge, Comprehension, Application	By running the code, the results are visualized.
16	Computer Architecture Mini-Game	To learn the main elements of a motherboard and the main elements of Boolean logic.	Cognitive: Knowledge, Comprehension	N/I
17	Computer Theory Jeopardy Game	To reinforce computer theory concepts.	Cognitive: Knowledge	Instructor's comments to the answers in the classroom.
18	Computing Networks Mini-Game	To learn computer network concepts.	Cognitive: Knowledge, Comprehension	N/I
19	Conquer the Net	To learn basic configuration commands (e.g., to assign an IP address, access lists (ACLs), basic PC configuration under Linux, standard port numbers).	Cognitive: Knowledge	In order to allow players to check their progress, a component that checks the connectivity between any two devices is provided.
20	Control-Alt-Hack	To increase people's awareness of computer security needs and challenges.	Cognitive: Knowledge, Comprehension; Affective: Valuing	N/I
21	CounterMeasures	To learn specific techniques used by security experts and to allow to practice these techniques.	Cognitive: Knowledge, Comprehension, Application	N/I
22	CyberCIEGE	To enhance computer security education by demonstrating the abstract functions and limitations of security mechanisms.	Cognitive: Knowledge, Comprehension, Application	After game play has completed, the player experiences a debriefing in which the player choices are critiqued by the system. The game displays a log of attacks describing the means by which assets were compromised.

23	Database concurrency control card game	To learn database concurrency, especially knowledge about transactions and concurrency control protocols to simulate the work of a transaction processing system.	Cognitive: Knowledge, Comprehension, Application; Skill: Mechanism	N/I
24	Databases: Lots of Data and Getting Quick Results	To learn basic database concepts such as table, records, fields, queries, and the performance differences between doing a table scan and indexing the table.	Cognitive: Knowledge, Comprehension	N/I
25	Dealing with difficult people	To illustrate difficulties and to make students recognize the importance of team management skills in software projects.	Affective: Valuing	N/I
26	Deliver!	To reinforce Earned Value Management (EVM) concepts and to teach the competency to apply basic EVM calculations.	Cognitive: Knowledge, Comprehension, Application	Feedback through debriefing discussion.
27	DesigMPS	To teach software process modeling in the context of software process improvement.	Cognitive: Knowledge, Comprehension, Application	At the end of each game level, the created process model is automatically compared to the correct solution and an error report is presented.
28	Detective Game – what killed the project?	To reinforce Earned Value Management (EVM) concepts and to exercise the application of EVM.	Cognitive: Knowledge, Comprehension, Application	Correct answers are presented in the end of the game and discussed with the students.
29	Digi Island	To learn K-map optimization.	Cognitive: Knowledge, Comprehension	N/I
30	Digital Logic and Electronics Concepts	To learn boolean algebra.	Cognitive: Knowledge, Comprehension	N/I
31	Digital System Game	To enhance learning on digital systems.	Cognitive: Knowledge, Comprehension	When the player has completed a circuit, s/he clicks a button that invokes a Boolean logic solver in the prototype, which determines if the circuit is equivalent to the truth table. The game visually indicates how many of the truth table combinations the circuit satisfies.
32	DSasketch	To practice concepts related to data structures and algorithms.	Cognitive: Knowledge, Comprehension	N/I
33	EleMental: The Recurrence	To learn recursion through depth-first search of a binary tree.	Cognitive: Knowledge, Comprehension, Application	If, at any time during the coding, students compile incorrect code, they receive the compiler error message, indicating which line is incorrect.

34	ERPsim	To develop hands-on understanding of the concepts underlying enterprise systems and to develop technical ERP skills as well as to experience the benefits of enterprise integration and how to create, execute, and adapt a business strategy in a real-time environment.	Cognitive: Knowledge, Comprehension, Application	Debriefing discussion.
35	Groupthink/Second Life	To teach students the importance of software specifications.	Affective: Valuing; Skills: Mechanism (teamwork, communication)	Correct answers are revealed at the end of the game.
36	hACME Game	To learn about software security, mainly, in the field of web applications.	Cognitive: Knowledge, Comprehension, Application	After completing a challenge, the user receives a performance statistic, including the achieved score, the number of hacking attempts used and the number of hints received for the given challenge.
37	Immersive Security Education Environment (I-SEE)	To understand basic concepts on security attacks.	Cognitive: Knowledge, Comprehension	The player can use the management system to obtain feedback on their performance.
38	INNOV8	To learn business process management.	Cognitive: Knowledge, Comprehension, Application	N/I
39	IT Manager	To learn IT competencies.	Cognitive: Knowledge, Comprehension, Application	N/I
40	JV2M	To learn how to compile Java and to promote a better understanding of the underlying mechanisms of object-oriented programming.	Cognitive: Knowledge, Comprehension, Application	Once a number of wrong instructions has been reached or time expires, the system notifies the player, and s/he can revert the JVM to the last correct state.
41	Kid Krypto	To learn about public key encryption.	Cognitive: Knowledge, Comprehension	N/I
42	Lego Factory	To learn about production optimization using artificial intelligence techniques.	Cognitive: Knowledge; Skill: Mechanism	During the course of the game, the effect of a change in the production process becomes immediately visible in order to guide the player in the configuration task.
43	Lego Tower Team Activity: Managing Change	To introduce students to team building in the context of design, specifically in terms of design volatility and the need for a flexible design.	Cognitive: Knowledge, Comprehension, Application; Affective: Valuing	Debriefing discussion.
44	Light-Bot	To teach programming logic (functions, conditionals, recursion).	Cognitive: Knowledge, Comprehension, Application	N/I



45	MIS Project Manager	To provide understanding on IS development project management and decision making.	Cognitive: Knowledge,Comprehension	Once completed the project, on-line feedback is given on the decisions made.
46	Mission to mars - Release or iteration planning	To learn about the planning process in iterative software development.	Cognitive: Knowledge,Comprehension, Application	Debriefing with the instructor.
47	myPlanNet Simulation	To learn how to build a computer network.	Cognitive: Knowledge, Comprehension, Application	N/I
48	Next Generation_7 helicopter game	To learn about the software process and to identify strengths and weakness of both waterfall and iterative development models.	Cognitive: Knowledge, Comprehension, Application; Affective: Responding; Skills: Mechanism (Communication and Coordination)	The post game retrospective enables both the students and the coordinators to learn together. The students reflect and articulate the strengths and weaknesses of the process models.
49	.NET Terrarium 2.0	To learn about the .NET programming model and languages.	Cognitive: Knowledge, Comprehension, Application	N/I
50	Operating systems role plays	To learn concepts of multi-programming, concurrency and synchronization.	Cognitive: Knowledge, Comprehension.	N/I
51	Paper Tower Competition	To learn to apply Earned Value Management (EVM) for project monitoring & control.	Cognitive: Knowledge, Comprehension, Application	Feedback is given during a debriefing discussion with the students.
52	Path	To learn basic programming concepts.	Cognitive: Knowledge, Comprehension, Application	Results of running the submitted solutions on public test cases are displayed to help students to check their answers.
53	PDConsole	To understand scalability issues, diversity of failure modes in and designing distributed algorithms that can handle those failures.	Cognitive: Knowledge, Comprehension	N/I
54	PlayScrum	To learn concepts, practices and practical issues related to Scrum.	Cognitive: Knowledge,Comprehension, Application	N/I
55	PM Master	To review and reinforce basic software project management concepts in conformance with PMBOK.	Cognitive: Knowledge,Comprehension	Correct answers are given on the back of each question card.
56	Problems and Programmers	To teach software process issues and to promote proper software engineering practices.	Cognitive: Knowledge,Comprehension, Application	N/I
57	Process State Transition	To teach modeling and data structures related to process management in the transition of processes between execution states, and the performance of the scheduling function.	Cognitive: Knowledge, Comprehension	N/I

58	Programming Fundamentals Mini-Game	To learn three different algorithms structures: sequential, selective and iterative.	Cognitive: Knowledge, Comprehension	N/I
59	Project Execution Game (PEG)	To provide experiences in project execution and to train for teamwork. The game also intends to improve the quality of planning and to improve risk management.	Cognitive: Knowledge, Comprehension, Application; Skill: Mechanism (Teamwork, Decision Making, Problem-Solving)	Through the simulation directing staff, a feedback is given continuously throughout the game.
60	Project Risk	To learn about risk management in alignment with PMBOK.	Cognitive: Knowledge, Comprehension, Application	N/I
61	Project-o-poly (PoP)	To learn project management.	Cognitive: Knowledge, Comprehension, Application	N/I
62	Requirements Activity: Lego House	To provide students the opportunity to experience the requirements engineering process.	Cognitive: Knowledge, Comprehension, Application; Affective: Valuing	Debriefing discussion
63	Requirements collection and analysis game (RCA G)	To learn about requirements collection and analysis.	Cognitive: Knowledge, Comprehension, Application	N/I
64	Requirements Game	To learn about requirements analysis.	Cognitive: Knowledge, Comprehension, Application	N/I
65	Robot Trouble	To learn simple procedural programming.	Cognitive: Knowledge, Comprehension	N/I
66	Robozzle	To learn programming.	Cognitive: Knowledge, Comprehension	N/I
67	Saving Princess Sera	To learn programming concepts, including, loop and quicksort algorithm.	Cognitive: Knowledge, Comprehension, Application	Feedback is given immediately visually and through simple explanations.
68	Scrum Lego City	To experience SCRUM.	Cognitive: Knowledge, Comprehension, Application; Skill: Mechanism (Cooperation, Communication, Mediation)	N/I
69	SDM - Software Development Manager	To assist in learning the concepts of human resource management.	Cognitive: Knowledge, Comprehension, Application; Affective: Valuing	At the end of the game, the player receives a simple performance report.
70	Secret Ninja Testing	To learn software testing.	Cognitive: Knowledge, Comprehension; Affective: Valuing	Feedback is given in form of experience points and achievements.
71	Secure Volunteer Game	To test Virtual Private Networking (VPN) skills.	Cognitive: Knowledge	Feedback on the correctness of responses is given immediately.

72	Security Protocol Game	To understand secure communication protocols.	Cognitive: Knowledge, Comprehension	N/I
73	SESAM	To experience software project management.	Cognitive: Knowledge, Comprehension, Application	In the end, a score is presented. The player can also analyze her/his performance using an analysis tool, which graphically displays the internal variables.
74	SharkWord	To experiment and gain experience with key aspects of project management.	Cognitive: Knowledge, Comprehension, Application	N/I
75	SimjavaSP	To allows students to gain experience of managing a software development project.	Cognitive: Knowledge, Comprehension, Application	Provides a combination of graphical and textual feedback. Students receive continual feedback as the project progresses.
76	SimSE	To learn software engineering practices and models.	Cognitive: Knowledge, Comprehension, Application	At the end of the game, a performance score report and a visual analysis of the game session is given, indicating which rules were triggered, a trace of events, and the "health" of various attributes (e.g., correctness of the code).
77	Simsoft Game	To learn software engineering and project management concepts.	Cognitive: Knowledge, Comprehension, Application, Analysis, Evaluation, Synthesis; Affective: Responding, Organizing; Skills: Mechanism (Communication & Teamwork)	Post-game gatherings are held where the players discuss their strategies and experiences.
78	SimSYS	To learn software engineering.	Cognitive: Knowledge, Comprehension, Application	N/I
79	Simulate Computer Game	To understand how computers work.	Cognitive: Knowledge, Comprehension	N/I
80	SimulES	To learn software project management.	Cognitive: Knowledge, Comprehension, Application	N/I
81	SimulES-W	To learn software project management.	Cognitive: Knowledge, Comprehension, Application	The game offers feedback during the gameplay.
82	SimVBSE	To better understand value-based software engineering.	Cognitive: Knowledge, Comprehension, Application	The game provides an assessment report that discusses the strengths and weaknesses of each of the options that were made available to the student based on his/her choices.
83	Sorting Algorithms	To learn sorting algorithms.	Cognitive: Knowledge	The game indicates errors.
84	SortingCasino	To learn sorting algorithms and understand related concepts, such as, stable, in-place and	Cognitive: Knowledge, Comprehension; Skill: Mechanism	It is up to the players to find out the right answers to the questions.

		big O notation.		
85	SortingGame	To learn sorting algorithms and understand related concepts, such as: stable, in-place and big O notation.	Cognitive: Knowledge, Comprehension; Skill: Mechanism	It is up to the players to find out the right answers to the questions.
86	Starter MMO	To learn about routing in computer networks.	Cognitive: Knowledge	N/I
87	Subnet Game	To learn IP subnetting.	Cognitive: Knowledge, Comprehension	N/I
88	Subnet Troubleshooting Game	To learn subnetting.	Cognitive: Knowledge, Comprehension	N/I
89	The Catacombs	To learn programming concepts (loops).	Cognitive: Knowledge, Comprehension	Incorrect answers result in decreasing player health.
90	The Hard Choices	To communicate the concepts of uncertainty, risk, options, and technical debt in the software development cycle.	Cognitive: Knowledge, Comprehension	Feedback is given during a debriefing discussion with the learners.
91	The Incredible Manager	To learn software project management.	Cognitive: Knowledge, Comprehension, Application	Achievement of steps is communicated during the gameplay.
92	The MIS Game	To provide understanding of traditional information systems development life-cycle concepts, IS development project management and decision making.	Cognitive: Knowledge, Comprehension	N/I
93	The Mystery of Traffic Lights	To learn about state machine diagram, state table, and behavior HDL.	Cognitive: Knowledge, Comprehension	N/I
94	The Orange game	To understand routing and deadlock problems.	Cognitive: Knowledge, Comprehension	Debriefing discussion.
95	The Software Development Game	To demonstrate the particularities of software development: communication breakdown, isolated work, and lack of planning.	Affective: Valuing	N/I
96	The Waterfall Game	To learn about working on a project using the Waterfall Model.	Cognitive: Knowledge, Comprehension	N/I
97	Tower of Cubes	To learn data structures on stacks and queues.	Cognitive: Knowledge, Comprehension	N/I
98	Treasure Hunt	To understand the basic concepts of programming.	Cognitive: Knowledge, Comprehension	By running the commands, the results are visualized.
99	TREEZ	To reinforce knowledge of tree traversal techniques.	Cognitive: Knowledge, Comprehension	N/I
100	Software Risk Management	To learn risk management concepts, to enable learners to make risk-based decisions on their	Cognitive: Knowledge, Comprehension, Application	N/I

	Game	own and to understand the complexity of software projects.		
101	U-Test	To recognize and understand software testing concepts and to apply techniques of data entry selection, equivalence class partitioning and boundary-value analysis.	Cognitive: Knowledge, Comprehension, Application	Feedback is provided through a graphical indicator at the end of each challenge. At the end of the game, the player receives a performance score and his/her placement among all players.
102	ViRPlay3D2	To learn object-oriented design concepts such as objects, classes and message passing.	Cognitive: Knowledge	Provides feedback while the game is played.
103	Welcome to SCRUMIA	To strengthen the understanding of SCRUM concepts and to exercise the application of the SCRUM process.	Cognitive: Knowledge, Comprehension, Application	Feedback is given during a debriefing discussion.
104	Wireless Explorer	To test networking knowledge and wireless skills.	Cognitive: Knowledge, Comprehension, Application	N/I
105	Wu's Castle	To learn for-, while- and do-while-loops and arrays.	Cognitive: Knowledge, Comprehension, Application	The player receives feedback after each answer attempt. If the student's code fails to compile, the <i>machina</i> falls unconscious, which represents a syntax error and the student must try again.
106	X-MED	To learn software measurement by simulating the definition and execution of a QOM-based measurement program for project management in alignment with CMMI-DEV maturity level 2.	Cognitive: Knowledge, Comprehension, Application	The game provides feedback immediately after each decision made during the game session.
107	Z-Buffer	To learn concepts related to the Z-buffer algorithm (depth and color test).	Cognitive: Knowledge, Comprehension, Application	N/I

### APÊNDICE C – Classificação dos jogos com relação aos aspectos de jogos.

No	Name	Media	Platform	Genre	Interaction mode	Duration of game session	Available language(s)	License	Cost (US\$)
1	3DAR Lego Game	Digital	PC/Stand-alone	Puzzle	Single-player	N/I	English	N/I	N/I
2	Age of Computers	Digital	PC/Online	Quiz	Multi-players	N/I	Norwegian	Open Source	US\$ 0
3	Algorithms Recursive Game	Digital	PC/Online	Roll-and-move	Multi-groups	N/I	English	N/I	N/I
4	AMEISE	Digital	PC/Stand-alone	Simulation	Multi-players	N/I	English	N/I	N/I
5	Anti-Phishing Phil	Digital	PC/Online	Action	Single-Player	10 min	English	Proprietary (developed by CMU)/ commercialized by Wombat Security Technologies)	N/I
6	Battleships	Non-Digital	Paper& Pencil	Strategy	Multi-groups	N/I	Brazilian Portuguese, English, French, Greek, Hungarian, Italian, Polish, Russian, Slovenian, Turkish	Creative Commons	N/I
7	BattleThreads	Non-Digital	Board	Strategy	Multi-groups	N/I	English	N/I	N/I
8	Binary Game	Digital	PC/Online	Action	Single-Player	N/I	English	Proprietary	US\$0
9	BINX	Digital	Console	Adventure	Single-player	90 min	English	N/I	N/I
10	Bomberman Game	Digital	PC/Stand-alone	Puzzle	Single-player	N/I	English	N/I	N/I
11	BOTS	Digital	PC/Online	Puzzle	Multi-players	N/I	English	N/I	N/I
12	CARGO-BOT	Digital	Mobile	Puzzle	Single-player	N/I	English	Proprietary	US\$0

13	CEEBOT	Digital	PC/Stand-alone	Puzzle	Single-player	N/I	English	Proprietary	US\$80.86
14	C-Jump	Non-Digital	Board	Racing	Multi-groups	30-50 min	English	Proprietary	US\$24.95 within US; US\$49.95 worldwide
15	COLOBOT	Digital	PC/Stand-alone	Adventure	Single-player	N/I	English, French, Polish	Proprietary	US\$80.86
16	Computer Architecture Mini-Game	Digital	PC/Online	Puzzle	Single-player	N/I	Spanish	N/I	US\$0
17	Computer Theory Jeopardy Game	Non-Digital	Paper& Pencil	Quiz	Multi-groups	N/I	English	N/I	N/I
18	Computing Networks Mini-Game	Digital	PC/Online	Puzzle	Single-player	N/I	Spanish	N/I	US\$0
19	Conquer the Net	Digital	PC/Stand-alone	Strategy	Multi-groups	N/I	English	N/I	N/I
20	Control-Alt-Hack	Non-Digital	Card	RPG	Multi-groups	60min	English	Proprietary	US\$25 per group
21	Counter Measures	Digital	PC/Online	Simulation	Single-Player	N/I	English	N/I	N/I
22	CyberCIEGE	Digital	PC/Online	Simulation	Single-Player	N/I	English	Proprietary	Available at no cost to agencies of the US Government and educational institutions
23	Database concurrency control card	Non-Digital	Card	Simulation	Multi-groups	N/I	Brazilian Portuguese	N/I	N/I

	game								
24	Databases: Lots of Data and Getting Quick Results	Non-Digital	Paper& Pencil	Simulation	Multi-groups	10-15min	English	Creative Commons	N/I
25	Dealing with difficult people	Non-Digital	Paper& Pencil	Simulation	Multi-groups	25min	English and Brazilian Portuguese	Creative Commons	US\$1 per group
26	Deliver!	Non-Digital	Board	Simulation	Multi-groups	90min	English and Brazilian Portuguese	Creative Commons	US\$10 per group
27	DesigMPS	Digital	PC/Stand-alone	Simulation	Single-player	90min	English	N/I	N/I
28	Detective Game – what killed the project?	Non-Digital	Paper& Pencil	Simulation	Multi-groups	90min	English and Brazilian Portuguese	Creative Commons	US\$5 per group
29	Digi Island	Digital	PC/Stand-alone	Strategy	Single-player	N/I	English	N/I	N/I
30	Digital Logic and Electronics Concepts	Digital	PC/Stand-alone	Puzzle	Single-player	N/I	English	N/I	N/I
31	Digital System Game	Digital	PC/Stand-alone	Adventure	Single-player	75 min	English	N/I	N/I
32	DSAsketch	Non-Digital	Paper& Pencil	Guessing	Multi-groups	90min	English	N/I	N/I
33	EleMental: The Recurrence	Digital	PC/Stand-alone	RPG	Single-Player	N/I	English	N/I	N/I
34	ERPsim	Digital	PC/Online	Simulation	Multi-groups	N/I	English	Proprietary	N/I
35	Groupthink/Second Life	Digital	PC/Online	Simulation	Multi-players	N/I	English	N/I	N/I
36	hACME Game	Digital	PC/Online	Adventure	Single-players	N/I	English	N/I	N/I
37	Immersive Security Education Environment (I-	Digital	PC/Online	Simulation	Multi-player	N/I	English	Open Source	US\$0



	SEE)								
38	INNOV8	Digital	PC/Online	Simulation	Single-player	N/I	English	Proprietary	US\$0
39	IT Manager	Digital	PC/Online	Simulation	Single-player / Multi-player	N/I	English, Brazil Portuguese, China, Japan, Spain, Turkey, Russia, Germany, France, Italy, Netherlands, Poland, Sweden	Proprietary	US\$0
40	JV2M	Digital	PC/Stand-alone	RPG	Single-player	N/I	English	N/I	US\$0
41	Kid Krypto	Non-Digital	Paper&Pencil	Simulation	Multi-groups	30min	English, Polish, Slovenian	Creative Commons	N/I
42	Lego Factory	Non-Digital	Prop (Lego)	Puzzle	Multi-groups	N/I	English	N/I	N/I
43	Lego Tower Team Activity: Managing Change	Non-Digital	Prop (Lego)	Simulation	Multi-groups	60min	English	Creative Commons	N/I
44	Light-Bot	Digital	PC/Online	Puzzle	Single-Player	N/I	English	Free	US\$0
45	MIS Project Manager	Digital	PC/Online	Strategy	Multi-groups	N/I	English	N/I	N/I
46	Mission to Mars	Non-Digital	Board	Roll-and-move	Multi-groups	60 - 100 min	English	Creative commons By-NC-AS	US\$0
47	myPlanNet Simulation	Digital	PC/Stand-alone	Simulation	Single-Player	N/I	English	Free for registered on the site	N/I
48	Next Generation _7 helicopter game	Non-Digital	Paper&Pencil	Simulation	Multi-groups	N/I	English	N/I	N/I
49	.NET Terrarium 2.0	Digital	PC/Stand-alone	Puzzle	Single-player / Multi-player	N/I	English	Proprietary	US\$0
50	Operating	Non-	Paper&Penc	Simulation	Multi-groups	N/I	English	N/I	N/I

	systems role plays	Digital	il						
51	Paper Tower Competition	Non-Digital	Paper& Pencil	Simulation	Multi-groups	65 min	English and Brazilian Portuguese	Creative Commons	US\$2 per group
52	Path	Digital	PC/Online	Quiz	Single-player	N/I	English	N/I	N/I
53	PDCConsole	Digital	PC/Online	Simulation	Multi-groups	N/I	English	N/I	N/I
54	PlayScrum	Non-Digital	Card	Simulation	Multi-groups	N/I	English	N/I	N/I
55	PM Master	Non-Digital	Board	Quiz	Multi-groups	90min	English and Brazilian Portuguese	Proprietary	US\$49.99 per group
56	Problems and Programmers	Non-Digital	Card	Simulation	Multi-groups	N/I	English	Free	N/I
57	Process State Transition	Non-Digital	Board	Simulation	Multi-groups	N/I	English	N/I	N/I
58	Programming Fundamentals Mini-Game	Digital	PC/Online	Puzzle	Single-player	N/I	Spanish	N/I	US\$ 0
59	Project Execution Game (PEG)	Digital	PC/Stand-alone	Simulation	Multi-players	Several weeks	English	N/I	N/I
60	Project Risk	Non-Digital	Board	Simulation	Multi-groups	60min	English and Spanish	Proprietary	US\$200.00 per group
61	Project-o-poly (PoP)	Non-Digital	Board	Roll-and-move	Multi-groups	N/I	English	Proprietary	US\$0
62	Requirements Activity: Lego House	Non-Digital	Prop (Lego)	Simulation	Multi-groups	60min	English	Creative Commons	N/I
63	Requirements collection and analysis game (RCAG)	Digital	PC/Stand-alone	Simulation	Multi-groups	N/I	English	N/I	N/I
64	Requirements Game	Non-Digital	Paper& Pencil	Simulation	Multi-groups	90-120min	English	N/I	N/I

65	Robot Trouble	Digital	PC/Online	Puzzle	Single-Player	N/I	English	Beta version free	N/I
66	Robozzle	Digital	PC/Online	Puzzle	Single-Player	N/I	English	Free	N/I
67	Saving Princess Sera	Digital	PC/Stand-alone	RPG	Single-Player	20 min	English	Free	N/I
68	Scrum Lego City	Non-Digital	Prop (Lego)	Simulation	Multi-groups	120min	English	Creative Commons	N/I
69	SDM - Software Development Manager	Digital	PC/Stand-alone	Simulation	Single-player	30min	English	N/I	N/I
70	Secret Ninja Testing	Digital	PC/Online	Adventure	Multi-players	N/I	English	N/I	N/I
71	Secure Volunteer Game	Digital	PC/Online	Simulation	Single-Player	N/I	English	Proprietary	US\$ 0
72	Security Protocol Game	Non-Digital	Paper& Pencil	Simulation	Multi-groups	120min	English	Proprietary	US\$0
73	SESAM	Digital	PC/Stand-alone	Simulation	Single-Player	N/I	German	N/I	N/I
74	SharkWord	Digital	PC/Online	Simulation	Single-Player	5 days	English and Spanish	N/I	US\$195.13
75	SimjavaSP	Digital	PC/Stand-alone	Simulation	Single-Player	N/I	English	N/I	N/I
76	SimSE	Digital	PC/Stand-alone	Simulation	Single-Player	N/I	English	Free	US\$0
77	Simsoft Game	Digital/Non-Digital	PC/Stand-alone/Board	Simulation	Multi-groups	N/I	English	N/I	N/I
78	SimSYS	Digital	PC/Stand-alone	Simulation	Single-player	N/I	English	N/I	N/I
79	Simulate Computer Game	Non-Digital	Prop	Simulation	Multi-groups	N/I	English	N/I	N/I
80	SimulES	Non-Digital	Board	Simulation	Multi-groups	N/I	Brazilian Portuguese	Free	US\$5 per group
81	SimulES-W	Digital	PC/Online	Simulation	Multi-players	N/I	Brazilian Portuguese	N/I	N/I
82	SimVBSE	Digital	PC/Stand-	Simulation	Single-player	N/I	English	N/I	N/I

			alone						
83	Sorting Algorithms	Digital	PC/Stand-alone	Puzzle	Single-Player	N/I	English	N/I	N/I
84	SortingCasino	Non-Digital	Card	Action	Multi-groups	N/I	English	N/I	N/I
85	SortingGame	Non-Digital	Cards	Action	Multi-groups	N/I	English	N/I	N/I
86	Starter MMO	Digital	PC/Online	Adventure	Single-player / Multi-player	N/I	English	N/I	N/I
87	Subnet Game	Digital	PC/Online	Adventure	Single-player	N/I	English	Proprietary	US\$0
88	Subnet Troubleshooting Game	Digital	PC/Online	Adventure	Single-player	N/I	English	Proprietary	N/I
89	The Catacombs	Digital	PC/Stand-alone	RPG	Single-Player	20 min	English	N/I	N/I
90	The Hard Choices	Non-Digital	Board	Roll-and-move	Multi-groups	60min (playing 2 rounds)	English	Creative Commons	US\$5 per group
91	The Incredible Manager	Digital	PC/Stand-alone	Simulation	Single-Player	50-140 min	English	N/I	N/I
92	The MIS Game	Non-Digital	Board	Simulation	Multi-groups	N/I	English	N/I	N/I
93	The Mystery of Traffic Lights	Digital	PC/Online	Adventure	Multi-player	N/I	English	Proprietary	N/I
94	The Orange game	Non-Digital	Prop (Oranges)	Simulation	Multi-groups	N/I	Brazilian Portuguese, English, French, Greek, Hungarian, Italian, Polish, Slovenian	Creative Commons	N/I
95	The Software Development Game	Non-Digital	Paper& Pencil	Simulation	Multi-groups	N/I	English	N/I	N/I
96	The Waterfall	Non-	Board	Roll-and-	Multi-groups	N/I	English	Free	US\$5 per

	Game	Digital		move					group
97	Tower of Cubes	Digital	PC/Stand-alone	Puzzle	Single-Player	30 min	English	N/I	N/I
98	Treasure Hunt	Digital	PC/Online	Puzzle	Single-player	30 min	English	Proprietary	US\$0
99	TREEZ	Digital	PC/Stand-alone	Action	Single-player / Multi-players	N/I	English	N/I	N/I
100	Software Risk Management Game	Non-Digital	Board	Simulation	Multi-groups	N/I	English	N/I	N/I
101	U-Test	Digital	PC/Online	Adventure	Single-Player	N/I	Brazilian Portuguese	N/I	N/I
102	ViRPlay3D2	Digital	PC/Stand-alone	RPG	Single-Player / Multi-Players	N/I	English	N/I	N/I
103	Welcome to SCRUMIA	Non-Digital	Paper& Pencil	Simulation	Multi-groups	60 min	English and Brazilian Portuguese	Creative Commons	N/I
104	Wireless Explorer	Digital	PC/Online	Adventure	Single-Player	N/I	English	Proprietary	US\$0
105	Wu's Castle	Digital	PC/Stand-alone	RPG	Single-Player	20-30 min	English	Free	N/I
106	X-MED	Digital	PC/Stand-alone	Simulation	Single-Player	120 min	Brazilian Portuguese	Creative Commons	US\$0
107	Z-Buffer	Digital	PC/Stand-alone	Action	Single-Player	N/I	English	N/I	N/I

## APÊNDICE D – Forma de desenvolvimento dos jogos selecionados na revisão sistemática da literatura.

N.	Name	Game design	Game developed	Evaluation level	Research design	Data collection instrument(s)
1	3DAR Lego Game	N/I	Desktop marker based AR system using the library ARtoolkit.	Reaction	Evaluated effectiveness and ease of play.	Collecting feedback after the game execution through questionnaire.
2	Age of Computers	N/I	N/I	Reaction	Evaluating effectiveness, degree of difficulty, preference of game as instructional strategy, through a case study using a one-shot post-test only (X O) design. In addition, the game has been evaluated in terms of effectiveness through an experiment using a randomized pretest - posttest control group design.	Data has been collected via questionnaire and game logs.
3	Algorithms Recursive Game	N/I	Adopted the generic Snakes and Ladders shell of Simulation Advanced Game Environment.	Reaction	Evaluated the ease of use of the game platform, learning effectiveness and the students' motivation through a case study using an one-shot post-test only (X O) design.	Data has been collected via questionnaire and interviews.
4	AMEISE	N/I	N/I	N/I	N/I	N/I
5	Anti-Phishing Phil	Developed using Learning Science Principles (LSP). It has been developed in an iterative approach creating prototypes and play-testing, both on the content of the	Anti-Phishing Phil is a SCORM-compliant computer security training game that can stand alone or be integrated into a Learning Management System (LMS).	Learning	Evaluated effectiveness, relation to demographics, and user feedback through an experiment using a Randomized pretest - posttest control group design.	Data has been collected through tests and questionnaire.

		game (what to teach) and the game design itself (presentation).				
6	Battleships	N/I	N/I	N/I	N/I	N/I
7	BattleThre ads	N/I	N/I	Reaction	Evaluated effectiveness.	The collecting of the feedback was performed with application of questionnaires after game.
8	Binary Game	N/I	N/I	N/I	N/I	N/I
9	BINX	N/I	XNA framework was used for the implementation.	Learning	Evaluated effectiveness.	Game was accomplished using questionnaires, evaluations and students made pre-assessments and post-assessments.
10	Bomberm- an Game	N/I	N/I	N/I	N/I	N/I
11	BOTS	N/I	N/I	Reaction	Evaluated effectiveness, enjoyment and clarity of visualization.	Collecting feedback after the game testing.
12	CARGO- BOT	N/I	N/I	N/I	N/I	N/I
13	CEEBOT	N/I	N/I	N/I	N/I	N/I
14	C-Jump	N/I	N/I	Reaction	Evaluating playability, rules, learning value, fun factor and aesthetic appearance of the game through a case study using a one-shot post-test only (X O) design.	Data has been collected via observation and questionnaire and expert interviews capturing the

						student's and instructor's viewpoint.
15	COLOBOT	N/I	N/I	N/I	N/I	N/I
16	Computer Architecture Mini-Game	Based on several design factors for developing educational games.	N/I	Reaction	Evaluated mainly aspects of the game (such as, hints) as well as subjectively also effectiveness through a case study with a one-shot post-test only design.	Data has been collected via questionnaire.
17	Computer Theory Jeopardy Game	N/I	N/I	N/I	N/I	N/I
18	Computing Networks Mini-Game	Based on several design factors for developing educational games.	N/I	Reaction	Evaluated mainly aspects of the game (such as, hints) as well as subjectively also effectiveness through a case study with a one-shot post-test only design.	Data has been collected via questionnaire.
19	Conquer the Net	N/I	N/I	N/I	N/I	N/I
20	Control-Alt-Hack	N/I	N/I	N/I	N/I	N/I
21	CounterMeasures	N/I	N/I	Learning	Evaluated effectiveness, study time and motivation through an experiment using a randomized pretest - posttest control group design.	Data has been collected through tests, game logs and questionnaire.
22	CyberCIEG E	N/I	N/I	Learning	Evaluating effectiveness, enjoyment and alignment with learning objectives through several quasi-experimental study using a static group pre-test – post-test design.	Data has been collected through tests and questionnaire.
23	Database concurrency control	N/I	N/I	Reaction	N/I	Were performed tests and questions in



	card game					classroom with students. And the professor had the support of a specialist in Psychology in Education.
24	Databases: Lots of Data and Getting Quick Results	N/I	N/I	N/I	N/I	N/I
25	Dealing with difficult people	N/I	N/I	N/I	Evaluating motivation, user experience and learning through a series of case studies (a one-shot post-test only (X O) design).	Was been using the MEEGA[Wangenheim et al. 2012] in two courses at a university.
26	Deliver!	N/I	N/I	Reaction	N/I	N/I
27	DesigMPS	Based on concept map principles.	N/I	Learning	Evaluated effectiveness, enjoyment, appropriateness, and strengths & weaknesses through an experiment using a randomized pretest posttest design.	Data has been collected via tests and questionnaire.
28	Detective Game – what killed the project?	N/I	N/I	Reaction	Evaluating motivation, user experience and learning through a series of case studies (a one-shot post-test only (X O) design) using the MEEGA[Wangenheim et al. 2012] model in two courses at a university.	N/I
29	Digi Island	N/I	Developed with XNA Game Studio 3.1.	N/I	N/I	N/I
30	Digital Logic and	N/I	Authored using the YOYO Games GameMaker game	N/I	N/I	N/I

	Electronics Concepts		authoring platform.			
31	Digital System Game	N/I	Developed in C# using the Microsoft XNA framework. Development and testing was done on PCs	Reaction and Learning	Evaluated effectiveness, engagement and usability mainly through a case-study using an one-shot post-test only design. In addition, some participants participated in pre- and post-test realizing a study in an one-shot pre-test – post-test design.	Collecting data via questionnaire and interviews.
32	DSAsketch	N/I	N/I	N/I	N/I	N/I
33	EleMental: The Recurrence	N/I	Developed using DarkWynter engine's code engine, which is coded using XNA's Game Studio 2.0, and integration with the C#'s Code Dom.	Learning	Evaluating effectiveness, enjoyment and preference as instructional strategy and to collected feedback through a case study using a one-shot pre-test – post-test (O X O) design.	Data has been collected through tests, game logs and questionnaires.
34	ERPsim	N/I	N/I	N/I	N/I	N/I
35	Groupthink /Second Life	N/I	N/I	Reaction	Evaluated effectiveness, effects of Second Life and strengths and improvement opportunities through a case study using a one-shot post-test only (X O) design.	Data has been collected via questionnaire.
36	hACME Game	N/I	N/I	N/I	Game logs have been evaluated in terms of success rate and time spent on challenges.	N/I
37	Immersive Security Education Environment (I-SEE)	N/I	Developed in OpenSim, 3-D simulator open source.	N/I	N/I	N/I
38	INNOV8	N/I	N/I	N/I	N/I	N/I
39	IT Manager	N/I	N/I	N/I	N/I	N/I
40	JV2M	N/I	Follows the model of linear progression borrowed from	N/I	N/I	N/I

			classic videogames.			
41	Kid Krypto	N/I	N/I	N/I	N/I	N/I
42	Lego Factory	N/I	N/I	Reaction	Evaluated effectiveness.	Collecting feedback after the game execution through questionnaires.
43	Lego Tower Team Activity: Managing Change	N/I	N/I	N/I	N/I	N/I
44	Light-Bot	N/I	N/I	N/I	N/I	N/I
45	MIS Project Manager (Formerly known as the Information Systems Project Manager Game)	N/I	N/I	N/I	N/I	N/I
46	Mission to mars - Release or iteration planning	N/I	N/I	N/I	Evaluated usability, tuning game parameters, scaling down to fit in 90 minutes.	
47	myPlanNet Simulation	N/I	N/I	N/I	N/I	N/I
48	Next Generation _7 helicopter	N/I	N/I	N/I	N/I	N/I

	game					
49	.NET Terrarium 2.0	N/I	Developers to learn about the new .NET programming model and language	N/I	N/I	N/I
50	Operating systems role plays	N/I	N/I	Reaction	Evaluated effectiveness.	Collecting feedback after the game execution.
51	Paper Tower Competitio n	N/I	N/I	N/I	N/I	N/I
52	Path	N/I	N/I	Reaction	Evaluating effectiveness, preference of game as instructional strategy, motivation and rate of submission of assignments within deadline, through a case study with an one-shot post- test only (X O) design. Informally results have been compared with earlier courses.	Data has been collected through questionnaires (at two different moments during the usage of the game).
53	PDConsole	N/I	N/I	Reaction	Evaluated effectiveness and engagement.	Collecting feedback through observation during the game execution.
54	PlayScrum	N/I	N/I	Reaction	Enjoyment, effectiveness and the game's strength and weaknesses have been evaluated through a case study using an one-shot post-test only (X O) design.	Data has been collected via questionnaire.
55	PM Master	N/I	N/I	N/I	N/I	N/I
56	Problems and Programm ers	N/I	N/I	Reaction	Effectiveness (reinforce and/or acquire new knowledge) and enjoyability have been evaluated through a case study with one-shot post-test only (X O) design.	Data has been collected via questionnaires.

57	Process State Transition	N/I	N/I	Reaction	Evaluated effectiveness.	Collecting feedback after the game executing through questionnaires.
58	Programming Fundamentals Mini-Game	Based on several design factors for developing educational games.	N/I	Reaction	Evaluated mainly aspects of the game (such as, hints) as well as subjectively also effectiveness through a case study with a one-shot post-test only design.	Data has been collected via questionnaire.
59	Project Execution Game (PEG)	N/I	N/I	Learning	The game has been evaluated in terms of effectiveness, degree of difficulty, challenge, usefulness, through two case studies using a One-shot pre-test – post-test design.	Data has been collected through questionnaires (auto-evaluation of learning through the students).
60	Project Risk	N/I	N/I	N/I	N/I	N/I
61	Project-opoly (PoP)	N/I	N/I	N/I	N/I	N/I
62	Requirements Activity: Lego House	N/I	N/I	N/I	N/I	N/I
63	Requirements collection and analysis game (RCAG)	N/I	N/I	Learning	The game has been evaluated in terms of effectiveness, usefulness, engagement as well as strengths through an experiment using a randomized pretest - posttest control group design.	Data has been collected through tests and questionnaire.
64	Requirements Game	N/I	N/I	N/I	N/I	N/I
65	Robot	N/I	N/I	N/I	N/I	N/I

	Trouble					
66	Robozzle	N/I	N/I	N/I	N/I	N/I
67	Saving Princess Sera	N/I	Developed using RPG Maker.	Learning	To evaluate effectiveness, engagement and motivation and to collect feedback a case study has been conducted using a one-shot pre-test – post-test design.	Data has been collected via tests, questionnaire and interviews.
68	Scrum Lego	N/I	N/I	N/I	N/I	N/I
69	SDM - Software Development Manager	N/I	N/I	Reaction	Evaluated effectiveness, and engagement through a case study using an one-shot post-test only (X O) design.	Collecting data via a short questionnaire.
70	Secret Ninja Testing	N/I	N/I	N/I	N/I	N/I
71	Secure Volunteer Game	N/I	N/I	N/I	N/I	N/I
72	Security Protocol Game	N/I	N/I	Reaction	Evaluated effectiveness, enjoyment and strengths through a case study using a one-shot post-test only (X O) design.	Data has been collected via questionnaire.
73	SESAM	N/I	N/I	Reaction and Learning	Effectiveness has been evaluated in different ways: one experiment has been conducted using a Randomized pre-test – post-test design and two case studies (using a one-shot pre-test post-test and one with a two-shot pre-test post-test design) have been run.	Data has been collected via questionnaires, tests and game scores and the analysis of the developed project plans.
74	SharkWord	N/I	N/I	N/I		N/I
75	SimjavaSP	N/I	N/I	Reaction	Evaluating effectiveness, enjoyability and ease of play through a case study	Data has been collected via

					with an one-shot post-test only (X O) design.	questionnaires.
76	SimSE	N/I	N/I	Reaction and Learning	Has been evaluated with respect to different aspects in several studies, including a series of case studies using a one-shot post-test only (X O) to evaluate enjoyability, effectiveness (reinforce and new knowledge) as well as strengths and weaknesses of the game. Furthermore a comparative study has been conducted through an experiment using a randomized pretest - posttest control group design.	Data has been collected via questionnaires and tests.
77	Simssoft Game	N/I	N/I	Reaction	Evaluation of learning principles and levels by conducting a series of case studies with an one-shot post-test only (X O) design.	Data has been collected from the game database, pre- and post-game surveys, interviews with the players, researcher memos, and field notes.
78	SimSYS	N/I	N/I	N/I	N/I	N/I
79	Simulate Computer Game	N/I	N/I	N/I	N/I	N/I
80	SimulES	N/I	N/I	N/I	N/I	N/I
81	SimulES-W	N/I	N/I	N/I	N/I	N/I
82	SimVBSE	N/I	N/I	N/I	N/I	N/I
83	Sorting Algorithms	N/I	N/I	N/I	N/I	N/I
84	SortingCas	N/I	N/I	Reaction	Evaluating effectiveness, usefulness	Data has been

	ino				and motivation as well as to collect feedback through a case study using a one-shot post-test only (X O) design.	collected through questionnaire and a mapping to exam results.
85	SortingGame	N/I	N/I	Reaction	Evaluating effectiveness, usefulness and motivation as well as to collect feedback through a case study using a one-shot post-test only (X O) design.	Data has been collected through questionnaire and a mapping to exam results.
86	Starter MMO	N/I	N/I	Reaction	Evaluation of effectiveness, enjoyment and usefulness in a case-study using a one-shot post-test only (X O) design.	Data has been collected via questionnaire.
87	Subnet Game	N/I	N/I	N/I	N/I	N/I
88	Subnet Troubleshooting Game	N/I	N/I	N/I	N/I	N/I
89	The Catacombs	N/I	Developed using the BioWare Aurora toolset for NeverWinter Nights (NWN).T87	Learning	To evaluate effectiveness, engagement and motivation and to collect feedback a case study has been conducted using a one-shot pre-test – post-test design.	Data has been collected via tests, questionnaire and interviews.
90	The Hard Choices	N/I	N/I	N/I	N/I	N/I
91	The Incredible Manager	N/I	N/I	Reaction	Evaluating effectiveness, enjoyability, interest in project management and strengths and improvement opportunities in a case study with professionals, undergraduates and graduate students using a one-shot post-test only (X O) design.	Data has been collected via questionnaires.
92	The MIS Game	N/I	N/I	Reaction	Evaluated effectiveness.	Collecting feedback after game executions



						through questionnaires.
93	The Mystery of Traffic Lights	N/I	N/I	Learning	Evaluated effectiveness and interest/motivation in learning through games. A quasi-experimental study has been conducted using a static group pre-test – post-test design.	Data has been collected through questionnaire and focus group interviews.
94	The Orange game	N/I	N/I	N/I	N/I	N/I
95	The Software Development Game	N/I	N/I	Reaction	Evaluating what the students learned from the game, what they need to win and how the game needs to be modified a series of nine case studies has been conducted using a one-shot post-test only (X O) design.	Data has been collected via questionnaires.
96	The Waterfall Game	N/I	N/I	N/I	N/I	N/I
97	Tower of Cubes	N/I	N/I	N/I	Evaluated effectiveness, enjoyment and ease of understanding of the concepts presented by the game through a case study using an one-shot post-test only (X O) design.	Data has been collected via questionnaire.
98	Treasure Hunt	N/I	N/I	N/I	N/I	N/I
99	TREEZ	N/I	N/I	N/I	N/I	N/I
100	Using Games in Software Engineering Education to	N/I	N/I	Reaction	Evaluating effectiveness, enjoyability, realism and simplicity through a series of case studies using a one-shot post-test only (X O) design in three different courses.	Data has been collected via questionnaires.

	Teach Risk Management					
101	U-Test	Developed following ADDIE.	N/I	Learning and Reaction	Effectiveness, appropriateness and engagement have been evaluated in two experiments using a randomized pretest - posttest control group design.	Data has been collected from tests and questionnaires.
102	ViRPlay3D 2	N/I	N/I	N/I	Evaluated usefulness, enjoyment and clarity of game objects through a case study using an one-shot post-test only design and collecting data via questionnaire. Effectiveness has been evaluated through an experiment using a randomized pretest - posttest control group and collecting data through tests.	
103	Welcome to SCRUMIA	Following ISD - Instructional System Design and based on teaching experiences.		N/I	N/I	N/I
104	Wireless Explorer	N/I	N/I	N/I	N/I	N/I
105	Wu's Castle	N/I	Developed using RPG Maker XP.	Learning	Effectiveness and subjective aspects, such enjoyability, motivation, degree of difficulty and preference of such kind of instructional strategy have been evaluated in experimental studies using a randomized pretest - posttest control group design with crossover.	Data has been collected through tests and questionnaire.
106	X-MED	N/I	N/I	Learning	Evaluating learning effectiveness, appropriateness, engagement and strengths & weaknesses through a series of experiments was conducted	Data has been collected through tests and questionnaires.

					in parallel in three master courses in Brazil using randomized pretest - posttest control group design.	
107	Z-Buffer	Based on a concept map created for this instructional unit.	N/I	N/I	N/I	N/I

## APÊNDICE E – Fases e atividades do processo ENgAGED.

Dados extraídos do relatório técnico: “ENgAGED: Processo de Desenvolvimento de Jogos para Ensino em Computação” (BATTISTELLA e WANGENHEIM, 2015).

### F1. ANÁLISE DA UNIDADE INSTRUCIONAL

Atividade	A1.1 Especificar unidade instrucional do jogo
<b>Descrição</b>	<p>Nesta atividade especifica-se a <b>unidade instrucional (UI)</b>. Uma UI é o conjunto de atividades organizadas em torno de tópicos ou temas principais da disciplina. As atividades podem ser lições, exercícios, jogos, aulas expositivas ou vídeos criados ou utilizados com objetivo de ensinar conteúdos instrucionais (HURWITZ e DAY, 2007, WISCONSIN, 2012).</p> <p>O objetivo desta atividade é especificar as informações que caracterizam a unidade instrucional (UI). No contexto do processo ENgAGED, a especificação da UI é exclusivamente para o ensino via jogo. Portanto, outras aulas (p. ex. ensino de conceitos teóricos por aulas-expositivas antes da aplicação do jogo) não são consideradas neste processo.</p>
<b>Premissa do jogo desenvolvido</b>	<ul style="list-style-type: none"> <li>• <i>Não deve ser obsceno, ofensivo, odioso ou violento.</i></li> <li>• <i>Não deve ter discriminação com base em raça, sexo, religião, nacionalidade, deficiência, orientação sexual ou idade.</i></li> <li>• <i>Não deve infringir qualquer direito autoral, respeitando a propriedade intelectual de qualquer pessoa.</i></li> <li>• <i>Não deve violar qualquer direito à privacidade de pessoas.</i></li> <li>• <i>Não deve violar qualquer direito a um terceiro, como um contrato ou acordo.</i></li> <li>• <i>Não deve desacatar as leis vigentes no país.</i></li> <li>• <i>Não deve molestar, chatear, constranger ou irritar qualquer pessoa.</i></li> <li>• <i>Não deve ser ilegal ou incitar qualquer pessoa a cometer atos ilegais.</i></li> </ul>

	<ul style="list-style-type: none"><li>• <i>Não deve conter publicações comerciais.</i></li></ul>
<b>Tarefa</b>	<p><b>1.1 Especificar informações gerais da disciplina</b></p> <p>Identificar o nome da universidade, o nome do curso, o nome da disciplina, o objetivo geral da disciplina, os objetivos específicos, quantidade de alunos na disciplina e pré-requisitos da disciplina. Estas informações são encontradas tipicamente no plano de ensino da disciplina e/ou na secretaria do curso.</p> <p><b>1.2 Especificar informações exclusivas da unidade instrucional</b></p> <p>Identificar o conteúdo programático da unidade instrucional, específicos para o jogo. O conteúdo programático é o conteúdo que será abordado na unidade instrucional e pode ser levantado a partir do plano de ensino.</p> <p>Outra informação necessária para especificar a unidade instrucional, é o módulo do jogo. A especificação deste elemento é feita por meio de um diagrama contendo as aulas antes e depois da aplicação do jogo. Por exemplo, informar que antes da aplicação do jogo é realizada uma aula-expositiva. A figura abaixo apresenta um exemplo do jogo aplicado na Disciplina de Gerência chamado SCRUM'ed (SCHNEIDER, 2015).</p>

	<pre> graph TD     A[Ganhar a atenção dos alunos] --&gt; B[Declarar objetivos]     B --&gt; C[Relembrar apreendido anterior]     C --&gt; D     subgraph D [Apresentar novas informações]         D1[Introdução Geral] --&gt; D2[Papéis]         D2 --&gt; D3[Cerimônias e Artefatos]     end     D --&gt; E     subgraph E [SCRUMed]         E1[Orientar os alunos] --&gt; E2[Elicitar desempenho]         E2 --&gt; E3[Fornecer feedback]     end     E --&gt; F     subgraph F [ ]         F1[Avaliar desempenho] --&gt; F2[Reforçar a retenção]     end     </pre> <p><b>1.3 Especificar informação do jogo</b></p> <p>Nesta última tarefa da atividade, identifica-se as restrições para aplicação do jogo. Tipicamente a restrição refere-se ao tempo de duração da aula em que o jogo será aplicado.</p>	
<p><b>Produtos de trabalho</b></p>	<p><b>Produto de entrada:</b> P1.1.1 -Plano de ensino da disciplina</p>	
	<p><b>Produto de saída:</b> P1.1.2 - Especificar informações gerais da disciplina</p>	
	<table border="1" style="width: 100%;"> <tr> <td style="width: 50%;">Universidade</td> <td style="width: 50%;">Nome da universidade</td> </tr> </table>	Universidade
Universidade	Nome da universidade	

Curso	Nome do curso
Disciplina	Código e nome da disciplina
Objetivo geral da disciplina	Objetivo geral da disciplina
Quantidade de alunos na disciplina	Quantidade de alunos participantes da disciplina
Pré-requisito(s) da disciplina	Pré-requisito da disciplina
<b>Produto de saída:</b> P1.1.3 - Especificar informações exclusivas da unidade instrucional	
Conteúdo programático	Conteúdo programático a ser abordado na unidade instrucional normalmente é retirado do plano de ensino.
Módulo do jogo	Módulo do jogo informa-se as aulas antes e depois da aplicação do jogo, como apresentado na figura 8.
<b>Produto de saída:</b> P1.1.4 - Especificar informação do jogo	
Restrições	Restrições do jogo, tipicamente representado pelo tempo de duração da aula que o jogo é aplicado.

<b>Atividade</b>	<b>A1.2 Caracterizar aprendizes</b>
<b>Descrição</b>	Nesta atividade o objetivo é identificar o público-alvo (aprendizes) da unidade instrucional (DICK e CAREY, 1996).
<b>Tarefa</b>	<b>1.2.1 Caracterizar aprendizes</b>  Nesta tarefa caracteriza-se o público-alvo (alunos da disciplina). Tipicamente a caracterização inclui o levantamento das seguintes informações: a faixa etária, preferências de gênero de jogos, a preferência

de plataformas de jogo, modo de interação dos jogos, a frequência em que os alunos jogam jogos digitais e não-digitais, jogos favoritos, os cenários preferidos de jogos.

O levantamento de dados para realizar a caracterização dos aprendizes pode ser feita, p. ex., por *survey* com os alunos do curso de computação. O **PI.2.1 – Questionário de caracterização dos aprendizes e contexto** apresenta um questionário que pode ser utilizado para auxiliar na caracterização dos aprendizes.

Caso deseje coletar dados de outro modo, é possível criar um questionário impresso ou em formulário eletrônico. No caso de formulário eletrônico, é possível utilizar o *Google Forms* (<https://docs.google.com/forms>). A vantagem de usar formulários eletrônicos é que os dados já são arquivados digitalmente, tornando a análise de dados mais simples em termos de organização dos dados.

Outras formas de levantamento de dados incluem consultas a literaturas existentes (p. ex. BATTISTELLA e WANGENHEIM, 2015) ou entrevistas com alunos.

Para ilustrar melhor a caracterização dos aprendizes, pode-se utilizar a técnica de Personas (COOPER, 2004). Personas são personagens fictícios, ilusórios ou ilustrativos produzidos para representar o público-alvo. Uma persona pode ser descrita por meio de uma narrativa.

A narrativa pode iniciar contextualizando a pessoa com nome fictício, foto, idade e gênero. Em seguida pode ser descrito os hábitos do aluno em termos de jogos favoritos, cenários de interesse, modo de interação (*singleplayer* ou *multiplayer*), quantidade de horas jogadas por dia, semanas ou meses. É interessante também citar atividades realizadas pela persona, que são relacionadas ao contexto do jogo. P. ex. quando o aluno assiste filmes de ficção científica e gosta de jogos com o mesmo cenário. Ou então quando o aluno gosta de algum esporte e gosta de jogar jogos que são relacionados ao esporte que pratica.



<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P1.2.1 - Questionário de caracterização dos aprendizes	
	<b>Aprendizes</b>	
	Qual curso você está cursando ou cursou?	<ul style="list-style-type: none"> <li>• Bacharelado em Ciência da Computação</li> <li>• Bacharelado em Sistemas de Informação</li> <li>• Bacharelado em Engenharia da Computação</li> <li>• Bacharelado em Engenharia de Software</li> <li>• Bacharelado em Informática</li> <li>• Licenciatura em Computação ou Informática</li> <li>• Tecnologia em Informática</li> <li>• Pós-graduação em Computação</li> <li>• Outro</li> </ul>
	Qual a sua faixa etária?	<ul style="list-style-type: none"> <li>• Menor de 15 anos</li> <li>• Dos 15 aos 18 anos</li> <li>• Dos 19 aos 22 anos</li> <li>• Dos 23 aos 25 anos</li> <li>• Dos 26 aos 30 anos</li> <li>• Dos 31 aos 40 anos</li> <li>• Maior de 40 anos</li> </ul>
Qual é o seu gênero?	<ul style="list-style-type: none"> <li>• Masculino</li> <li>• Feminino</li> </ul>	

Qual é a sua preferência de gêneros de jogo?

- **Ação-aventura:** Um jogo que exigem que os jogadores utilizem reflexos rápidos, com precisão para superar obstáculos, resolver desafios ou para responder perguntas.
- **Adivinhação:** Um jogo em que o objetivo é identificar algum tipo de informação, como uma palavra, a partir de desenhos ou imitação de outros jogadores.
- **Corrida:** Um jogo em que o jogador comanda um veículo ou participa de uma corrida, tentando se mover mais rápido do que os oponentes para alcançar um objetivo específico ou ser mais rápido que um determinado tempo.
- **Estratégia:** Um jogo que é focado na jogabilidade e exige pensamentos táticos e cuidados, a fim de conseguir a vitória.
- **Puzzle:** Um jogo que envolve personagens que controlam um conjunto de comandos, para navegar através de labirintos ou para organizar os objetos de maneira correta.
- **Quiz:** Um jogo onde o jogador precisa responder perguntas para uma determinada área de conhecimento.
- **Role-playing game (RPG):** Um jogo onde o jogador controla ações de um protagonista e com este personagem vive imerso em um mundo fictício. Em jogo de *role-playing* os personagens interagem com este mundo e ficam mais fortes.
- **Roll-and-move:** Jogos de tabuleiro em que as fichas são movidas com base nos resultados mostrados em um ou mais dados (por exemplo: *snake and ladders* e *monopoly*).
- **Simulação:** Um jogo desenvolvido para colocar o jogador no controle de um determinado ambiente ou atividade, o qual busca ser o mais realista possível.
- **Outro**

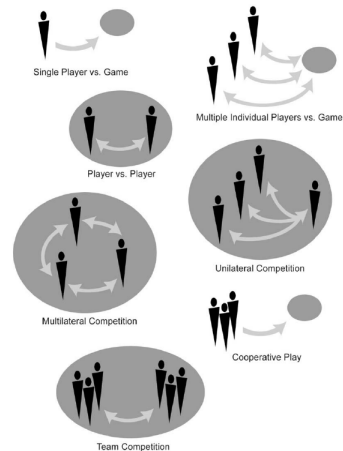
(HERZ, 1997; ADAMS e ROLLINGS, 2006; SCHIFFLER, 2006; WOLF, 2001)

	<p>Você tipicamente prefere jogar jogos em qual plataforma?</p>	<ul style="list-style-type: none"><li>• Jogo digital de computador stand-alone p. ex. computador pessoal sem acesso a internet.</li><li>• Jogo digital de computador online p. ex. computador pessoal com acesso a internet.</li><li>• Jogo digital de console p. ex. XBox 360, Playstation, Wii - Nintendo.</li><li>• Jogo digital <i>mobile</i> - <i>smartphone</i> p. ex. iPhone, Samsung Galaxy.</li><li>• Jogo digital <i>mobile</i> - <i>tablet</i> p. ex. iPad, Samsung Galaxy Note.</li><li>• Jogo não-digital de tabuleiro.</li><li>• Jogo não-digital de cartas.</li><li>• Jogo não-digital papel &amp; lápis (para ser jogado apenas com papel e lápis.) p. ex. jogo imagem e ação, "caça-palavras".</li><li>• Jogo não-digital acessórios (para ser jogado com acessórios ou adereços (objetos portáteis)) p. ex. Lego.</li><li>• Outro</li></ul> <p>(CONNOLLY et al., 2012; CAULFIELD, VEAL e MAJ, 2011)</p>
--	---	---

Qual é o modo de interação preferido de jogo?

- Single-player: você joga sozinho, competindo contra si mesmo ou o computador.
- Multiplayers competitivo: vários jogadores competem entre si.
- Multiplayers cooperativo: vários jogadores cooperam para vencer o jogo em conjunto.
- Multigrupos competitivo: grupos competem entre si.
- Multigrupos cooperativos: grupos reunidos de forma cooperativa.
- Outro

Modo de interação definidas com base em Fullerton (2008).



Fonte: FULLERTON, 2008.

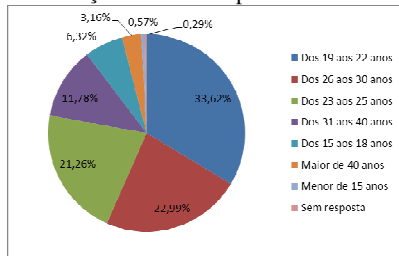
Com que frequência você joga jogos digitais?	<p><b>Nunca:</b> nunca joga.</p> <p><b>Raramente:</b> jogo de tempos em tempos.</p> <p><b>Mensalmente:</b> jogo pelo menos uma vez por mês.</p> <p><b>Semanalmente:</b> jogo pelo menos uma vez por semana.</p> <p><b>Diariamente:</b> jogo todos os dias.</p>
Com que frequência você joga jogos não-digitais?	<p><b>Nunca:</b> nunca joga.</p> <p><b>Raramente:</b> jogo de tempos em tempos.</p> <p><b>Mensalmente:</b> jogo pelo menos uma vez por mês.</p> <p><b>Semanalmente:</b> jogo pelo menos uma vez por semana.</p> <p><b>Diariamente:</b> jogo todos os dias.</p>
Liste os nomes dos seus três jogos favoritos?	Listagem dos 3 principais jogos favoritos.
Quais são os seus cenários de jogos favoritos?	<ul style="list-style-type: none"> <li>• <b>Esporte:</b> cenários de esporte, como corridas, futebol, vôlei, basquete, natação, tênis.</li> <li>• <b>Medieval:</b> cenários que envolvem cenários antigos contendo reis, rainhas, princesas, magos, castelos.</li> <li>• <b>Dungeons&amp;Dragons:</b> cenários mitológicos ou antigos envolvendo monstros, dragões, fantasmas, bárbaros.</li> <li>• <b>Ficção científica:</b> cenários com tecnologias avançadas e naves espaciais futuristas.</li> <li>• <b>Animais:</b> cenários que envolvem o cuidado e atenção aos animais, como cuidar de uma fazenda.</li> <li>• <b>Outro</b></li> </ul> <p>(ARMOR GAMES, 2014; CLICK JOGOS, 2014; DONJON, 2014)</p>
<b>Contexto</b>	

	Quais dispositivos eletrônicos você utiliza em geral, não apenas para jogo?	<ul style="list-style-type: none"> <li>• Computador</li> <li>• <i>Tablet</i></li> <li>• <i>Smartphone, celular</i></li> <li>• Outro</li> </ul>
	Onde você joga jogos?	<ul style="list-style-type: none"> <li>• Casa</li> <li>• <i>LAN house</i></li> <li>• Universidade</li> <li>• Outro</li> </ul>
	Qual sistema operacional para computador, você utiliza em casa?	<ul style="list-style-type: none"> <li>• Windows, versão:___</li> <li>• MacOS, versão:___</li> <li>• Linux, distribuição:___</li> <li>• <i>Android</i>, versão:___</li> <li>• Outro</li> </ul>
	Qual é a velocidade de banda da sua conexão de internet em sua casa?	<ul style="list-style-type: none"> <li>• 1 Mb</li> <li>• 2 Mb</li> <li>• 4 Mb</li> <li>• 5 Mb</li> <li>• 8 Mb</li> <li>• 10 Mb</li> <li>• 20 Mb</li> <li>• 30 Mb</li> <li>• 50 Mb</li> <li>• 100 Mb</li> <li>• Outro</li> </ul>

	Quais navegadores de internet você mais usa?	<ul style="list-style-type: none"> <li>• Internet Explorer</li> <li>• Firefox</li> <li>• Chrome</li> <li>• Safari</li> <li>• Opera</li> <li>• Outro</li> </ul>
	Qual a resolução do monitor que você utiliza?	<ul style="list-style-type: none"> <li>• 176 x 144 (px)</li> <li>• 320x200 (px)</li> <li>• 352 x 288 (px)</li> <li>• 480 x 320 (px)</li> <li>• 720 x 240 (px)</li> <li>• 720 x 480 (px)</li> <li>• 768 x 576 (px)</li> <li>• 854 x 480 (px)</li> <li>• 1920 x 1080 (px)</li> <li>• 2560 x 2048 (px)</li> <li>• 3840 x 2400 (px)</li> <li>• Outros</li> </ul>
	Você possui algum comentário a mais para fazer sobre suas preferências de jogo?	Informações adicionais do jogo.
<b>Produto de saída:</b> P1.2.2 - Caracterização dos aprendizes		

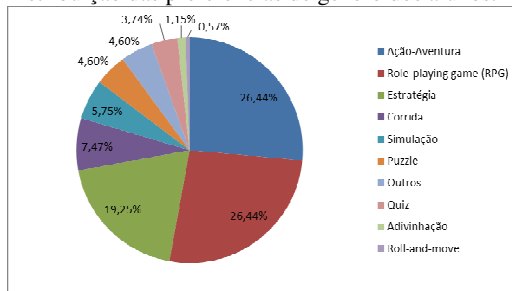
Faixa etária

Distribuição dos alunos por faixa etária.



Preferência de gêneros de jogo

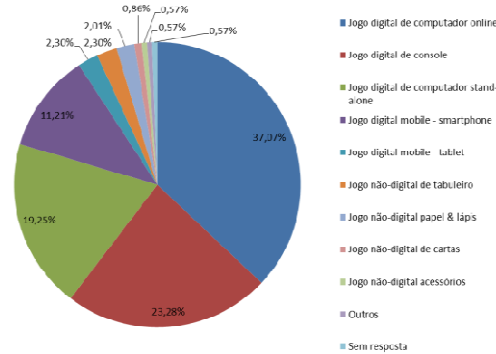
Distribuição das preferências de gênero dos alunos.





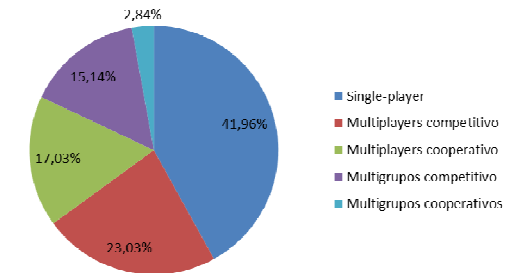
Preferência de plataformas de jogo

Distribuição das preferências de plataformas dos alunos.



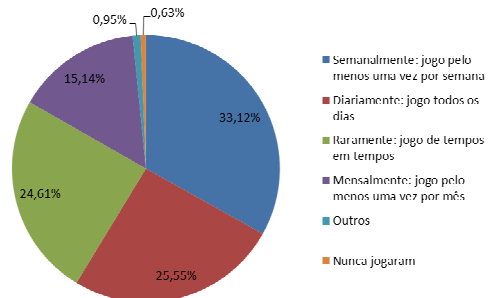
Modo de interação preferido

Distribuição do modo de interação do jogo dos alunos.



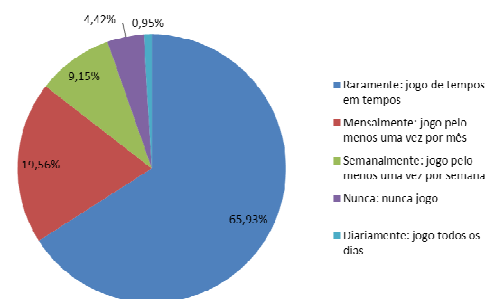
Frequência para jogar jogos digitais

Distribuição da frequência para jogar jogos digitais dos alunos.



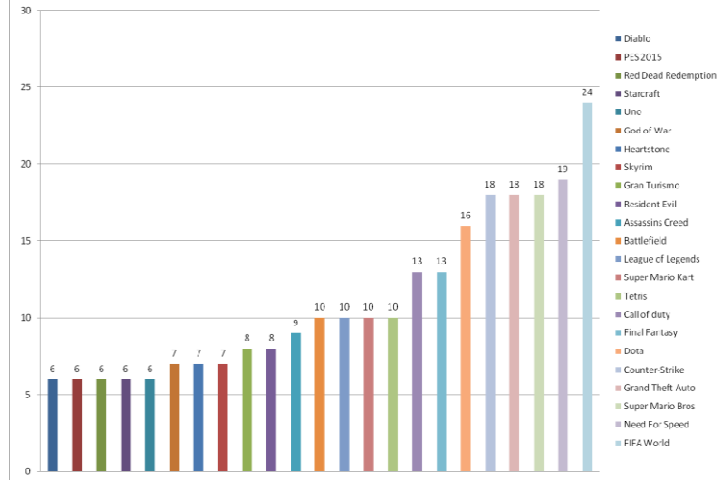
Frequência para jogar jogos não-digitais

Distribuição da frequência para jogar jogos não-digitais dos alunos.



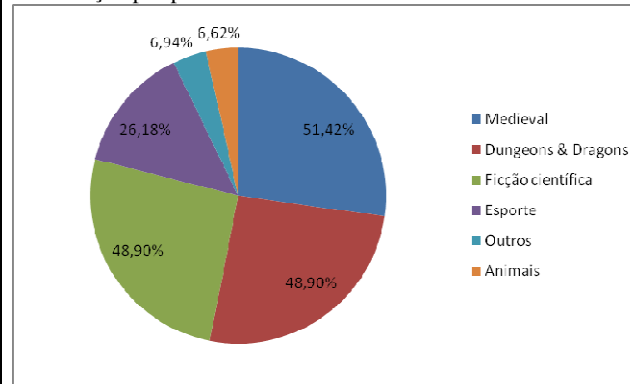
## Jogos favoritos

Os 10 jogos mais citados pelos aprendizes como favoritos.



## Cenários favoritos

## Distribuição por preferência de cenários.



Persona



José Bonifácio, 23 anos, masculino, é aluno do curso de Ciências da Computação da UFSC. Ele atualmente está na 6ª fase do curso e está matriculado na disciplina de Gerenciamento e Planejamento de Projetos. Ele passa horas vendo filmes do gênero medieval e de ação. Seu interesse por jogos de computador tem aumentado com o lançamento, no ano passado, de um novo jogo de RPG com cenário medieval. Ele fica mais de 6 horas semanais jogando online com seus amigos da universidade, onde formaram uma equipe que joga contra outras equipes de diversos países. Ele pretende até o final do semestre finalizar o jogo, para que no próximo ano possa jogar a versão que ainda será lançada.

O computador que José utiliza em casa é um notebook Avell, de 17 polegadas, HD de 1 TB e placa de vídeo GeForce de 3 GB. Ele comprou o computador após trabalhar durante dois anos na empresa de tecnologia do seu pai.

<b>Atividade</b>	<b>A1.3 Definir objetivo(s) de desempenho</b>
<b>Descrição</b>	<p>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 (MARZANO, 2007).</p> <p>Por meio dos objetivos de desempenho é possível avaliar se os alunos adquiriram as competências necessárias para aprendizagem (MECCLELLAND, 1973). A competência pode ser definida como a capacidade de ter conhecimento, habilidade e atitude em diversas situações (como a formação superior ou experiência prática) e passíveis de serem mobilizadas em situações específicas de trabalho (RUAS, 2004). O conhecimento é o saber que está relacionado a <i>saber fazer algo</i>. A habilidade é o conhecimento aplicado no dia-a-dia. A atitude é o querer fazer, ou ter motivação para aplicar o conhecimento e exercitar as habilidades (MASCARENHAS, 2008).</p>
<b>Tarefa</b>	<p><b>1.3 Definir objetivos de desempenho</b></p> <p>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 (DICK e CAREY, 1996; MAGER, 1997). O objetivo de desempenho, tipicamente é definido por <b>desempenho + condição + critério</b>.</p> <p>O <b>desempenho</b> é a descrição do que o aluno aprendeu, demonstrando domínio daquele objetivo proposto.</p> <p><b>Exemplos de desempenho:</b></p> <ol style="list-style-type: none"> <li>1. Ser capaz de compreender os papeis, artefatos e cerimônias do SCRUM.</li> <li>2. Ser capaz de aplicar os conceitos de orientação a objeto.</li> <li>3. Ser capaz de escrever um plano de projeto.</li> </ol>

A **condição** descreve as circunstâncias que o desempenho será realizado.

**Exemplos de condição:**

1. Dado os conceitos iniciais do desenvolvimento ágil de software....
2. Considerando as estruturas básicas de programa (laços, estruturas de decisão, variáveis, tipos primitivos)....

O **critério** define o grau de acerto do desempenho, sendo considerado suficientemente aceitável (MAGER, 1997).

**Exemplos de critérios:**

1. ... dentro de 2 horas/aula.
2. ... com não mais que duas ocorrências erradas.
3. ... dentro de um prazo de 10 horas.

Na criação de um desempenho é possível utilizar a taxonomia de Bloom (1956), a qual é dividida em três níveis: cognitivo(conhecimento) (BLOOM, 1956), nível psicomotor (habilidade) (SIMPSON, 1972) e afetivo (atitude) (KRATHWOHL et al., 1973).

**Taxonomia de Bloom**

<b>Nível</b>	<b>Item</b>	<b>Descrição</b>	<b>Verbos</b>
Nível cognitivo	<b>1. Conhecimento</b>	Memorização de materiais previamente aprendidos por fatos, termos, conceitos básicos e respostas.	conhecer; identificar; relatar; listar; definir; recordar; memorizar; repetir; nomear

		<p><b>2.Compreensão</b></p> <p>Comparação demonstrativa, organizando, comparando, traduzindo e interpretando fatos e ideias.</p>	<p>reapresentar; localizar; reportar; recordar; explicar; expressar; identificar; discutir; descrever; revisar; inferir; ilustrar; interpretar; desenhar; representar; diferenciar; concluir</p>
		<p><b>3. Aplicação</b></p> <p>Aplicar o novo conhecimento. Conseguir resolver problemas por meio da aplicação de um conhecimento, fatos e técnicas.</p>	<p>aplicar; relatar; desenvolver; traduzir; usar; operar; organizar; empregar; interpretar; demonstrar; ilustrar; praticar; calcular; mostrar; exibir; dramatizar</p>
		<p><b>4. Análise</b></p> <p>Examinar e dividir informações em partes, identificando suas causas. Fazer inferência e encontrar provas.</p>	<p>analisar; comparar; provar; investigar; examinar; contrastar; categorizar; diferenciar; detectar; pesquisar; classificar; deduzir; experimentar; descobrir; inspecionar; discriminar; separar</p>



	<b>5. Síntese</b>	Compilar informações criando resumos ou sínteses de um determinado conhecimento.	compor; produzir; projetar; montar; criar; preparar; modificar; dizer; planejar; inventar; formular; coletar; generalizar; documentar; combinar; relatar; propor; desenvolver; construir; organizar; derivar; escrever
	<b>6. Avaliação</b>	Apresentar opinião e fazer julgamentos sobre informações e ideias.	julgar; aferir; comparar; avaliar; concluir; mensurar; deduzir; argumentar; decidir; escolher; taxar; selecionar; estimar; validar; considerar; valorizar; criticar; inferir
	<b>1. Percepção</b>	Habilidade de usar estímulos sensoriais para orientar-se nas atividades.	escolher; descrever; detectar; diferenciar; distinguir; identificar; isolar; separar; selecionar
Nível psicomotor			

	<p><b>2. Prontidão</b></p>	<p>Prontidão para agir, por exemplo, conhecer e agir em uma sequência de etapas em um processo de comunicação.</p>	<p>começar;           exibir; explicar;    movimentar; prosseguir;    reagir; responder</p>
	<p><b>3. Resposta guiada</b></p>	<p>Estágios iniciais de uma aprendizagem de uma habilidade mais complexa, que inclui imitação, tentativa e erro. A adequação ao desempenho é alcançada por meio da prática.</p>	<p>montar;        construir; calibrar;     construir; desarticular;   fixar; corrigir;   manipular; medir;        esboçar; organizar</p>
	<p><b>4. Mecanismo</b></p>	<p>Estágio intermediário de uma habilidade complexa. Respostas aprendidas se tornam habituais, realizando movimentos com alguma confiança e proficiência.</p>	<p>montar;        construir; calibrar;     construir; desarticular;   fixar; corrigir;   manipular; medir;        esboçar; organizar</p>
	<p><b>5. Respostas complexas</b></p>	<p>Desempenho hábil, dos atos que envolvem padrões complexos e sem hesitação tendo desempenho automático.</p>	<p>montar;        construir; calibrar;     construir; desarticular;   fixar; corrigir;   manipular; medir;        esboçar; organizar</p>

		<b>6. Adaptação</b>	As habilidades são bem desenvolvidas e o indivíduo pode modificar os padrões de movimento para atender as necessidades especiais.	adaptar; alterar; mudar; reorganizar; rearranjar; revisar; variar
		<b>7. Originalidade</b>	Criação de novos padrões de movimento para atender a uma determinada situação ou problema específico. Os resultados da aprendizagem enfatizam a criatividade com base em habilidades altamente desenvolvidas, como a construção de uma nova teoria de comunicação.	organizar; combinar; compor; construir; criar; projetar; ser original
	Nível afetivo	<b>1. Receber</b>	Participar de uma situação com atenção passiva.	aceitar; participar; desenvolver; reconhecer
		<b>2. Responder</b>	Participar ativamente do processo de aprendizado e reagir expressando sua vontade (motivação).	completar; cumprir; cooperar; discutir; examinar; obedecer; responder;

	<b>3. Valorizar</b>	Atribuir valor para objetivos, fenômenos ou informações.	aceitar; defender; dedicar; propor; buscar
	<b>4. Organizar</b>	Apresentar diferentes valores, informações e ideias organizando em seu próprio esquema. Contrastando diferentes valores, resolvendo conflito e criando um sistema único de valores pessoais.	codificar; discriminar; exibir; ordenar; organizar; sistematizar
	<b>5. Caracterizar</b>	Apresentar um determinado valor que exerce influência em seus comportamentos, até se tornar uma característica pessoal.	internalizar; verificar
(WILLIAMS, 2004; ROTHWELL e KAZANAS, 1998; DICK e CAREY, 1996; WISCONSIN, 2012; MARZANO, 2007, BLOOM, 1956).			
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P1.1.1 - Plano de ensino da disciplina; P1.2.2 - Caracterização dos aprendizes.		
	<b>Produto de saída:</b> 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 (MAGER, 1997).	

## F2. PROJETO DA UNIDADE INSTRUCIONAL

<b>Atividade</b>	<b>A2.1 Definir avaliação do aluno</b>
<b>Descrição</b>	<p>Nesta atividade o objetivo é a definição da avaliação do aluno. A avaliação pode ser produzida de diversas formas, tipicamente são aplicadas em forma de exercícios, questões dissertativas, questões objetivas, interpretações de texto, chats, fóruns, trabalhos escritos, apresentações, e prova (DICK e CAREY, 1996; HOUAISS, 2009, BRANCH, 2009).</p> <p>No processo ENgAGED, considera-se principalmente que a avaliação é inserida no próprio jogo educacional. Por exemplo, ao inserir regras, no jogo, que avaliam o conhecimento do aluno em um determinado conceito.</p>
<b>Tarefa</b>	<p><b>2.1 Definir avaliação do aluno</b></p> <p>Nesta tarefa define-se a avaliação do aluno. A definição da avaliação inicia com a identificação dos objetivos de desempenhos que serão avaliados. Em seguida identifica-se a forma de avaliar os objetivos de desempenho por meio da avaliação.</p> <p>A avaliação pode ser inserida dentro do próprio jogo, por meio de regras ou objetivos do jogo. Deste modo, a medida que o aluno avança no jogo é avaliado. As regras do jogo são condições que dizem o que o jogador pode ou não fazer. Os objetivos são tarefas que o jogador precisa executar para poder vencer o jogo (FULLERTON, 2008; KREMERS, 2009).</p> <p>Existem outras formas de avaliação, como questões dissertativas, questões objetivas, interpretações de texto, chats, fóruns, trabalhos escritos, apresentações, e prova. Estas formas distintas de avaliação também podem ser inseridas no escopo da unidade instrucional para avaliar o jogo.</p> <p>Um exemplo de avaliação inserida no próprio jogo é a utilização da ferramenta Kahoot</p>

	<p>(<a href="https://getkahoot.com/">https://getkahoot.com/</a>). Com ela cria-se um <i>quiz</i> onde os alunos ficam conectados via <i>smartphone</i> ou navegador do computador para responder as respostas. Ao utilizar esta ferramenta todas as respostas dos alunos são salvas em uma planilha eletrônica que o Professor pode avaliar o resultado da aplicação do jogo.</p> <p>Um exemplo de outra forma de avaliação do jogo é a utilização de questões objetivas sobre o conteúdo após a aplicação de um jogo digital de RPG, como o jogo SCRUM-Scape (CAMARGO, 2013).</p>	
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P1.1.1 - Plano de ensino da disciplina; P1.3.1 - Objetivo(s) de desempenho.	
	<b>Produto de saída:</b> P2.1 - Avaliação do Aluno	
	Objetivo(s) de desempenho a serem avaliados	Objetivo(s) de desempenho extraídos do produto de trabalho <b><i>P1.3.1 - Objetivo(s) de desempenho</i></b> que serão avaliados.

	Avaliação	<p>Descrição da avaliação do aluno. Para uma avaliação inserida no jogo educacional, descreve-se como o jogo fará a avaliação. Por exemplo, para um dado conteúdo que será abordado no jogo, são criadas regras em um determinado nível (fases) do jogo para avaliar o conhecimento do aluno.</p> <p>Outras formas de avaliação também podem ser utilizadas, como:</p> <ul style="list-style-type: none"> <li>• Utilização de questões objetivas ou descritivas sobre um dado conceito abordado no jogo. Sendo entregue aos alunos em uma folha A4 após a aplicação do jogo.</li> <li>• Apresentação dos alunos sobre um dado conceito após a aplicação do jogo.</li> <li>• Realização de debate entre os alunos sobre os conceitos abordados no jogo.</li> <li>• Utilização de fórum para os alunos responderem questões específicas do conteúdo do jogo inseridas na <i>Learning Management System – LMS</i>, como o <i>Moodle</i>.</li> </ul>
--	-----------	--

<b>Atividade</b>	<b>A2.2 Definir conteúdo da estratégia instrucional</b>
<b>Descrição</b>	<p>Esta atividade tem por objetivo definir o conteúdo da estratégia instrucional e como ele será abordado ao longo do jogo. A estratégia instrucional define a organização e o seqüenciamento das atividades de aprendizagem. As atividades de aprendizagem são tentativas de organizar um conjunto de conteúdos, facilitando a interpretação e construção de conhecimentos e habilidades para o estudante (BRANCH, 2009).</p> <p>Existem diversos tipos de estratégias instrucionais como aula-expositiva, vídeo, palestra, laboratórios, leitura, exercício ou jogos (LABURÚ e CARVALHO, 2013; IEEE, 2014, DICK e CAREY, 1996).</p>

	<p>Para o processo ENgAGED a estratégia instrucional utilizada é o jogo educacional, sendo necessário nesta atividade definir os conceitos e sequenciamento do conteúdo ao longo do jogo.</p>
<p><b>Tarefa</b></p>	<p><b>2.2 Definir o conteúdo da estratégia instrucional</b></p> <p>Nesta tarefa o conteúdo da estratégia instrucional é definido. Ela inicia com a definição dos conteúdos abordados no jogo. Em seguida identifica-se o sequenciamento do conteúdo ao longo do jogo. Ao final, define-se a forma de interação do aluno (individual ou grupal).</p> <p>A seguir apresenta-se um exemplo de execução desta tarefa com base no jogo SCRUM-Scape (CAMARGO, 2013):</p> <p><b>Conteúdos abordados:</b></p> <ul style="list-style-type: none"> <li>• <i>Responsabilidades de Papéis:</i> SCRUM Master, DevelopmentTeam e ProductOwner.</li> <li>• <i>Artefatos:</i>Product Backlog, Sprint Backlog e Taskboard. Serão abordados principalmente uma visão geral dos objetivos do artefato e a sua conexão com os papéis.</li> <li>• <i>Cerimônias:</i>Daily SCRUM, Sprint Retrospective, Sprint Review e Sprint Planning. Serão abordados os conceitos e os objetivos ao longo do jogo.</li> </ul> <p><b>Sequenciamento do conteúdo:</b></p> <ul style="list-style-type: none"> <li>• <i>Etapa I:</i> Nesta etapa o conteúdo abordado serão os papéis.</li> <li>• <i>Etapa II:</i> Será inserido no jogo o conteúdo sobre as cerimônias.</li> <li>• <i>Etapa III:</i> Serão abordados os conceitos sobre os artefatos.</li> </ul> <p><b>Forma de interação dos alunos:</b> A forma de interação dos alunos será por aprendizado individual ou grupal.</p>
<p><b>Produtos de trabalho</b></p>	<p><b>Produto de entrada:</b> P1.1.2 - Especificação da Unidade Instrucional; P1.2.2 - Caracterização dos aprendizes; P1.2.3 - Caracterização do contexto; P1.3.1 - Objetivo(s) de desempenho.</p>



	<b>Produto de saída:</b> P2.2.1 – Conteúdos da Estratégias Instrucionais	
	Conteúdo	Conteúdo(s) abordado(s) ao longo do jogo educacional.
	Sequenciamento do conteúdo	Sequenciar os conteúdos abordados no jogo. Considerando que conteúdos mais básicos são apresentados primeiro e a medida que os conceitos tornam-se complexos são apresentados mais ao final do jogo.
	Forma de interação	Forma de interação dos alunos no jogo, sendo dividido em interação individual ou grupal.

<b>Atividade</b>	<b>A2.3 Definir pelo desenvolvimento do jogo ou utilizar jogo desenvolvido</b>
<b>Descrição</b>	Esta atividade tem por objetivo definir se o jogo educacional será desenvolvido ou se será utilizado um jogo já desenvolvido.
<b>Tarefa</b>	<p><b>2.4 Definir pelo desenvolvimento do jogo ou utilizar jogo desenvolvido</b></p> <p>Caso nesta tarefa seja definido que um novo jogo será desenvolvido, deve-se passar para a fase <i>F3 – Desenvolvimento do jogo educacional</i>.</p> <p>Caso nesta tarefa seja definido que será utilizado um jogo já desenvolvido pode-se buscar o jogo em sites como apresentado abaixo e em seguida deve-se passar para a fase <i>F4 - Execução da unidade instrucional</i>.</p> <p>Atualmente existem diversos jogos para ensinar computação, que estão disponíveis na literatura. Para maiores informações sobre jogos para ensinar computação sugere-se o relatório técnico <i>Systematic Literature Review (SLR) of Game-Based Learning for Computing</i> (BATTISTELLA e WANGENGHEIM, 2014), ou consultar, por exemplo, os seguintes repositórios educacionais para computação:</p>

	<p><b>1. Computer Science Unplugged:</b> apresenta diversas atividades de ensino de jogos não digitais para Ciência da Computação. Endereço: <a href="http://csunplugged.org/">http://csunplugged.org/</a></p> <p><b>2. Engineering PathWay:</b> apresenta diversos recursos educacionais para ensino em engenharia, ciência, matemática aplicada, ciência da computação/sistema de informação e engenharia da computação. Endereço: <a href="http://www.engineeringpathway.com">http://www.engineeringpathway.com</a></p> <p><b>3. NSDL – National Science Digital Library:</b> apresenta diversos recursos para aprendizagem nas áreas ciências, tecnologia, engenharia e matemática. Endereço: <a href="https://nsdl.org/">https://nsdl.org/</a></p> <p><b>4. Instructional Games Repository:</b> apresenta um repositório de jogos para ensinar computação. Endereço: <a href="http://srv.gqs.ufsc.br:8080/InstructionalGamesRepository">http://srv.gqs.ufsc.br:8080/InstructionalGamesRepository</a></p>	
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P1.1.2 - Especificação da Unidade Instrucional; P1.2.2 - Caracterização dos aprendizes; P1.2.3 - Caracterização do contexto; P1.3.1 - Objetivo(s) de desempenho.	
	<b>Produto de saída:</b> P2.3.1 – Definição pelo desenvolvimento do jogo ou utilização de jogo desenvolvido	
	Definição	Definir se será desenvolvido um novo jogo educacional ou se será utilizado um jogo já desenvolvido. Para o desenvolvimento do jogo deve-se seguir para a fase F3. Desenvolvimento do jogo educacional. Para a utilização do jogo já desenvolvido, deve-se seguir para a fase F4. Execução da Unidade Instrucional.

<b>Atividade</b>	<b>A2.4 Revisar o modelo de avaliação do jogo</b>
<b>Descrição</b>	<p>Esta atividade tem por objetivo revisar o modelo de avaliação de jogos educacionais para utilizá-lo após a aplicação do jogo na fase <b>F4 Execução da Unidade Instrucional</b>. O modelo de avaliação pré-definido no processo ENGAGED é o modelo MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011).</p> <p>Este modelo avalia a qualidade dos jogos educacionais com base nas variáveis abaixo (SAVI, 2011):</p> <ul style="list-style-type: none"> <li>• Motivação <ul style="list-style-type: none"> <li>○ Satisfação</li> <li>○ Confiança</li> <li>○ Relevância</li> <li>○ Atenção</li> </ul> </li> <li>• Experiência do Usuário <ul style="list-style-type: none"> <li>○ Competência</li> <li>○ Divertimento</li> <li>○ Desafio</li> <li>○ Interação Social</li> <li>○ Imersão</li> </ul> </li> <li>• Aprendizagem <ul style="list-style-type: none"> <li>○ Contribuição para vida profissional</li> <li>○ Eficiência na aprendizagem</li> <li>○ Contribuição no aprendizado na disciplina</li> </ul> </li> </ul>
<b>Tarefa</b>	<b>2.4 Revisar o modelo de avaliação do jogo</b>

Nesta tarefa deve-se revisar o modelo para avaliação do jogo educacional. No processo ENgAGED utiliza-se como padrão o modelo de avaliação MEEGA (SAVI, WANGENHEIM e BORGATTO, 2011). A tarefa inicia com a definição do objetivo da avaliação. Para criar o objetivo utiliza-se o seguinte modelo (SAVI, 2011):

*Analisar <objeto de estudo>  
Com o propósito de <propósito>  
Com respeito a <foco de qualidade>  
Sob o ponto de vista de <perspectiva>  
No contexto de <contexto>*

A seguir, apresenta-se um exemplo de objetivo (SAVI, 2011):

Analisar **um jogo educacional** com o propósito de avaliar a qualidade com respeito a **motivação, experiência do usuário e aprendizagem**, sob o ponto de vista **da percepção dos alunos** no contexto de **unidades instrucionais de Engenharia de Software**.

Deve-se definir o contexto onde o jogo será aplicado. O contexto da unidade instrucional foi definido na atividade **PI.1.2 - Especificação da Unidade Instrucional**. Nesta atividade o contexto que deve-se definir é o local em que a avaliação é realizada, por exemplo, logo após a aplicação do jogo na sala de aula ou depois da aplicação do jogo, estando os alunos em suas casas.

Deve-se definir também o design de pesquisa. O design de pesquisa do MEEGA é não experimental com pós-teste ( X O ) usando um único grupo. Entretanto, existem outros design de pesquisa que podem ser utilizados, como apresenta-se na tabela abaixo:

Tipo de estudo	Design	Representação
		(X=tratamento; O=mensurar/ evidência; R=atribuição aleatória)

Não-experimental	<b>One-shot post-test only</b>	X O
	<b>One-shot pre-test – post-test</b>	O X O
Quase-experimental	<b>Static group comparison group</b>	X O O
	<b>Static group pre-test – post-test</b>	O X O OO
	<b>Times Series</b>	O X O
Experimental	<b>Randomized post-test only</b>	R XO R O
	<b>Randomized pre-test – post-test</b>	R O XO R O
	<b>Randomized pretest - posttest control group</b>	R O X1O R O X2O

Deve-se também selecionar um instrumento para coleta de dados. No caso do modelo MEEGA apresenta-se um questionário para coleta de dados pré-definido. Porém, permite que sejam adicionadas perguntas sobre o conteúdo abordado no jogo. A tabela abaixo apresenta o questionário para coleta de dados do modelo MEEGA e em vermelho indica o local onde as questões referentes ao conteúdo do jogo devem ser adicionadas.

#### Questionário para Avaliação do Jogo Educacional.

##### Nome do jogo:

Gostaríamos que você respondesse as questões abaixo para nos ajudar a melhorar este jogo. Todos os dados são coletados anonimamente e somente serão utilizados no contexto desta pesquisa. Algumas fotografias poderão ser feitas como registro desta atividade, mas não serão publicadas em nenhum local sem autorização.

**Disciplina:**

Por favor, circule **um número** de acordo com o quanto você concorda ou discorda de cada afirmação abaixo.

Afirmações	Sua avaliação			Comentários sobre a questão
O design do jogo é atraente (interface ou objetos, como cartas ou tabuleiros).	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Houve algo interessante no início do jogo que capturou minha atenção.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
A variação (de forma, conteúdo ou de atividades) ajudou a me manter atento ao jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O conteúdo do jogo é relevante para os meus interesses.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O funcionamento deste jogo está adequado ao meu jeito de aprender.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O conteúdo do jogo está conectado com outros conhecimentos que eu já possuía.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Foi fácil entender o jogo e	Discordo		Concordo	

começar a utilizá-lo como material de estudo.	Fortemente	-2-1 0 +1 +2	Fortemente	
Ao passar pelas etapas do jogo senti confiança de que estava aprendendo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Estou satisfeito porque sei que terei oportunidades de utilizar na prática coisas que aprendi com o jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
É por causa do meu esforço pessoal que consigo avançar no jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Temporariamente esqueci das minhas preocupações do dia-a-dia, fiquei totalmente concentrado no jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Eu não percebi o tempo passar enquanto jogava, quando vi o jogo acabou.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Me senti mais no ambiente do jogo do que no mundo real, esquecendo do que estava ao meu redor.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Pude interagir com outras pessoas durante o jogo	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	

Me diverti junto com outras pessoas	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O jogo promove momentos de cooperação e/ou competição entre as pessoas que participam.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Este jogo é adequadamente desafiador para mim, as tarefas não são muito fáceis nem muito difíceis.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O jogo evolui num ritmo adequado e não fica monótono – oferece novos obstáculos, situações ou variações de atividades.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Me diverti com o jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Quando interrompido, fiquei desapontado que o jogo tinha acabado (gostaria de jogar mais).	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Eu recomendaria este jogo para meus colegas.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Gostaria de utilizar este jogo novamente	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Consegui atingir os objetivos do	Discordo		Concordo	



jogo por meio das minhas habilidades.	Fortemente	-2-1 0 +1 +2	Fortemente	
Tive sentimentos positivos de eficiência no desenrolar do jogo	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	

– O jogo contribuiu para a minha aprendizagem na disciplina  
(*Discordo fortemente*)-2 -1 0 +1 +2(*Concordo fortemente*)

– O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina.  
(*Discordo fortemente*)-2 -1 0 +1 +2(*Concordo fortemente*)

– A experiência com o jogo vai contribuir para meu desempenho na vida profissional.  
(*Discordo fortemente*)-2 -1 0 +1 +2(*Concordo fortemente*)

– Atribua uma nota de 1,0 a 5,0 para seu nível de conhecimento antes e depois do jogo aos conceitos listados na tabela abaixo (1,0 – pouco;5,0 – muito).

Conceitos	Lembrar o que é		Compreender como funciona		Aplicar na prática	
	Antes	Depois	Antes	Depois	Antes	Depois
Adicionar perguntas sobre conceitos abordados no jogo						

– Cite 3 pontos fortes do

	jogo: _____ _____ _____ _____  – Por favor, dê 3 sugestões para a melhoria do jogo: _____ _____ _____ _____  Muito obrigado pela sua contribuição!		
	<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P1.1.2 - Especificação da Unidade Instrucional; P1.2.2 - Caracterização dos aprendizes; P1.2.3 - Caracterização do contexto; P1.3.1 - Objetivo(s) de desempenho.	
		<b>Produto de saída:</b> P2.4.1 – Modelo de avaliação revisado	
		Objetivo	Objetivo(s) definidos para a avaliação.
		Contexto	Contexto para aplicação da avaliação do jogo.
	Design de pesquisa	Design de pesquisa utilizado para avaliar o jogo.	

	Instrumento para coleta de dados	Instrumento para coleta de dados, para o MEEGA utiliza-se o <i>Questionário de Avaliação de Jogos Educacionais</i> , sendo necessário adicionar perguntas específicas sobre o conhecimento do conteúdo do jogo.
--	----------------------------------	---

### F3. DESENVOLVIMENTO DO JOGO EDUCACIONAL

#### F3.1 ANÁLISE DO JOGO

<b>Atividade</b>	<b>A3.1.1 Levantar requisitos do jogo</b>
<b>Descrição</b>	<p>Esta atividade tem por objetivo identificar as funções e funcionalidades indispensáveis, necessárias e requeridas do jogo.</p> <p>(SWEBOK, 2004; SOMMERVILLE, 2007; ROMAN, 1985; COCKBURN, 2001; BOOCH, RUMBAUGH e JACOBSON, 2005).</p>
<b>Tarefa</b>	<p><b>3.1.1 Levantar requisitos do jogo</b></p> <p>Nesta tarefa levanta-se os requisitos para o desenvolvimento do jogo. Tipicamente este levantamento é realizado por meio de <i>brainstorming</i>, questionário, entrevista, observação ou casos de uso (FULLERTON, 2008; SCHELL, 2008; SOMMERVILLE, 2007).</p> <p>Existem dois tipos de requisitos, os funcionais e os não-funcionais. Os requisitos funcionais descrevem as funções que o jogo irá executar. Os requisitos não-funcionais são restrições do jogo (SWEBOK, 2004).</p> <p><b>Exemplos de requisitos funcionais do jogo:</b></p>

- O primeiro nível do jogo deve abordar os conceitos: *SCRUM Master*, *Development Team* e *Product Owner*.
- O jogo deve informar ao Professor da disciplina quais conceitos o jogador acertou e errou.
- O jogo deve simular o desenvolvimento de um software, permitindo o jogador simular as fases de iniciação, planejamento, execução, monitoramento e controle, encerramento de um projeto.
- O jogo deve permitir a execução de um *Sprint* do SCRUM.
- O jogo deve ensinar a ordenação de números inteiros utilizando o algoritmo *quicksort*.

**Exemplos de requisitos não-funcionais do jogo:**

- As cartas de um jogo não-digital devem ser impressas com o tamanho A3 e papel *couche* colorido.
- Para um jogo de *smartphone* o jogo só deve iniciar se o *vibracall* estiver ativado.
- Para um jogo digital conectado a banco de dados, o jogo deve salvar as informações localmente caso ocorra problema de conexão de internet. Após restabelecer a conexão, os dados salvos localmente são salvos no banco de dados do servidor.
- O jogo deve estar sob a licença GPL (*General Public License*).
- As fontes do jogo digital devem ser empacotadas em uma pasta dentro do diretório do jogo. Quando o jogo é instalado, o gerenciador de instalação deve salvar as fontes na pasta de fontes do sistema operacional.
- Em um jogo digital, o jogador não deve esperar mais do que 2 segunda para visualizar o curso mudando de lugar dentro do cenário.
- O jogo digital deve rodar em múltiplos sistemas operacionais.
- A duração da aplicação do jogo não deve passar de 2 horas/aula.

Para descrever as restrições do jogo, serão utilizados três elementos: tipo de requisito, descrição e Prioridade:

- **Tipo de requisito:** utiliza-se RF para requisitos funcionais e RNF para requisitos não funcionais.

	<ul style="list-style-type: none"> <li>• <b>Descrição:</b> cria-se uma descrição para cada restrição do jogo, com base nas técnicas citadas anteriormente (<i>brainstorming</i>, questionário, entrevista, observação, cenário, casos de uso e etnografia).</li> <li>• <b>Prioridade:</b> identifica-se a prioridade de cada restrição com base em três opções: essencial, importante ou desejável (FILHO, 2010; IEEE STD 830, 1998; SOMMERVILLE, 2007): <ul style="list-style-type: none"> <li>○ A prioridade essencial é o requisito sem o qual o jogo não funciona corretamente.</li> <li>○ A prioridade importante é o requisito sem o qual o jogo não para de funcionar, mas deve ser implementado o mais breve possível.</li> <li>○ A prioridade desejável é o requisito que não compromete as funcionalidades básicas do jogo. Este tipo de requisito pode ser desenvolvido por último, sem comprometer o sistema.</li> </ul> </li> </ul> <p>(FULLERTON, 2008; SCHELL, 2008).</p>				
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P1.1.2 - Especificação da Unidade Instrucional; P1.2.2 - Caracterização dos aprendizes; P1.2.3 - Caracterização do contexto; P1.3.1 - Objetivo(s) de desempenho				
	<b>Produto de saída:</b> P3.1.1 - Requisitos do jogo				
	Tipo de Requisito  Requisitos funcionais: <b>RF</b>  Requisitos não-funcionais: <b>RNF</b>	Descrição  Descrição contendo o requisito do sistema	Prioridade  <input type="checkbox"/> Essencial <input type="checkbox"/> Importante <input type="checkbox"/> Desejável		

## F3.2 CONCEPÇÃO DO JOGO

<b>Atividade</b>	<b>A3.2.1 Conceber o jogo</b>
------------------	-------------------------------

<b>Descrição</b>	<p>Esta atividade tem por objetivo conceber o jogo, descrevendo as principais características do jogo como objetivos do jogo, história do jogo, regras do jogo, elementos do jogo, <i>feedbacks</i> e nome do jogo.</p> <p>(FULLERTON, 2008; CHANDLER, 2012; KREMERS, 2009; SALEN e ZIMMERMAN, 2009).</p>
<b>Diversão na concepção de jogos</b>	<p>No processo de desenvolvimento de jogos educacionais é necessário que o criador de jogos desenvolva jogos realmente divertidos. A seguir, são apresentadas características para tornar os jogos educacionais mais divertidos do ponto de vista do jogador (FULLERTON, 2008; KREMERS, 2009; SALEN e ZIMMERMAN, 2009, OMERINCK, 2004).</p> <p>Para que os jogos sejam realmente divertidos o criador de jogos precisa equilibrar três características básicas: desafio, fantasia e curiosidade. Para que o equilíbrio entre estes fatores ocorra, é necessário que o jogo apresente as seguintes características (MALONE, 1980):</p> <ul style="list-style-type: none"> <li>• Ter objetivos claros.</li> <li>• Ter várias formas de pontuação, para apresentar <i>feedback</i> do jogador a medida que avança no jogo.</li> <li>• Ter vários níveis de dificuldade, para estimular os alunos a superar os desafios do jogo.</li> <li>• Ter elementos aleatórios que surpreendam o jogador, tornando o jogo dinâmico</li> <li>• Ter uma história de fantasia envolvente e emocionante, para permitir maior imersão dos jogadores.</li> </ul> <p>Um dos fatores estreitamente relacionados com a diversão em jogos é o aprendizado do jogador ao longo do jogo. A seguir são apresentadas algumas características que auxiliam o criador de jogos a tornar um jogo mais divertido (GUEDES, OLIVEIRA e PEREIRA, 2009):</p> <ul style="list-style-type: none"> <li>• Ter um “mundo” consistente, permitindo que o jogador compreenda o resultado das suas ações ao longo do jogo.</li> <li>• Ter entendimento dos limites do “mundo”, permitindo que o jogador saiba o que pode ou não pode fazer durante o jogo.</li> </ul>

- O jogador precisa saber qual o “caminho” precisa seguir para vencer o jogo.
- O jogo precisa ser planejado de modo que o jogador veja que ao cumprir as tarefas, veja seu progresso no jogo.
- O jogo precisa criar um ambiente de imersão para o jogador “entrar no mundo do jogo”. Deste modo, o jogador ficará concentrado no jogo.
- Jogos fáceis ou difíceis de mais frustram o jogador. Deste modo, o jogo precisa criar um equilíbrio entre as falhas que o jogador comente, permitindo que mesmo cometendo alguns erros (dificuldade do jogo), ele consiga avançar no jogo.
- O jogo precisa ser estruturado de modo que os desafios do jogo sejam possíveis de serem vencidos.
- Para manter o jogador motivado, ele precisa ver que está aprendendo o jogo, adquirindo habilidades novas, deste modo ele continuará descobrindo novas fases, níveis ou desafios do jogo.

A seguir apresenta-se mais orientações para tornar um jogo mais divertido (FULLERTON, 2008; KREMERS, 2009; SALEN e ZIMMERMAN, 2009, OMERINCK, 2004):

- Criar uma abertura de jogo adequada, que não demore um longo período de tempo e chame a atenção do jogador. A abertura é a primeira impressão do jogo e desde o início é necessário envolver o jogador no enredo do jogo.
- Ter nas primeiras missões, desafios mais fáceis. A medida que o jogador avança nas missões, o jogo torna-se mais complexos.
- Ter a opção de vários *avatares* (personagem do jogador), assim cada jogador pode escolher um personagem que mais se adéque ao seu estilo de jogo.
- Ter jogabilidade, ou seja, permitir que o jogador realize ações durante o jogo com liberdade para explorar o ambiente, sendo fácil e intuitivo para jogar.
- Caso seja um jogo digital é importante ter gráficos realistas, com texturas adequadas e artes bem criadas. Jogos com texturas ruins dão a impressão que o jogo não foi bem desenvolvido.
- Estruturar o jogo integrando os sons do jogo, como músicas e sons dos personagens. Sons repetitivos e monótonos podem levar o jogador a abandonar o jogo.

	<ul style="list-style-type: none"> <li>• A história do jogo precisa ser interessante, desafiando o jogador a descobrir as “novidades” do jogo. P. ex. em um jogo que a medida que o jogador avança, descobre novas fases, tendo missões cada vez mais complexas.</li> <li>• Ter comandos fáceis de utilizar. Para computadores e consoles o professor pode utilizar comandos que normalmente são empregados em outros jogos de mesmo gênero.</li> <li>• Ter a opção para o jogador escolher muitas perspectivas de visão do jogo. Normalmente apresentando câmeras de diversos ângulos do cenário.</li> <li>• Ter um painel de controle, com as informações sobre visíveis, como vida do jogador, duração da missão, velocidade do automóvel (caso seja um jogo de corrida), menu de utilidades, mapas.</li> <li>• Tipicamente para o desenvolvimento de elementos do jogo é utilizado fotos reais de locais, objetivos e personagens para que na implementação destes elementos fiquem mais próximo possível do mundo real.</li> </ul>
<p><b>Tarefa</b></p>	<p><b>3.2.1 Conceber jogo</b></p> <p><b>Objetivo do jogo</b>  Esta tarefa inicia com a definição do objetivo do jogo. O objetivo é o propósito do jogo, tipicamente representado por meio de missões. Uma missão é o dever do jogador a ser completado ou conquistado ao longo do jogo. Um jogo pode ter uma ou várias objetivos. Quando o jogador vence o(s) objetivo(s) do jogo, pode-se dizer que ele venceu o jogo.</p> <p>O objetivo do jogo pode ser definido da seguinte maneira: <b>artigo definido + sujeito + verbo (ação) + objetivo</b> (FULLERTON, 2008, CHANDLER, 2012; KREMERS, 2009; SALEN e ZIMMERMAN, 2009).</p> <p>A concepção do objetivo é descrito com artigo definido (o, a, os, as), seguido do sujeito (avatar, personagem principal, jogador), seguido do verbo que demonstra ação (por exemplo: vencer, alinhar, conquistar, ultrapassar, resgatar, gerenciar) e por fim o objetivo.</p>



Na criação do objetivo do jogo é necessário formá-lo com base nos objetivos de desempenho (**PI.3.1 - Objetivo(s) de desempenho**) definidos anteriormente. Os exemplos a seguir mostram como criar um objetivo de jogo alinhado ao objetivo de desempenho do jogo.

#### **Exemplos de objetivos:**

- O *avatar* deve libertar os personagens do jogo respondendo perguntas sobre o SCRUM.
- O jogador deve responder corretamente as perguntas do *quiz* sobre Gerência de Projetos.
- O jogador deve ordenar o conjunto de números em ordem lógica seguindo os passos do algoritmo *Heapsort*.
- O avatar deve corrigir algoritmos ao longo do jogo para salvar a princesa no castelo do dragão.

#### **Gênero do jogo**

O gênero de um jogo é a categorização que classifica e agrupa os tipos de jogos de acordo com suas características. A categorização utilizada pelo processo ENgAGED, como gênero de jogo são (HERZ, 1997; ADAMS e ROLLINGS, 2006; SCHIFFLER, 2006; WOLF, 2001):

- **Ação-aventura:** Um jogo que exigem que os jogadores utilizem reflexos rápidos, com precisão para superar obstáculos, resolver desafios ou para responder perguntas.
- **Adivinhação:** Um jogo em que o objetivo é identificar algum tipo de informação, como uma palavra, a partir de desenhos ou imitação de outros jogadores.
- **Corrida:** Um jogo em que o jogador comanda um veículo ou participa de uma corrida, tentando se mover mais rápido do que os oponentes para alcançar um objetivo específico ou ser mais rápido que um determinado tempo.
- **Estratégia:** Um jogo que é focado na jogabilidade e exige pensamentos táticos e cuidados, a fim de conseguir a vitória.
- **Puzzle:** Um jogo que envolve personagens que controlam um conjunto de comandos, para navegar através de labirintos ou para organizar os objetos de maneira correta.

- **Quiz:** Um jogo onde o jogador precisa responder perguntas para uma determinada área de conhecimento.
- **Role-playing game (RPG):** Um jogo onde o jogador controla ações de um protagonista e com este personagem vive imerso em um mundo fictício. Em jogo de *role-playing* os personagens interagem com este mundo e ficam mais fortes.
- **Roll-and-move:** Jogos de tabuleiro em que as fichas são movidas com base nos resultados mostrados em um ou mais dados (por exemplo: *snake and ladders* e *Monopoly*).
- **Simulação:** Um jogo desenvolvido para colocar o jogador no controle de um determinado ambiente ou atividade, o qual busca ser o mais realista possível.

Para auxiliar na definição do gênero do jogo, apresenta-se abaixo uma relação entre os níveis de aprendizagem da Taxonomia de Bloom e gênero de jogos, criada com base em Battistella e Wangenheim (2014).

Nível cognitivo	<b>1. Conhecimento</b>	Quiz, puzzle, roll-and-move, estratégia
	<b>2. Compreensão</b>	Ação-aventura, puzzle, estratégia, RPG, Adivinhação
	<b>3. Aplicação</b>	Simulação, ação-aventura, corrida, RPG
	<b>4. Análise</b>	Simulação
	<b>5. Síntese</b>	Simulação
	<b>6. Avaliação</b>	Simulação
Nível psicomotor	<b>1. Percepção</b>	Ação-aventura, corrida
	<b>2. Prontidão</b>	Ação-aventura, corrida
	<b>3. Resposta guiada</b>	RPG, Simulação
	<b>4. Mecanismo</b>	Simulação, puzzle, ação-aventura

	<b>5. Respostas complexas</b>	Estratégia, Simulação
	<b>6. Adaptação</b>	Simulação
	<b>7. Originalidade</b>	Simulação
Nível afetivo	<b>1. Receber</b>	Estratégia, RPG
	<b>2. Responder</b>	Simulação, Quiz, RPG
	<b>3. Valorizar</b>	RPG, simulação, ação-aventura
	<b>4. Organizar</b>	Simulação
	<b>5. Caracterizar</b>	Simulação

#### **Plataforma do jogo**

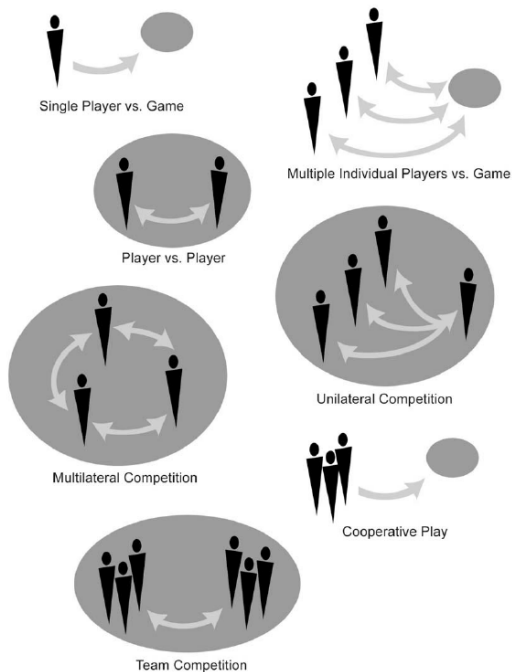
A plataforma classifica se o jogo é digital ou não-digital. Abaixo apresenta-se as plataformas sugeridas para desenvolvimento de jogos educacionais (CONNOLLY et al., 2012; CAULFIELD, VEAL e MAJ, 2011).

- Preferência de plataforma de jogo:
- **Jogo digital de computador stand-alone** p. ex. computador pessoal sem acesso a internet.
- **Jogo digital de computador online** p. ex. computador pessoal com acesso a internet.
- **Jogo digital de console** p. ex. Xbox 360, Playstation, Wii - Nintendo.
- **Jogo digital mobile - smartphone** p. ex. iPhone, Samsung Galaxy.
- **Jogo digital mobile - tablet** p. ex. iPad, Samsung Galaxy Note.
- **Jogo não-digital de tabuleiro.**
- **Jogo não-digital de cartas.**
- **Jogo não-digital papel & lápis** (para ser jogado apenas com papel e lápis.) p. ex. jogo imagem e ação, "caça-palavras".
- **Jogo não-digital acessórios** (para ser jogado com acessórios ou adereços (objetos portáteis)) p.

ex. Lego.

**Modo de interação**

O modo de interação é a maneira como o jogador pode interagir com os outros jogador(es) e/ou computador(es)/console/mobile. Os modos de interação do jogo definidos pelo processo são *single-player*; *multiplayer* e *multigrupos* (FULLERTON, 2008; KREMERS, 2009).



### Regras do jogo

As regras são restrições contidas no jogo que visam orientar o jogador ao longo do jogo, determinando condutas corretas ou erradas. As regras são as normas, orientações, fluxo de ações, ou situações que

limitam a atuação do jogador dentro do jogo. Elas definem a qualidade do jogo e apresentam um número fixo de diretrizes abstratas que descrevem como o funciona o jogo (SALEN e ZIMMERMAN, 2009).

Regras complexas podem influenciar o jogador a deixar de jogar o jogo. Regras excessivamente simples também podem levar o jogador a desistir do jogo, por falta de desafio ou motivação. As regras precisam estar alinhadas ao objetivo do jogo. A seguir, apresenta-se exemplos de regras de jogo.

1. O personagem principal não pode exceder o tempo máximo 30 minutos em cada missão;
2. Caso o jogador não encontre alguns artefatos do jogo, serão descontados pontos;
3. Para avançar de nível ou fase, o jogador precisa desvendar todos os enigmas daquele nível;
4. O jogador terá em cada nível a oportunidade de solicitar ajuda de um “mago”, o qual dará dicas do que ainda precisa ser feito naquele nível.
5. Em um *quiz game* cada jogador terá 30 segundos para responder cada questão.

#### **Mecanismo do jogo**

A mecânica do jogo são as ações do personagem realizada durante o jogo, elas são regras específicas para limitar o que o personagem pode realizar durante o jogo. Um exemplo de mecânica do jogo digital: utilizar o teclado do computador para correr, saltar, caminhar e abaixar o personagem no cenário do jogo. Exemplo de mecânica de jogo não-digital: jogar dois dados para movimentar a ficha do jogador (BRATHWAITE e SCHREIBER, 2009).

#### **Narrativa**

A narrativa é uma história com narrações ordenadas, descrevendo acontecimentos ocorrido(s) no(s) cenário(s) onde se passa o jogo. Na narrativa, descreve-se o local aonde vivem os personagens, a época, as relações entre os personagens do jogo. A história é o enredo do jogo que precisa motivar o jogador a ficar cada vez mais imerso, jogando o jogo (KREMERS, 2009).

**Elementos do jogo**

Um jogo pode ser constituído diversos elementos que compõem, estruturam ou formam o jogo. A seguir apresenta-se os elementos básicos utilizados como sugestão pelo processo ENgAGED, entretanto, outros elementos também podem ser concebidos pelo criador de jogos (FULLERTON, 2008; KREMERS, 2009; SCHELL, 2008).

- **Personagem:** O personagem é um papel fictício utilizando por um ator ao longo do jogo. Um personagem deve ter um nome, tipo (jogador, adversário, máquina) e uma descrição das suas ações no jogo.
- **Artefatos:** Os artefatos são objetos que terão maior interação com o jogador, normalmente sendo atribuído algum ponto, por exemplo, o jogador precisa percorrer uma nave espacial buscando peças da nave, ou então, em um jogo de corrida o jogador deve pegar o maior número de moedas de ouro.
- **Cenários:** Os cenários são conjuntos de objetos ou acessórios que ocupam a cena do jogo. Eles são enredos ou ambientes nos quais os jogadores movimentarão seus personagens permitindo deste modo prosseguir e avançar no jogo. São exemplos de cenários: castelos, cidades, naves espaciais, pista de corrida.
- **Menu:** O menu do jogo contém informações gerais de configuração do jogo, inventário dos jogadores, opções para salvar, abrir ou sair do jogo.

**Critérios para vencer o jogo**

Um critério é uma norma a ser seguida pelo jogador para alcançar os objetivos do jogo e vencê-lo. Os critérios mensuram o desempenho do jogador, por exemplo, utilizando tempo de jogo, vidas, objetivos do jogo conquistados. Um jogo pode ser constituído de vários critérios, mas para tornar um jogo divertido os critérios não podem ser excessivamente complexos ou extremamente fáceis

(FULLERTON, 2008; KREMERS, 2009; SCHELL, 2008).

São exemplos de critérios do jogo:

- Cada nível do jogo tem um tempo de duração de no máximo 30 minutos, caso exceda este tempo o jogo é finalizado.
- Se o jogador não encontrar um determinado objeto, não conseguirá vencer o jogo.
- Se o personagem perder mais do que três “vidas” o jogo é finalizado.
- Se o jogador errar mais que 4 conceitos do conteúdo educacional do jogo, o personagem volta para o primeiro nível do jogo.

### **Pontuação do jogo**

A pontuação do jogo é um valor atribuído aos acertos ou erros do jogador. Portanto, a cada interação com personagens ou resolução de desafios do jogo, o jogador pode pontuar.

A pontuação deve ser identificada do seguinte modo: contendo uma descrição do que é a pontuação, o valor da pontuação e o nível do jogo em que a pontuação é válida. Abaixo apresenta-se exemplos de pontuação do jogo:

**Descrição:** O jogador deve conversar com os membros de sua equipe.

**Pontuação:** 10

**Nível:** Primeiro nível do jogo.

### ***Feedback educacional***

O *feedback* educacional é a informação apresentada ao jogador, relatando os erros ou acertos sobre os conceitos definidos nos *objetivos de desempenho* na fase F1 de Análise da Unidade Instrucional (BRANCH, 2009). O *feedback* é uma poderosa ferramenta para ensino e pode impactar positivamente ou negativamente no aprendizado do aluno (HATTIE e TIMPERLEY, 2007). O impacto positivo ocorre quando o aluno recebe o *feedback* e não desmotiva o aluno, permitindo aprender mesmo recebendo um feedback do que não conseguiu aprender. O impacto negativo ocorre quando o *feedback* tem efeito contrário ao esperado, em vez de ensinar o aluno, acaba desmotivando a aprender o



conteúdo (HATTIE e TIMPERLEY, 2007).

Em um jogo o *feedback* precisa ser positivo, motivando os jogadores e levando a vencer o jogo. O *feedback* educacional precisa ser inserido no jogo conforme as seguintes recomendações (BRANCH, 2009):

- Motivar o jogador quando acerta um conceito do jogo.
- Corrigir os erros conceituais que o jogador comente, sem que desista de jogar.
- Oferecer *feedback* no mesmo instante da ação que o jogador cometeu o erro, mas também no final do jogo.

A criação do *feedback* educacional segue os seguintes passos:

- Levantar os conceitos referentes ao conteúdo instrucional que será ensinado, conforme a fase F1. Análise da Unidade Instrucional.
- Para cada conceito, informa-se como será apresentado ao jogador o *feedback* educacional dos seus acertos.
- Para cada conceito, informa-se como será apresentado ao jogador o *feedback* educacional dos seus erros.
- Definir se o jogo terá *feedback* educacional no mesmo instante em que comente o erro, ou será ao final do jogo.

Abaixo apresenta os tipos de *feedback* educacionais que um jogo pode ter (FULLERTON, 2008; KREMERS, 2009; SCHELL, 2008, HATTIE e TIMPERLEY, 2007):

- **Feedback positivo:** apresenta os aspectos positivos (acertos) do jogador.
- **Feedback negativo:** apresenta os aspectos negativos (erros) do jogador.
- **Feedback de desempenho:** apresenta *feedback* se o movimento que o jogador realizou foi correto ou errado. Portanto estando relacionado ao desempenho do jogador em realizar alguma determinada ação no jogo.
- **Feedback simultâneo:** apresenta *feedback* positivo ou negativo ao jogador no instante em que ele

	<p>realizou uma determinada ação.</p> <ul style="list-style-type: none"> <li>• <b>Feedback final:</b> apresenta <i>feedback</i> positivo ou negativo ao jogador no final do jogo.</li> <li>• <b>Feedback quantitativo:</b> apresenta <i>feedback</i> em forma de pontuação ou classificação.</li> <li>• <b>Feedback qualitativo:</b> apresenta <i>feedback</i> sobre a performance do jogador em níveis mais fáceis ou difíceis.</li> </ul> <p><b>Nome do jogo</b> Esta tarefa finaliza com a definição do nome do jogo. Tipicamente o nome do jogo não deve ser extenso e deve ser facilmente lembrando pelo jogador.</p>	
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P1.2.2 - Caracterização dos aprendizes; P1.3.1 - Objetivo(s) de desempenho; P3.1.1 - Requisitos do jogo;	
	<b>Produto de saída:</b> P3.2.1 – Concepção do jogo	
	Objetivo(s) do jogo	Artigo definido + sujeito + verbo (ação) + objetivo.
	<b>Características do Jogo</b>	
Gênero do jogo	Ação-aventura, Adivinhação, Corrida, Estratégia, Puzzle, Quiz, Role-playing game (RPG), Roll-and-move e Simulação (HERZ, 1997; ADAMS e ROLLINGS, 2006; SCHIFFLER, 2006; WOLF, 2001).	

	Plataforma do jogo	<p>Preferência de plataforma de jogo:</p> <ol style="list-style-type: none"> <li>1. <b>Jogo digital de computador <i>stand-alone</i></b> p. ex. computador pessoal sem acesso a internet.</li> <li>2. <b>Jogo digital de computador online</b> p. ex. computador pessoal com acesso a internet.</li> <li>3. <b>Jogo digital de console</b> p. ex. XBox 360, Playstation, Wii - Nintendo.</li> <li>4. <b>Jogo digital mobile - smartphone</b> p. ex. iPhone, Samsung Galaxy.</li> <li>5. <b>Jogo digital mobile - tablet</b> p. ex. IPad, Samsung Galaxy Note.</li> <li>6. <b>Jogo não-digital de tabuleiro.</b></li> <li>7. <b>Jogo não-digital de cartas.</b></li> <li>8. <b>Jogo não-digital papel &amp; lápis</b> (para ser jogado apenas com papel e lápis.) p. ex. jogo imagem e ação, "caça-palavras".</li> <li>9. <b>Jogo não-digital acessórios</b> (para ser jogado com acessórios ou adereços (objetos portáteis)) p. ex. Lego.</li> </ol> <p>(CONNOLLY et al., 2012; CAULFIELD, VEAL e MAJ, 2011)</p>
	Modo de interação	<ul style="list-style-type: none"> <li>• <b>Single-player:</b> você joga sozinho, competindo contra si mesmo ou o computador.</li> <li>• <b>Multiplayers competitivo:</b> vários jogadores competem entre si.</li> <li>• <b>Multiplayers cooperativo:</b> vários jogadores cooperam para vencer o jogo em conjunto.</li> <li>• <b>Multigrupos competitivo:</b> grupos competem entre si.</li> <li>• <b>Multigrupos cooperativos:</b> grupos reunidos de forma cooperativa.</li> </ul>
	Regras	<p>Regra do jogo, contendo as normas, restrições ou penalizações para ganhar pontuações enquanto o jogador joga o jogo.</p>

Mecânica	A mecânica são as ações que limitam o comportamento ou ações do(s) personagem(ns) no jogo.
Narrativa	<p>Descrição geral da história do jogo, sendo um roteiro a ser seguido na fase de design do jogo. A narrativa permite que os cenários sejam encadeados e organizem de forma lógica e sequencial os elementos do jogo.</p> <p>Por exemplo, em um jogo de RPG deve-se descrever a contexto do jogo (época, cenário) e descrever como o(s) personagem(ns) avança(m) no jogo, passando por vários objetivos de jogo.</p>
<b>Elementos do Jogo</b>	
Nome	Nome do elemento do jogo
Tipo	<ul style="list-style-type: none"> <li>• <b>Personagem:</b> Existem dois tipos de personagens segundo Oliveira (2014): <ul style="list-style-type: none"> <li>○ <b>Avatar:</b> personagem controlado pelo jogador.</li> <li>○ <b>NPC (<i>non-player character</i>):</b> personagens que não são controlados pelo jogador.</li> </ul> </li> <li>• <b>Artefatos:</b> Os artefatos são elementos para manipulação ou interação do personagem com os cenários.</li> <li>• <b>Cenário:</b> O cenário do jogo corresponde a cena na qual os personagens vão interagir objetivando avançar nas cenas.</li> <li>• <b>Menu:</b> O menu permite ao jogador navegar entre as opções do jogo.</li> </ul>
Papel	No caso do personagem, deve-se inserir o papel que o personagem executa.
Função	Se necessário, deve-se identificar a função do elemento do jogo.
Nível	No caso do tipo de elemento “Cenário”, deve-se informar o nível de um cenário.

<b>Critérios para vencer o jogo</b>	
Critérios	Descrever os critérios necessários para o jogador vencer o jogo. Informando desta forma alguma forma de pontuação, por exemplos, vencer o jogo com um número “x” de vidas ou alcançar “x” pontos em todos os níveis do jogo.
<b>Pontuação do jogo</b>	
Descrição	Descrição do que o jogador precisa fazer para ganhar pontos
Pontuação	Valor indicando a pontuação do jogador, caso vença um desafio, solucione um problema ou não solucione um problema. Portanto, pode ser um valor positivo ou negativo.
Nível	Nível em que aquele tipo de pontuação é válido.
<b>Feedbacks educacionais ao jogador</b>	
Feedback	<p>Seguir os seguintes passos para criação do <i>feedback</i> educacional:</p> <ul style="list-style-type: none"> <li>• Levantar todos os conceitos referente ao conteúdo que será ensinado, conforme a fase F1. Análise da Unidade Instrucional.</li> <li>• Para cada conceito, informa-se como será apresentado ao jogador o <i>feedback</i> educacional dos seus acertos.</li> <li>• Para cada conceito, informa-se como será apresentado ao jogador o <i>feedback</i> educacional dos seus erros.</li> <li>• Definir se o jogo terá <i>feedback</i> educacional no mesmo instante em que comente o erro, ou será ao final do jogo.</li> </ul>
Nome do jogo	Nome do jogo.

### F3.3 DESIGN DO JOGO

<b>Atividade</b>	<b>A3.3.1 Definir linguagem de programação ou <i>game engine</i> do jogo digital</b>
<b>Descrição</b>	<p>Esta atividade tem por objetivo definir a linguagem de programação e ou <i>game engine</i>. A linguagem de programação é o meio pelo qual descreve-se precisamente conceitos, algoritmos e resolvem problemas (ACM/IEEE, 2013). <i>Game engine</i> é um <i>framework</i> criado para desenvolver jogos, permitindo, por exemplo, a criação de gráficos em 2D ou 3D, detecção de colisão entre elementos do jogo, utilização de som para cenários, objetos e personagens, criação de <i>scripts</i> que descrevem as ações dos elementos do jogo e inteligência artificial. (GREGORY, 2014).</p> <p>Esta atividade, aplica-se exclusivamente para jogos do gênero digital.</p>
<b>Tarefa</b>	<p><b>3.3.1 Definir linguagem de programação ou <i>game engine</i> do jogo digital</b></p> <p>Nesta tarefa é definida a linguagem de programação para desenvolver o jogo, podendo haver mais de uma linguagem. São exemplos de linguagem de programação: Java, JavaScript, HTML5, CSS, C, C++, C#, .Net, pascal, PHP.</p> <p>Nesta tarefa é definida a <i>game engine</i> do jogo. Tipicamente utiliza-se apenas uma <i>game engine</i> para desenvolver jogos. São exemplos de <i>game engine</i>: 3D GameStudio, BGE (Blender), C4 Engine, DX Studio, Esperient Creator, Game Maker, Irrlicht, NeoAxis 3D Engine, Panda3D, RPG Maker, RPG Toolkit, Torque Game Engine, Truevision3D, Unity, Unreal Engine, Crystal Space, jME, OGRE, OpenSceneGraph.</p> <p>Tipicamente a utilização de <i>game engine</i> no processo de desenvolvimento de jogos tem por objetivo poupar tempo e recurso. Porém, esta decisão pode limitar o desenvolvimento do jogo. Por exemplo, ao selecionar uma <i>engine</i> com poucas funcionalidades, pode tornar o jogo mais simples do que o previsto inicialmente na fase de concepção (FULLERTON, 2008; KREMERS, 2009; SALEN e ZIMMERMAN,</p>

2009; SCHELL, 2008). Para auxiliar na utilização destes recursos o professor pode procurar por tutoriais e exemplos. Antes de iniciar o desenvolvimento utilizando um *game engine* é necessário procurar e testar diversos *gameengines*. O criador de jogos também pode consultar fóruns na internet que descrevem pontos positivos, negativos, funcionalidades e recursos de *game engines*. Assim o professor evitará problemas futuros, como falta de funcionalidades para desenvolver um nível do jogo que exige recursos mais avançados que a *engine* oferece.

#### **Funcionalidades das linguagens de programação para desenvolver jogos**

BlitzBasic	<ul style="list-style-type: none"> <li>• Linguagem simples para o desenvolvimento de jogos e apresenta vários recursos de desenvolvimento.</li> <li>• Baseada em Basic que é uma linguagem conhecida e popular.</li> <li>• Apresenta diversos comandos pré-configurados para criação de distintos tipos de jogos.</li> <li>• Considerada uma linguagem de alta produtividade, porque torna o desenvolvimento de jogos rápido, permitindo a criação de jogos em 2D e 3D sem utilizar bibliotecas ou outros componentes.</li> </ul> <p>(WIKIDOT, 2014)</p>
C	<ul style="list-style-type: none"> <li>• Linguagem muito usada no desenvolvimento de jogos, apresenta bibliotecas, códigos e outros recursos prontos para desenvolver jogos.</li> <li>• A linguagem possui materiais difundidos na internet, o que auxiliam o criador de jogos a tirar dúvidas e consultar exemplos parecidos ao que está desenvolvendo.</li> <li>• Um fator negativo é que a linguagem não é orientada a objeto.</li> </ul> <p>(WIKIDOT, 2014)</p>
C++	<ul style="list-style-type: none"> <li>• Linguagem que apresenta embutida em seu compilador o reconhecimento para a linguagem C, o que permite a utilização das duas linguagens.</li> <li>• O C++ é fortemente utilizado pelos professores de jogos comerciais, tendo</li> </ul>

	<p>uma grande quantidade de materiais disponíveis.</p> <ul style="list-style-type: none"> <li>• Apesar de não apresentar uma biblioteca de jogos integrada como Java e Python, é possível utilizar bibliotecas disponíveis na internet.</li> </ul> <p>(WIKIDOT, 2014)</p>
Java	<ul style="list-style-type: none"> <li>• Linguagem fortemente orientada a objetos.</li> <li>• Por ser uma linguagem interpretada é mais lenta que outras linguagens como C/C++.</li> <li>• Não é uma linguagem amplamente utilizada por desenvolvedores de jogos comerciais para computadores.</li> <li>• Linguagem mais utilizada para desenvolvimento de jogos para celulares.</li> </ul> <p>(WIKIDOT, 2014)</p>
PureBasic	<ul style="list-style-type: none"> <li>• Linguagem de programação baseada na linguagem Basic.</li> <li>• Sua principal característica é a portabilidade para <i>Windows, MacOS e Linux</i>.</li> <li>• Possui bibliotecas para suporte a 2D e 3D o que torna o desenvolvimento de jogos rápido.</li> <li>• Linguagem compilada em nível de máquina, tornando-se muitas vezes mais rápida que linguagem C++.</li> </ul> <p>(WIKIDOT, 2014)</p>
Python	<ul style="list-style-type: none"> <li>• Linguagem que apresenta uma sintaxe mais simples, orientada a objeto e de fácil aprendizado.</li> <li>• Por ser uma linguagem de código aberto, existem diversos módulos para serem utilizados e torna possível o desenvolvimento de jogos em 2D e 3D.</li> </ul> <p>(WIKIDOT, 2014)</p>
<b>Funcionalidades das <i>game engine</i> para desenvolver jogos</b>	
Angle	<ul style="list-style-type: none"> <li>• <i>Game engine</i> desenvolvida em Java,</li> <li>• Utilizada para <i>smartphone</i> Android.</li> <li>• <i>Engine</i> focada para desenvolver jogos em 2D utilizando OpenGL (<i>Open</i></li> </ul>



	<p><i>Graphics Library</i>).</p> <ul style="list-style-type: none"> <li>Os jogos desenvolvidas por esta <i>engine</i> são rápidos e oferece diversos tutoriais na internet.</li> </ul> <p>(WIKIDOT, 2014, ANGLE, 2009).</p>
ClanLib	<ul style="list-style-type: none"> <li><i>Game engine</i> para desenvolvimento de jogos em 2D e 3D.</li> <li><i>Engine open source</i> que permite o desenvolvimento de jogos nos sistemas operacionais <i>Windows, MacOs e Linux</i>.</li> <li>Pode ser utilizada para desenvolver jogos <i>móvil</i>, porém apresenta suporte parcial para este tipo de plataforma.</li> <li><i>Engine</i> principalmente aplicada a jogos de computadores, sendo desenvolvida em C++.</li> </ul> <p>(WIKIDOT, 2014, CLANLIB, 2015)</p>
CryEngine	<ul style="list-style-type: none"> <li><i>Game engine</i> de alta performance para desenvolver jogos para consoles (vídeo-game) e computadores.</li> <li>Suporta o desenvolvimento de jogos com gráficos em 3D de alta definição.</li> <li>Apresenta como recurso a utilização de Inteligência Artificial.</li> <li>Desenvolvida em C++.</li> </ul> <p>(WIKIDOT, 2014, CRYENGINE, 2015)</p>
IrrLicht	<ul style="list-style-type: none"> <li><i>Game engine</i> desenvolvida para jogos de tempo real em 2D e 3D, desenvolvida em C++.</li> <li>Faz renderização (processamento final de gráficos) em 3D utilizando DirectX e OpenGL.</li> <li>Desenvolve jogos para os sistemas operacionais <i>Windows, Mac OS, Linux e Solaris</i>.</li> <li>Apresenta uma ampla quantidade de bibliotecas para consulta.</li> <li>Permite que as cenas do jogo apareçam no jogo em harmonia.</li> <li>Permite a importação de arquivos no formato: Maya (.obj), 3DStudio (.3ds),</li> </ul>

	<p>COLLADA (.dae), Blitz3D (.b3d), Milkshape (.ms3d), Quake 3 levels (.bsp), Quake2 models (.md2), Microsoft DirectX, Windows Bitmap (.bmp), Portable Network Graphics (.png), Adobe Photoshop (.psd), JPEG File Interchange Format (.jpg), Truevision Targa (.tga), ZSoft Painbrush (.pcx).</p> <p>(IRRLICHT, 2015)</p>
OGRE	<ul style="list-style-type: none"> <li>• <i>Game engine</i> flexível, orientada a objetos e <i>open source</i>.</li> <li>• Permite o desenvolvimento de jogos 3D.</li> <li>• Desenvolvida em C++ e tem suporte a bibliotecas Direct3D e OpenGL.</li> <li>• Apresenta interface simples e documentação completa.</li> <li>• Desenvolve jogos para os sistemas operacionais <i>Windows, Linux, MacOS, Android, iOS</i> e <i>Windows Phone</i>.</li> <li>• Permite que sejam carregadas imagens do formato PNG, JPEG, TGA, BMP e DDS.</li> </ul> <p>(WIKIDOT, 2014, OGRE, 2015)</p>
UDK	<ul style="list-style-type: none"> <li>• <i>Game engine</i> desenvolvida para produção de jogos em 2D e 3D.</li> <li>• Desenvolvida C++.</li> <li>• Permite a produção de jogos de tempo real e multipayers.</li> <li>• Tem suporte ao DirectX (<i>application programming interfaces – API</i>) oferecendo suporte a vídeo, multimídia, programação de jogos e vídeo (plataforma da <i>Microsoft</i>).</li> <li>• Oferece aos criadores de jogos suporte a Inteligência Artificial.</li> <li>• Permite a integração com hardwares como placas de vídeo NVIDIA.</li> <li>• Permite a criação de animações gráficas utilizando um editor visual interativo.</li> </ul> <p>(WIKIDOT, 2014, UDK, 2015).</p>
Unity	<ul style="list-style-type: none"> <li>• <i>Game engine</i> mais popular entre os desenvolvedores que não estão vinculados a empresas que desenvolvem jogos comercialmente.</li> </ul>

		<ul style="list-style-type: none"> <li>• Suas principais características são baixo custo, variedade de sistemas operacionais que suportam os jogos desta <i>engine</i>, possibilidade de jogar via internet.</li> <li>• Pode ser utilizada por diferentes plataformas como computadores, console, <i>mobile</i> e TV.</li> <li>• Apresenta uma grande quantidade de comunidades, bibliotecas e tutoriais disponíveis na internet.</li> </ul> <p>(WIKIDOT, 2014, UNITY, 2015)</p>
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P3.1.1 - Requisitos do jogo; P3.2.1 – Concepção do jogo	
	<b>Produto de saída:</b> P3.3.1 - Linguagens de programação ou <i>game engine</i> de jogo digital	
	Linguagem de programação	Linguagens de programação possíveis para serem utilizadas no desenvolvimento do jogo, por exemplo: Java, JavaScript, HTML5, CSS, C, C++, C#, .Net, pascal, PHP.
	<i>Game engine</i>	<p>Existem diversas <i>engines</i> de jogos disponíveis, sejam elas <i>free</i> ou pagas. Porém, antes de definir qual <i>engine</i> utilizar é necessário fazer um levantamento dos jogos que cada tipo de <i>engine</i> pode desenvolver, as bibliotecas disponíveis para baixar ou comprar, os fóruns de discussão para tirar dúvidas.</p> <p>A escolha de uma <i>engine</i> sem muitos recursos, torna o desenvolvimento do jogo mais rígido e com as funcionalidades limitadas a partir do que está disponível na <i>engine</i> de jogo.</p> <p>São exemplos de <i>game engines</i>: 3D GameStudio, BGE (Blender), C4 Engine, DX Studio, Esperient Creator, Game Maker, Irrlicht, NeoAxis 3D Engine, Panda3D, RPG Maker, RPG Toolkit, Torque Game Engine, Truevision3D, Unity, Unreal Engine, Crystal Space, jME, OGRE, OpenSceneGraph.</p>

<b>Atividade</b>	<b>A3.3.2 Produzir ilustrações dos elementos do jogo</b>
<b>Descrição</b>	<p>Esta atividade tem por objetivo produzir as ilustrações que representam os elementos do jogo. São considerados elementos do jogo os objetos, cenários e personagens. Os elementos podem ser divididos em elementos de jogos digitais e não-digitais (FULLERTON, 2008; CHANDLER, 2012).</p> <p>As ilustrações típicas de elementos de jogos digitais são: cenários, personagens, artefatos, objetos do jogo. As ilustrações típicas de elementos de jogos não-digitais são: tabuleiro, cartas, dados, “peões”, fichas (peças do jogo), peças de montar (Lego).</p> <p>A presente atividade não é obrigatória. Porque existem situações do processo de desenvolvimento de jogos onde, por exemplo, a linguagem de programação ou <i>game engine</i> disponibiliza os elementos do jogo necessário, deste modo não é necessário criar as ilustrações dos elementos porque eles já estão prontos.</p>
<b>Tarefa</b>	<p><b>3.3.2 Produzir as ilustrações dos elementos do jogo</b></p> <p>Nesta tarefa os elementos do jogo concebidos em <i>P3.2.1 – Concepção do jogo</i> são criados em forma de ilustrações. Os elementos podem ser criados utilizando papel &amp; caneta ou então recursos tecnológicos, como ferramentas digitais para prototipação ou edição de imagens.</p> <p>O design do jogo pode ser realizado de modo iterativo, iniciando com elementos mais simples, por exemplo, criados em papel &amp; lápis, e depois cria-se os elementos mais complexos, utilizando ferramentas para prototipação.</p> <p>A utilização destas ferramentas também podem ser aplicadas para situações onde baixa-se imagens sem direitos autorais da internet e utiliza-se as ferramentas para editar as imagens. São exemplos de sites para baixar imagens sem direitos autorais:</p> <ul style="list-style-type: none"> <li>• Pixabay: <a href="https://pixabay.com/pt">https://pixabay.com/pt</a></li> </ul>

- FreePik: <http://br.freepik.com>
- GettyImages: [www.gettyimages.pt/creative-images/royaltyfree](http://www.gettyimages.pt/creative-images/royaltyfree)

A tabela abaixo apresenta algumas ferramentas para criação de imagens e destaca-se algumas funcionalidades.

Ferramentas para edição de imagens	
Adobe Illustrator	<ul style="list-style-type: none"> <li>• Ferramenta para criação e edição de imagens da empresa Adobe, sendo uma das ferramentas mais utilizadas para produção de imagens.</li> <li>• Ferramenta proprietária.</li> <li>• Oferece diversos recursos e efeitos para trabalhar com imagens.</li> <li>• Ferramenta utilizada amplamente por designers.</li> <li>• Oferece aos usuários 40 milhões de imagens de alta qualidade para os usuários.</li> <li>• Disponibiliza diversos tutoriais.</li> <li>• A edição de imagens é em vetor, deste modo todos os elementos criados podem ser editados separadamente.</li> <li>• Sistema operacional: <i>Windows</i> e <i>MacOS</i>.</li> </ul> (ILUSTRATOR, 2015)
Adobe Photoshop	<ul style="list-style-type: none"> <li>• Editor de imagens da empresa Adobe, focado principalmente para edição de imagens.</li> <li>• Diferentemente do Illustrator, ele não cria imagens em vetor.</li> <li>• Excelente ferramenta para aplicar filtros e efeitos em imagens prontas.</li> <li>• Ferramenta proprietária, com versão de avaliação.</li> <li>• Ferramenta consolidada entre os <i>designers</i> por oferecer diversos recursos técnicos para manipulação de imagens.</li> </ul> (PHOTOSHOP, 2015)

Inkscape	<ul style="list-style-type: none"> <li>• Ferramenta para criação e edição de imagens.</li> <li>• Ferramenta <i>free</i>.</li> <li>• Oferece suporte para diferentes tipos de arquivos.</li> <li>• Considerada ferramenta flexível.</li> <li>• Oferece recursos para efeitos visuais.</li> </ul> <p>(INKSCAPE, 2015)</p>
GIMP	<ul style="list-style-type: none"> <li>• Ferramenta para edição de imagens.</li> <li>• Opção <i>free</i> para o Photoshop.</li> <li>• Oferece diversos recursos técnicos para criação de efeitos e filtros em imagens.</li> <li>• Está disponível para os sistemas operacionais: <i>Windows, Linux, MacOS, Sun OpenSolaris e FreeBSD</i>.</li> </ul>
<b>Ferramentas para prototipação</b>	
Axure RP Pro	<ul style="list-style-type: none"> <li>• Editor especializado na produção de protótipos para computadores e <i>mobile</i>.</li> <li>• Ferramenta proprietária.</li> <li>• Versão <i>trial</i> de 30 dias.</li> <li>• Disponibiliza diversos recursos e efeitos, como variedade de cores, gradientes, preenchimentos, caixas de texto configuráveis.</li> <li>• Sistema operacional: <i>Windows e MacOS</i>.</li> </ul> <p>(AXURE, 2015)</p>
Lumzy	<ul style="list-style-type: none"> <li>• Editor para criação de protótipos e <i>MockUp</i> (modelos em escala real).</li> <li>• Ferramenta online e <i>free</i>.</li> <li>• Permite o compartilhamento de projetos.</li> <li>• Permite a criação de protótipos utilizando <i>drag'n drop</i>.</li> <li>• Permite a simulação do protótipo criado.</li> </ul> <p>(LUMZY, 2015)</p>
Maqetta	<ul style="list-style-type: none"> <li>• Editor para produção de protótipos.</li> </ul>

- Sistema operacional: *MacOS*.
- Ferramenta *Open Source*.
- Focado para computadores, mas também pode criar protótipos para *mobile*.

Para cada elemento criado, define-se nome e tipo de elemento. Para os elementos do tipo personagem deve-se identificar o papel. Existem diversos tipos de papéis que um personagem pode ter, como: gerente de projetos, testador, programador.

Para elementos do tipo cenário, deve-se identificar a sequência do cenário, informando a ordem entre os cenários criados.

Para elementos do tipo cenário, deve-se identificar os níveis (também chamados de fases) do jogo. Um nível de jogo pode ser formado por diversos cenários. Tipicamente o jogo é produzido de forma a tornar possível alcançar os objetivos do jogo (**P3.2.1 – Conceção do jogo**) ao longo dos níveis do jogo. Por exemplo, quando o jogador completa o objetivo do jogo parcialmente a medida que vence cada nível do jogo.

Para os elementos concebidos, serão criadas as ilustrações utilizando papel & caneta, ou ferramentas como as sugeridas na tabela anterior.

#### **Acessibilidade**

Em 2010, cerca de 8,3% da população brasileira apresentava pelo menos um tipo de deficiência severa, podendo ser visual, auditiva, motora, ou mental (OLIVEIRA, 2012). Deste modo, é necessário que ao realizar o design de um jogo considere-se aspectos gerais de acessibilidade.

A seguir apresenta-se recomendações de acessibilidade para jogos digitais (BARLET e SPOHN, 2012):

#### **Mobilidade**

- Comandos reconfiguráveis
- Configuração da sensibilidade da câmera, *mouse* ou teclados

- Funções adequadas na tela do teclado
- Sem botões misturados
- Não haver necessidade de precisão de movimento do *mouse* ou teclado
- Permitir jogar apenas com o *mouse*
- Permitir jogar apenas com o teclado
- Poder mover a interface do usuário
- Ter níveis de dificuldade
- Ter *help* para o jogo

**Visual**

- Não ter elementos chaves importantes do jogo na cor vermelha ou verde
- Ter opção de cores para daltonismo
- Permitir a mudança de fonte
- Permitir mudar o tamanho e tipo da fonte
- Poder ser visualizado em alto contraste
- Ter legendas fáceis de ler
- Ter menu fácil de ler e usar

**Audição**

- Ser possível utilizar legendas para sons
- Todos os sinais de áudio são acompanhados por sinais visuais
- Permitir concluir o jogo com êxito, sem utilizar o som

A seguir são apresentados aspectos fundamentais para impressão de matérias considerando a acessibilidade (CNIB, 2015). Eles podem ser utilizados para impressão de materiais utilizados nos jogos não-digitais:

- **Contraste:** usar cores de alto contraste entre texto e plano de fundo (*background*). Bons exemplos são preto ou azul sobre um fundo branco.
- **Tipo de cor:** material impresso é mais legível em preto e branco. Se utilizar texto colorido,









	<p>restringa-os a títulos, cabeçalhos ou materiais que devem ser destacados.</p> <ul style="list-style-type: none"> <li>• <b>Tamanho da fonte:</b> manter o texto de preferência grande, entre 12 a 18 pontos, dependendo do tamanho da fonte. Também é interessante permitir que os jogadores possam escolher o tamanho da fonte.</li> <li>• <b>Espaçamento:</b> manter espaçamento de linhas entre 25 a 30 por cento. Para permitir que os jogadores possam mover os olhos mais facilmente entre as linhas.</li> <li>• <b>Fontes:</b> evitar fontes complexas ou decorativas, escolhendo sempre fontes padrões e que os jogadores estão habituados e reconhecerão facilmente.</li> <li>• <b>Negrito:</b> escolher fontes mais fortes, evitando utilizar o <i>itálico</i>. Ao enfatizar uma palavra, utiliza-se o negrito.</li> <li>• <b>Espaço entre letras:</b> manter espaço entre as letras, evitando espaços pequenos, dificultando a leitura.</li> <li>• <b>Colunas:</b> separar o texto em colunas para evitar maiores movimentos dos olhos.</li> <li>• <b>Margens:</b> manter margens nas páginas, evitando escrever nas bordas da página.</li> <li>• <b>Plano de fundo:</b> utilizar acabamentos foscos ou não-brilhantes para reduzir o brilho. Reduzir as distrações com plano de fundo complicados ou marca d'água difíceis de entender.</li> <li>• <b>Design limpo e simplicidade:</b> realizar a distinção das cores, usando fontes e formas tornando o material mais fácil de compreender.</li> </ul> <p>(FULLERTON, 2008; KREMERS, 2009; SALEN e ZIMMERMAN, 2009; SCHELL, 2008)</p>
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P3.1.1 - Requisitos do jogo; P3.2.1 – Concepção do jogo; P3.3.1 - Linguagens e/ou engine de jogo digital.
	<b>Produto de saída:</b> P3.3.2 - Elementos do jogo
	<b>Elementos do jogo</b>
Tipo	Tipos de elementos de jogo: personagens, artefatos, cenários e menus.

Nome	Nome do elemento do jogo.
Papel	No caso do elemento de jogo do tipo personagem, deve-se inserir o papel que o personagem executa.
Sequência	Para elementos do tipo cenário, deve-se informar a sequência entre os cenários criados.
Nível	Para elementos do tipo cenário, deve-se informar o nível (fase) de um determinado cenário).
Ilustração	Ilustração do elemento do jogo.
<b>Acessibilidade</b>	
Mobilidade (jogo digital)	<ul style="list-style-type: none"> <li>• Comandos reconfiguráveis</li> <li>• Configuração da sensibilidade da câmera, <i>mouse</i> ou teclados</li> <li>• Funções adequadas na tela do teclado</li> <li>• Sem botões misturados</li> <li>• Não haver necessidade de precisão de movimento do <i>mouse</i> ou teclado</li> <li>• Permitir jogar apenas com o <i>mouse</i></li> <li>• Permitir jogar apenas com o teclado</li> <li>• Poder mover a interface do usuário</li> <li>• Ter níveis de dificuldade</li> <li>• Ter <i>help</i> para o jogo</li> </ul>

	Visual (jogo digital)	<ul style="list-style-type: none"><li>• Não ter elementos chaves importantes do jogo na cor vermelha ou verde</li><li>• Ter opção de cores para daltonismo</li><li>• Permitir a mudança de fonte</li><li>• Permitir mudar o tamanho e tipo da fonte</li><li>• Poder ser visualizado em alto contraste</li><li>• Ter legendas fáceis de ler</li><li>• Ter menu fácil de ler e usar</li></ul>
	Audição (jogo digital)	<ul style="list-style-type: none"><li>• Ser possível utilizar legendas para sons</li><li>• Todos os sinais de áudio são acompanhados por sinais visuais</li><li>• Permitir concluir o jogo com êxito, sem utilizar o som</li></ul>

	Material impresso (jogo não-digital)	<ul style="list-style-type: none"><li>• <b>Contraste:</b> usar cores de alto contraste entre texto e plano de fundo (<i>background</i>). Bons exemplos são preto ou azul sobre um fundo branco.</li><li>• <b>Tipo de cor:</b> material impresso é mais legível em preto e branco. Se utilizar texto colorido, restrinja-os a títulos, cabeçalhos ou materiais que devem ser destacados.</li><li>• <b>Tamanho da fonte:</b> manter o texto de preferência grande, entre 12 a 18 pontos, dependendo do tamanho da fonte. Também é interessante permitir que os jogadores possam escolher o tamanho da fonte.</li><li>• <b>Espaçamento:</b> manter espaçamento de linhas entre 25 a 30 por cento. Para permitir que os jogadores possam mover os olhos mais facilmente entre as linhas.</li><li>• <b>Fontes:</b> evitar fontes complexas ou decorativas, escolhendo sempre fontes padrões e que os jogadores estão habituados e reconhecerão facilmente.</li><li>• <b>Negrito:</b> escolher fontes mais fortes, evitando utilizar o <i>itálico</i>. Ao enfatizar uma palavra, utiliza-se o negrito.</li><li>• <b>Espaço entre letras:</b> manter espaço entre as letras, evitando espaços pequenos, dificultando a leitura.</li><li>• <b>Colunas:</b> separar o texto em colunas para evitar maiores movimentos dos olhos.</li><li>• <b>Margens:</b> manter margens nas páginas, evitando escrever nas bordas da página.</li><li>• <b>Plano de fundo:</b> utilizar acabamentos foscos ou não-brilhantes para reduzir o brilho. Reduzir as distrações com plano de fundo complicados ou marca d'água difíceis de entender.</li><li>• <b>Design limpo e simplicidade:</b> realizar a distinção das cores, usando fontes e formas tornando o material mais fácil de compreender.</li></ul>
--	--------------------------------------	--

<b>Atividade</b>	<b>A3.3.3 Modelar o jogo</b>	
<b>Descrição</b>	Esta atividade tem por objetivo a modelagem do fluxograma dos níveis do jogo, das bibliotecas de jogo tipicamente adicionada à <i>game engine</i> , dos <i>feedbacks</i> educacionais e dos diálogos dos personagens. A atividade aplica-se tanto para jogos digitais, quanto para não-digitais.	
<b>Tarefa</b>	<b>3.3.3 Modelar o jogo</b>	
	<p><b>Fluxograma dos níveis do jogo</b></p> <p>Esta atividade inicia com a produção do fluxograma de níveis (<i>level flowchart</i>) do jogo. O fluxograma é a representação gráfica de um determinado fluxo de operações (HOUAISS, 2009). Nesta tarefa é necessário criar fluxograma de progressão do personagem no jogo. Entende-se a progressão por passos ou etapas que o personagem deve realizar em cada nível do jogo, normalmente objetivando a vitória do jogo (KREMER, 2009).</p> <p>Com objetivo de auxiliar a criação do fluxograma de níveis, a tabela abaixo descrever os principais elementos que compõem o fluxograma do jogo (CACOO, 2015):</p>	
	<b>Elementos do Fluxograma dos Níveis do Jogo</b>	
		Indica o início do nível do jogo.
	Indica ações a serem executadas.	
	Indica um ponto para tomada de decisão. Apresenta duas alternativas de resposta, tipicamente verdadeiro ou falso.	

	Indica os documentos utilizados no jogo.
	Indica o sentido do fluxo de dados.
	Indica o fim do nível do jogo

A seguir apresenta um exemplo de fluxograma de um nível de jogo. Ele descreve um nível de exemplo, onde o jogador entra em uma sala e precisa explorar o local em busca de novas habilidades. Quando adquire habilidades novas, pode responder questões e procurar a chave para ir para a próxima sala do ambiente onde está. O nível termina quando o personagem respondeu corretamente as questões e encontrou a chave certa da próxima sala.



***Biblioteca externa***

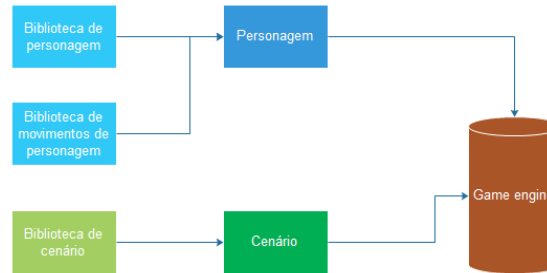
A biblioteca externa é o código-fonte ou pacotes de códigos-fonte que são adicionados à linguagem de programação ou *game engine*. Portanto, aplica-se para jogos digitais.

Uma biblioteca externa pode acelerar o desenvolvimento do jogo, por exemplo, ao adicionar personagens ou cenários prontos.

Caso o jogo utilize uma *game engine* é necessário representar graficamente as bibliotecas de jogo adicionadas à ela. A representação das bibliotecas concentra-se apenas naquelas que foram adicionadas à *game engine*. Por exemplo, no caso da *game engine Unity* as bibliotecas são chamadas de *asset* e os criadores de jogos podem comprar ou baixar diversos *asset* no próprio site da *game engine* e utilizar no jogo que está sendo desenvolvido.

A figura a seguir apresenta um exemplo da representação gráfica de duas bibliotecas da *game engine*. A primeira é referente aos personagens do jogo que foram adicionadas à *game engine* e a segunda aos cenários que também foram adicionados. Ao adicionar estas bibliotecas o criador de jogos não precisa criar os personagens e cenários, porque eles estão prontos.

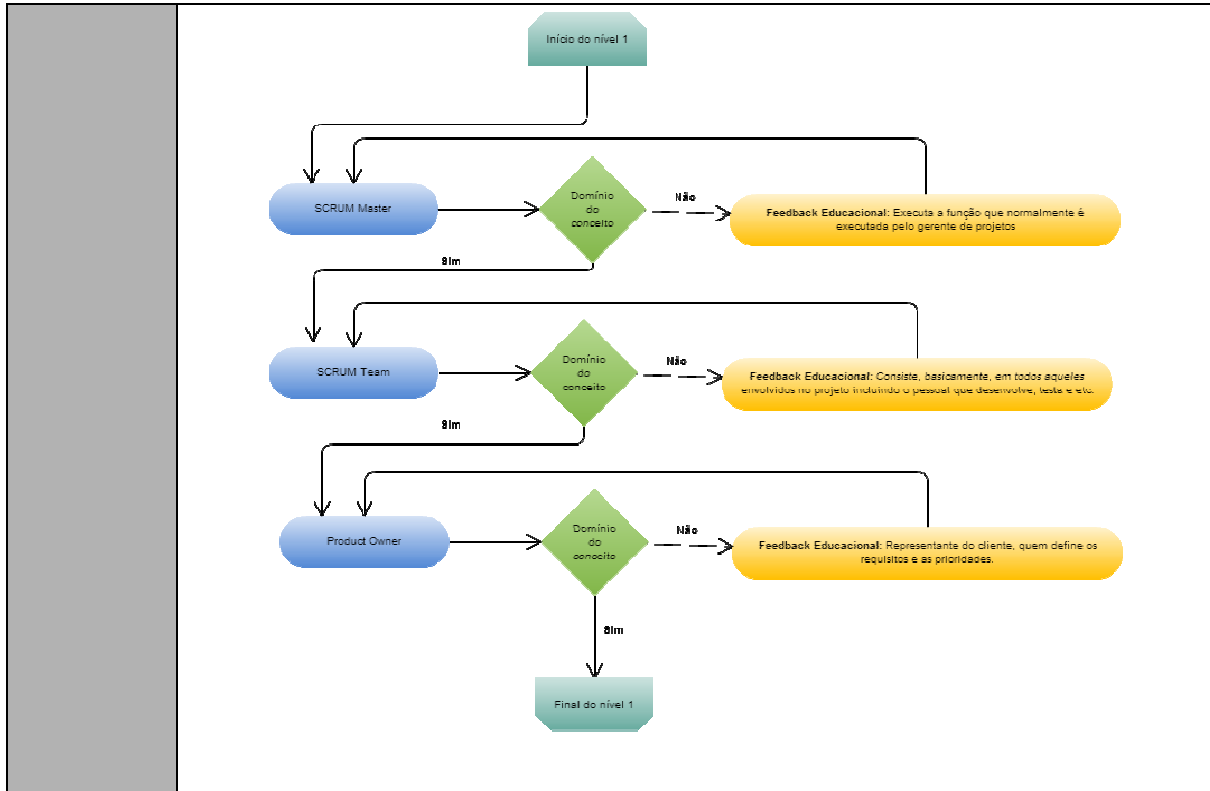




### ***Feedback educacional***





O *feedback* educacional foi criado na atividade de concepção (**P3.2.1 – Concepção do jogo**), nesta tarefa o *feedback* é modelado por meio de fluxograma. Será criado um fluxograma para cada nível do jogo, informando os conceitos que serão abordados, caso o jogador erre ou acerte algum conceito o fluxograma indicará qual o próximo passo do jogador dentro do nível do jogo.

A seguir apresenta-se um exemplo de *feedback* educacional da primeira fase do jogo SCRUM-Scape (CAMARGO, 2013). Neste exemplo, mostra que no primeiro nível do jogo é necessário ter o domínio de três conceitos do SCRUM. Caso o jogador não tenha domínio do conteúdo será apresentado um *feedback* educacional.

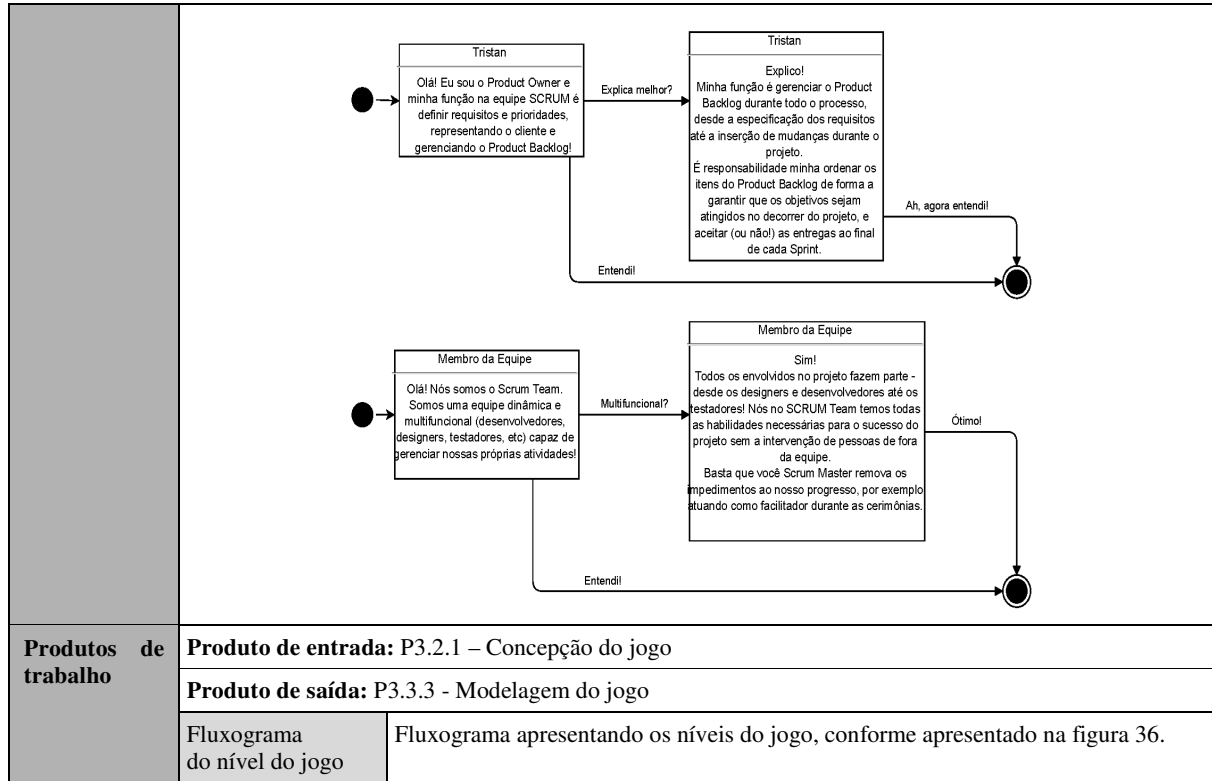


### Diálogo do jogo

Os diálogos presentes no jogo precisam ser descritos para permitir a correção de possíveis erros de conteúdo. Ele é realizado por meio da criação de um diagrama de estado (OMG, 2015). A tabela a seguir apresenta os principais elementos do diagrama para documentar o diálogo do jogo.

	Elemento que indica início do diálogo.
	Elemento que é inserido o nome do personagem e a descrição do diálogo.
	Elemento que indica direção do diálogo.
	Elemento que indica o final do diálogo.

A seguir apresenta-se um exemplo de modelagem do diálogo do jogo. Neste exemplo é utilizado um extrato do diálogo do jogo SCRUM'ed (SCHNEIDER, 2015).



<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P3.2.1 – Concepção do jogo
	<b>Produto de saída:</b> P3.3.3 - Modelagem do jogo
	Fluxograma do nível do jogo
	Fluxograma apresentando os níveis do jogo, conforme apresentado na figura 36.

	Biblioteca externa	Aplicado para jogos digitais que utilizam para o desenvolvimento bibliotecas externas à linguagem de programação ou <i>game engine</i> . Um exemplo é apresentado na figura 37.
	<i>Feedback</i> educacional	<i>Feedback</i> dos conceitos educacionais do jogo, criado por nível do jogo. Um exemplo é apresentado na figura 38.
	Diálogo do jogo	O diálogo do jogo representa as falas dos personagens ao longo do jogo. Um exemplo é apresentado na figura 39.

### F3.4 IMPLEMENTAÇÃO DO JOGO

<b>Atividade</b>	<b>A3.4.1 Produzir elementos do jogo</b>	
<b>Descrição</b>	Esta atividade tem por objetivo a produção dos elementos do jogo. No caso de jogos digitais os elementos são codificados utilizando a linguagem de programação ou <i>game engine</i> definida em <b>P3.3.1 - Linguagens de programação ou game engine de jogo digital</b> . No caso dos jogos não-digitais, os elementos são criados “ <i>fisicamente</i> ”.	
<b>Tarefa</b>	<b>3.4.1 Produzir elementos do jogo</b>	
	Nesta tarefa, os elementos do jogo são produzidos. A tarefa inicia com a identificação dos nomes dos elementos dos jogos, os quais foram definidos na fase de concepção do jogo (F3.2 Conceber jogo).	
	<b>Jogo digital</b>	A implementação é a codificação das especificações técnicas e requisitos para criação de um jogo (SWEBOK, 2004). Nesta tarefa, todo elemento deve ser codificado e ser salvo. Por exemplo, ao desenvolver um personagem deve-se inserir o código que permite ele caminhar pelo cenário, interagir com outros personagens e manipular objetos do cenário.

Todo código-fonte criado deve ser comentado. O comentário é necessário para que outros criadores de jogos saibam a funcionalidade de cada trecho de código, mas também para que o próprio criador do jogo não esqueça com o tempo qual a função de um determinado código.

Sempre que o código-fonte for modificado é necessário atualizar estas informações nesta atividade. Mantendo deste modo o código-fonte desta atividade sempre atualizado.

Para todo código inserido, deve-se inserir também a versão do código-fonte. Para auxiliar no controle de versões do código-fonte, é possível utilizar ferramentas de controle de versão, como:

- **Concurrent Version System (CVS):** ferramenta *free* para controle de versão bastante difundida entre os desenvolvedores.
- **Subversion (SVN):** ferramenta *free* para controle de versão, considerada mais eficiente que o CVS.
- **Visual SourceSafe (VSS):** ferramenta proprietária de controle de versão da Microsoft.


Tipicamente as ferramentas que controlam a versão de código-fonte contêm um número de versão para o arquivo de código-fonte. Este número deve ser informado nesta tarefa.

A utilização destas ferramentas também permite que sejam criados *backups* para todos os códigos-fonte. Deste modo, versões anteriores do código-fonte podem ser reutilizadas.

	<p>Também é necessário informa nesta tarefa o endereço eletrônico para baixar os fontes do jogo. Caso o jogo apresente um arquivo executável (.EXE), deve-se descrever os passos para baixá-lo.</p>																								
<p><b>Jogo não-digital</b></p>	<p>Para a criação dos elementos para jogos não-digitais, deve-se descrever os materiais que são utilizados para compor o elemento do jogo. São exemplos de materiais do jogo: tabuleiros, fichas, cartas, dados.</p> <p>Caso o elemento seja criado digitalmente utilizando ferramentas como <i>Adobe Illustrator</i> ou <i>InkScapei</i> é necessário adicionar esta imagem digital nesta tarefa.</p> <p>Caso o elemento do jogo seja impresso após a criação digital, deve-se definir o tamanho do papel. Tipicamente utiliza-se os seguintes tamanhos do papel (em milímetros) (PROFCARDY, 2015):</p> <table border="1" data-bbox="817 543 1050 890"> <thead> <tr> <th colspan="2">A</th> </tr> </thead> <tbody> <tr> <td>A0</td> <td>841 × 1189</td> </tr> <tr> <td>A1</td> <td>594 × 841</td> </tr> <tr> <td>A2</td> <td>420 × 594</td> </tr> <tr> <td>A3</td> <td>297 × 420</td> </tr> <tr> <td>A4</td> <td>210 × 297</td> </tr> <tr> <td>A5</td> <td>148 × 210</td> </tr> <tr> <td>A6</td> <td>105 × 148</td> </tr> <tr> <td>A7</td> <td>74 × 105</td> </tr> <tr> <td>A8</td> <td>52 × 74</td> </tr> <tr> <td>A9</td> <td>37 × 52</td> </tr> <tr> <td>A10</td> <td>26 × 37</td> </tr> </tbody> </table>	A		A0	841 × 1189	A1	594 × 841	A2	420 × 594	A3	297 × 420	A4	210 × 297	A5	148 × 210	A6	105 × 148	A7	74 × 105	A8	52 × 74	A9	37 × 52	A10	26 × 37
A																									
A0	841 × 1189																								
A1	594 × 841																								
A2	420 × 594																								
A3	297 × 420																								
A4	210 × 297																								
A5	148 × 210																								
A6	105 × 148																								
A7	74 × 105																								
A8	52 × 74																								
A9	37 × 52																								
A10	26 × 37																								

<p>Ao definir o tamanho do papel, define-se o tipo de papel. A seguir apresenta-se 5 tipos de papeis utilizados popularmente para impressão de materiais (GOMAO, 2015):</p>	
<b>Tipos de papeis</b>	
Jornal	Utilizado para impressão de jornais.
Offset	Utilizado para impressão de livros e revistas, como característica destaca sua durabilidade e resistência à umidade.
Fotográfico	Utilizado para impressão de fotografias, é um papel de maior qualidade. Pode ser encontrado do tipo brilhante ou fosca.
<i>Couchê</i>	Utilizado para impressões de alta qualidade, normalmente utilizado para <i>folders, flyers</i> . Apresenta uma camada que torna o papel mais brilhante.
Revista	Utilizado para impressão de revista, sua qualidade não é seu ponto forte. Tem a característica de envelhecer rapidamente.
<p>Ao definir o tipo do papel, é necessário definir a gramatura do papel. A gramatura significa grama por metros quadrados. Ou seja, a gramatura define a espessura do papel. A figura abaixo apresenta a gramatura dos papeis utilizados normalmente para impressão:</p>	



	 <p>Os materiais como fichas, dados, cartas, tabuleiros, botões e outros objetos podem ser comprados em papelarias, lojas de brinquedos ou online. A seguir apresenta-se alguns sites para compra de materiais para produção de jogos não-digitais:</p> <ul style="list-style-type: none"><li>• No site <i>The Game Crafter</i> (<a href="https://www.thegamecrafter.com/">https://www.thegamecrafter.com/</a>), jogos de tabuleiro e cartas podem ser criados pelo próprio autor e depois vendidos online.</li><li>• No site Menestrel (<a href="http://menestrel.com.br">http://menestrel.com.br</a>) materiais como baralhos, dominós e tabuleiros podem ser comprados online.</li><li>• No site Mercado Livre (<a href="http://lista.mercadolivre.com.br/jogos/fichas-para-jogos">http://lista.mercadolivre.com.br/jogos/fichas-para-jogos</a>) fichas para jogos podem ser compradas online.</li><li>• No site Leisure Games (<a href="http://www.leisuregames.com">http://www.leisuregames.com</a>) é possível comprar materiais e acessórios para jogos.</li></ul>
	<p><b>Tipo de elemento do jogo</b> Tipicamente os tipos de elementos são: Personagens, Cenários, Artefatos e Menus.</p> <p><b>Nome do elemento do jogo</b></p>

	<p>Nome do elemento do jogo.</p> <p><b>Papel do elemento do jogo</b> Para o tipo de elemento Personagem, deve-se identificar o papel que o personagem possui no jogo.</p> <p><b>Nível do elemento do jogo</b> Para o tipo de elemento Cenário, deve-se identificar o nível (fase) do jogo.</p> <p><b>Sequência do elemento do jogo</b> Para o tipo de elemento Cenário, deve-se identificar a sequência do cenário para cada nível de jogo.</p> <p><b>Imagem do elemento do jogo</b> Nesta tarefa, também deve-se inserir o <i>screenshot</i> dos elementos do jogo criados, caso seja um jogo digital. Ou inserir a imagem (foto) do elemento criado, caso seja um jogo não-digital.</p>
<b>Produtos de trabalho</b>	<p><b>Produto de entrada:</b> P3.2.1 – Concepção do jogo; P3.3.1 - Linguagens de programação ou <i>game engine</i> de jogo digital; P3.3.2 - Elementos do jogo.</p> <p><b>Produto de saída:</b> P3.4.1 - Elementos do jogo implementados</p> <p style="text-align: center;"><b>Jogo Digital</b></p>

Versão do código-fonte	<p>Versão do código-fonte do elemento do jogo. Informação retirada tipicamente do número da versão que a ferramenta de controle de versão definiu.</p> <p>Exemplos de ferramentas de controle de versão:</p> <ul style="list-style-type: none"> <li>• <b>Concurrent Version System (CVS):</b> ferramenta <i>free</i> para controle de versão bastante difundida entre os desenvolvedores.</li> <li>• <b>Subversion (SVN):</b> ferramenta <i>free</i> para controle de versão, considerada mais eficiente que o CVS.</li> <li>• <b>Visual SourceSafe (VSS):</b> ferramenta proprietária de controle de versão da Microsoft.</li> </ul>
Código-fonte do jogo	Código-fonte do elemento do jogo.
Endereço eletrônico	Endereço eletrônico para baixar os códigos-fonte do jogo. Caso o jogo apresenta um executável (.EXE), também deve-se informar o endereço e as instruções para abaixá-lo.
<b>Jogo Não-Digital</b>	
Material	Identificar os materiais que compõem o elemento do jogo. São exemplos de materiais: tabuleiro, fichas, dados, cartas, peças e acessórios.
Papel	Caso o elemento do jogo seja impresso, deve-se identificar o tamanho do papel, o tipo do papel e a gramatura.
Tipo	Tipo do elemento
Nome	Nome do elemento do jogo

	Papel	Papel do tipo de elemento Personagem.
	Nível	Nível (fase) do tipo de elemento Cenário.
	Sequência	Sequência do tipo de elemento Cenário.
	Imagem	Para jogos digitais deve-se apresentar o <i>screenshot</i> do elemento do jogo criado. Para jogos não-digitais deve-se apresentar a imagem “física” do elemento, tipicamente utilizando uma foto digitalizada do elemento.

### F3.5 TESTE DO JOGO

<b>Atividade</b>	<b>A3.5.1 Realizar testes do jogo</b>
<b>Descrição</b>	Esta atividade tem por objetivo realizar testes do jogo desenvolvido. A realização deles busca <i>feedback</i> úteis para melhoria do jogo (SWEBOK, 2004; FULLERTON, 2008; CHANDLER, 2012).
<b>Tarefa</b>	<p><b>3.5.1 Realizar testes do jogo</b></p> <p>Nesta tarefa os testes são realizados com objetivo de identificação erros no jogo. Os testes são atividades realizadas para avaliar a qualidade do produto e para sua melhoria, por meio da identificação de defeitos e problemas (SWEBOK, 2004).</p> <p>Nesta tarefa identifica os seguintes elementos:</p> <ul style="list-style-type: none"> <li>• Tipo de elemento do jogo testado (personagem, cenário, artefato, menu);</li> <li>• Nome do elemento testado;</li> <li>• Nível do jogo testado, aplicado ao tipo de elemento “cenário”;</li> <li>• Funcionalidades testadas (p. ex. movimentação de um personagem; mudança de câmeras do cenário), aplicado ao tipo de elemento “personagem”, “artefato” e “menu”;</li> </ul>

	<ul style="list-style-type: none"> <li>• Performance dos elementos testado (p. ex. cenário com muitos elementos do jogo que torna o jogo lento), aplicado ao tipo de elemento “cenário”;</li> <li>• Problemas identificados no teste (p. ex. o personagem não obedece os comandos do teclado, o cenário apresenta paredes sem textura);</li> <li>• Status dos problemas (pendente, ignorado, solucionado) identificados.</li> <li>• Testador que realiza os testes.</li> </ul> <p>Os testes são realizados inicialmente pelo criador de jogos. Após os testes serem realizados pelo criador de jogos, deve-se selecionar uma amostra do público-alvo, tipicamente ex-alunos da disciplina, para testar o jogo. Esta amostra pode ser reduzida, como, por exemplo, 10 alunos.</p>	
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P3.4.1 - Elementos do jogo implementados	
	<b>Produto de saída:</b> P3.5.1 – Teste do jogo.	
	Tipo de elemento do jogo	Elemento do jogo testado (personagem, cenário, artefato, menu).
	Nome	Nome do elemento.
	Nível	Nível em que o elemento do jogo foi testado, aplicado ao tipo de elemento “Cenário”.
	Funcionalidade	Funcionalidades que foram testados, aplicado ao tipo de elemento “Personagem”, “Artefato” e “Menu”.
	Performance	Performance do jogo que foram testadas, como
	Problema	Problemas identificados no jogo.
Status	Status dos problemas: pendente, ignorado ou resolvido.	

	Testador	Nome do testador. Os testes são realizados tipicamente pelo criador de jogos, porém podem ser selecionados ex-alunos da disciplina da unidade instrucional para testar o jogo.
--	----------	--

#### F4. EXECUÇÃO DA UNIDADE INSTRUCIONAL

<b>Atividade</b>	<b>A4.1 Planejar a execução do jogo</b>
<b>Descrição</b>	Esta atividade tem por objetivo planejar a execução do jogo, planejando as necessidades do Professor, dos alunos e do ambiente onde é aplicado o jogo (DICK e CAREY, 1996, BRANCH, 2009).
<b>Tarefa</b>	<p><b>4.1.1 Planejar as necessidades do Professor</b></p> <p>Nesta tarefa busca-se planejar a execução do jogo levantando as necessidades do Professor. Ela inicia com a identificação dos conteúdos que o Professor deve dominar para aplicar a unidade instrucional (BRANCH, 2009).</p> <p>Tipicamente o jogo é aplicado pelo próprio Professor que desenvolveu o jogo ou orientou um aluno no desenvolvimento. Este Professor possui domínio do conteúdo do jogo, porém em algumas situações a aplicação do jogo pode ser realizada por outro Professor. Neste caso este Professor deve ter domínio do conteúdo do jogo (BRANCH, 2009). Estas situações podem ocorrer no contexto do processo ENgAGED quando um criador de jogos desenvolve um novo jogo educacional e outros Professores queiram aplicar o jogo em suas disciplinas.</p> <p>Abaixo, apresenta-se um exemplo dos conteúdos que o Professor deve ter para aplicação do jogo:</p> <ul style="list-style-type: none"> <li>• Ter conhecimento em Gerência de Projeto de Software.</li> <li>• Ter conhecimento em desenvolvimento ágil utilizando SCRUM.</li> </ul>

Deve-se levantar também os materiais que o Professor utilizará para aplicar o jogo, como os listados abaixo:

- Computador com acesso a internet.
- No computador deve ter instalado o *PowerPoint*.
- Projetor LCD.
- Quadro branco.
- Canetas para quadro branco.
- Apagador para canetas.

Deve-se levantar o tempo total para aplicação do jogo, tempo para apresentação dos objetivos do jogo, tempo para ensinar os alunos a jogar o jogo, tempo para jogar o jogo e tempo para realizar a avaliação do jogo. Abaixo apresenta um exemplo do levantamento de tempo:

- Duração total para aplicação do jogo: 2 horas.
  - Duração para apresentação dos objetivos: 15 minutos.
  - Duração para ensinar os alunos a jogarem o jogo: 15 minutos.
  - Duração para os alunos jogarem o jogo: 1 hora.
  - Duração para os alunos realizarem a avaliação do jogo: 30 minutos.

Esta tarefa finaliza com o levantamento da data e local onde o jogo será aplicado.

#### **4.1.2 Planejar as necessidades dos aprendizes**

Nesta tarefa busca-se levantar as necessidades dos aprendizes (alunos) objetivando uma participação ativa e interação eficaz com a estratégia instrucional desenvolvida (BRANCH, 2009).

Para realizar o levantamento das necessidades do aprendiz, deve-se identificar a quantidade de alunos matriculados. Abaixo apresenta-se dois exemplos:

- Aplicação de um jogo digital em laboratório

- Quantidade de alunos: 34.
- Os alunos jogarão em duplas.
- Aplicação de um jogo não-digital em sala de aula:
  - Quantidade de alunos: 20.
  - Os alunos jogarão de grupos de 4 alunos (5 grupos).

#### 4.1.3 Planejar as necessidades do ambiente

Nesta tarefa planeja-se as necessidades do ambiente, levantando os materiais e equipamentos que são utilizados para aplicação do jogo em sala de aula ou extraclasse.

Inicialmente deve-se definir se a aplicação do jogo ocorrerá em sala de aula, laboratório de informática ou extraclasse.

Deve-se identificar também as necessidades do ambiente onde o jogo é aplicado, como a quantidade e configuração dos computadores. Abaixo apresenta-se uma tabela com elementos relevantes para aplicação do jogo digitais. Os elementos não são obrigatórios. Por exemplo, para jogos que não utilizam conexão de internet, os elementos “internet” e “navegador” não se aplica.

Computadores	Quantidade de computadores
Sistema operacional	Windows, MacOS, Linux.
Espaço	Espaço livre no <i>Hard Disk</i> (HD)
Memória	Quantidade de memória RAM necessária para rodar o jogo
Processador	Processadores requeridos para rodar o jogo.
Placa de vídeo	Placa de vídeo para rodar o jogo.
Internet	Velocidade da banda de internet: <ul style="list-style-type: none"> <li>● 1 Mb</li> <li>● 2 Mb</li> <li>● 4 Mb</li> </ul>



	<ul style="list-style-type: none"><li>• 5 Mb</li><li>• 8 Mb</li><li>• 10 Mb</li><li>• 20 Mb</li><li>• 30 Mb</li><li>• 50 Mb</li><li>• 100 Mb</li><li>• Outro</li></ul>
Navegador	Negador de internet necessário: <ul style="list-style-type: none"><li>• Internet Explorer</li><li>• Firefox</li><li>• Chrome</li><li>• Safari</li><li>• Opera</li><li>• Outro</li></ul>
Monitor	Resolução do monitor: <ul style="list-style-type: none"><li>• 176 x 144 (px)</li><li>• 320x200 (px)</li><li>• 352 x 288 (px)</li><li>• 480 x 320 (px)</li><li>• 720 x 240 (px)</li><li>• 720 x 480 (px)</li><li>• 768 x 576 (px)</li><li>• 854 x 480 (px)</li><li>• 1920 x 1080 (px)</li><li>• 2560 x 2048 (px)</li><li>• 3840 x 2400 (px)</li></ul>

	Dispositivo para instalação	Dispositivo para instalação do jogo para executáveis (.EXE), sendo em <i>Pen Drive</i> (2 Gb, 4 Gb, 8 Gb, 16 Gb, 32 GB) ou CD/DVD (4.7 Gb, 8.5 Gb, 9.4 Gb, 17,08 Gb).
	<i>Mobile</i>	Sistema operacional para <i>mobilei</i> : <i>Android, Windows Phone, iOS</i> .
	Aplicativo	Aplicativo <i>mobile</i> : Wi-Fi, GPS, navegador <i>Chrome</i> .
	Para jogos não-digitais deve-se identificar os moveis (cadeiras e mesas) e materiais de apoio (canetas, lápis, folhas em branco).	
	Cadeiras	Quantidade de cadeiras utilizadas.
	Mesas	Quantidade de mesas utilizadas.
	Materiais de apoio	Materiais de apoio para aplicação do jogo: folhas em branco, canetas, lápis, clipes, grampeador, borracha, tesoura.
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P3.1.1 - Requisitos do jogo; P3.2.1 – Concepção do jogo; P3.3.1 - Linguagens de programação ou <i>game engine</i> de jogo digital; P3.3.2 - Elementos do jogo; P3.3.3 - Modelagem do jogo; P3.4.1 - Elementos do jogo implementados; P3.5.1 – Teste do jogo	
	<b>Produto de saída:</b> P4.1.1 - Planejar as necessidades do Professor	
	Conteúdo	Conteúdos que o Professor deve dominar para aplicar o jogo.
	Sistema operacional	Sistema operacional do computador que será utilizado pelo Professor para apresentar o jogo.
	Espaço	Espaço livre no <i>Hard Disk</i> (HD) do computador.
	Memória	Quantidade de memória RAM.
	Processador	Processadores requeridos.

Internet	Velocidade da banda de internet.
Navegador	Navegador de internet.
Monitor	Resolução do monitor
<i>Datashow</i>	Marca do <i>datashow</i> utilizado para apresentar o jogo. Também é necessário identificar o tipo de conexão com o computador: HDMI, USB, VGA.
Materiais de apoio	Materiais de apoio: canetas para quadro-branco, quadro-branco, apagador.
Duração total	Duração total da aplicação do jogo.
Duração da apresentação	Duração da apresentação dos objetivos do jogo.
Duração da explicação do jogo	Duração da explicação do funcionamento e comandos do jogo.
Duração para jogar o jogo	Duração em que os alunos jogam o jogo.
Duração para avaliação	Duração para responder o questionário de avaliação do jogo.
<b>Produto de saída:</b> P4.1.2 - Planejar as necessidades dos aprendizes	
Tipo de aplicação	<ul style="list-style-type: none"> <li>• Extraclasse: aplicado na casa do aluno ou outro ambiente, fora da universidade.</li> <li>• Sala de aula: aplicado em sala de aula, utilizando carteiras e cadeiras.</li> <li>• Laboratório: aplicado em laboratório de informática. Neste caso são utilizados os computadores do próprio laboratório para jogar.</li> </ul>
Quantidade de alunos	Quantidade dos alunos que estão matriculados na disciplina.

Divisão dos alunos	<p>Divisão em que os alunos seguirão para jogar o jogo:</p> <ul style="list-style-type: none"> <li>• Individualmente.</li> <li>• Duplas</li> <li>• Trio</li> <li>• Quarteto</li> <li>• Quinteto</li> <li>• Outro</li> </ul> <p>Para divisões em grupos com mais de 2 alunos, deve-se informar a quantidade de grupos. Por exemplo, em uma turma de 30 alunos, pode-se formar 10 grupos de 3 pessoas.</p>
<b>Produto de saída:</b> P4.1.3 - Planejar as necessidades do ambiente	
<b>Jogo digital</b>	
Computadores	Quantidade de computadores para os alunos jogarem o jogo.
Sistema operacional	Sistema operacional para jogar o jogo.
Espaço	Espaço livre no <i>Hard Disk</i> (HD) do computador.
Memória	Quantidade de memória RAM.
Processador	Processadores requeridos.
Placa de vídeo	Placa de vídeo para rodar o jogo.
Internet	Velocidade da banda de internet.
Navegador	Navegador de internet.
Monitor	Resolução do monitor

Dispositivo para instalação	Dispositivo para instalação do jogo para executáveis (.EXE), sendo em <i>Pen Drive</i> (2 Gb, 4 Gb, 8 Gb, 16 Gb, 32 GB) ou CD/DVD (4.7 Gb, 8.5 Gb, 9.4 Gb, 17,08 Gb).
<i>Mobile</i>	Sistema operacional para <i>mobilei</i> : <i>Android, Windows Phone, iOS</i> .
Quantidade	Quantidade de <i>smartphones</i> necessários para aplicação do jogo
Aplicativo	Aplicativos necessários para rodar o jogo: Wi-Fi, GPS, navegador <i>Chrome</i> .
Jogo não-digital	
Mesa	Mesas necessárias para jogar o jogo. Tipicamente tabuleiros precisam de bastante espaço para que todos alunos possam ter acesso ao tabuleiro do jogo, então deve-se verificar a quantidade de mesas para que a aplicação do jogo permita que todos alunos consigam acessar o tabuleiro.
Cadeira	Cadeiras necessárias para os alunos sentarem. Tipicamente a sala de aula onde o jogo é aplicado é a mesma em que a disciplina é realizada. Mas em algumas aplicações do jogo pode-se utilizar uma sala na qual o Professor não conhece, sendo necessário verificar se terá cadeira para todos os alunos.
Material de apoio	Materiais utilizados como apoio à aplicação do jogo, como folhas em branco, caneta, papel, lápis, borracha, calculadora, cliques, grampeador, caneta marca-texto, pincel.

<b>Atividade</b>	<b>A4.2 Instalar o jogo digital</b>	
<b>Descrição</b>	Esta atividade tem por objetivo a instalação do jogo, aplicando-se exclusivamente para jogos digitais.	
<b>Tarefa</b>	<p><b>4.2 Instalar o jogo digital</b></p> <p>Nesta tarefa deve-se informar os passos para instalação do jogo. Caso o jogo seja aplicado em laboratório, deve-se descrever os passos para instalação do jogo, por exemplo, copiando o arquivo executável (jogo) do <i>Windows</i> e salvando em todos os computadores do laboratório. Para salvar o arquivo do jogo para rodar nos computadores, utiliza-se o Pen Drive ou CD/DVD definido na atividade anterior.</p> <p>Deve-se verificar após a instalação do jogo se os alunos terão permissão de acesso de usuário ao jogo.</p> <p>Caso o jogo seja instalado em um servidor para ser jogador online, deve-se descrever os passos para configuração do jogo, o tipo de servidor (<i>apache, tomcat</i>) e habilitar as permissões aos arquivos para que o jogo funcione corretamente. Após a configuração do jogo, deve-se estar o jogo, por exemplo, no laboratório onde o jogo será aplicado. O teste é realizado pelo próprio criador de jogos, tipicamente testando o jogo nos computadores no laboratório onde o jogo será aplicado.</p>	
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P4.1.1 - Planejar as necessidades do Professor; P4.1.2 - Planejar as necessidades dos aprendizes; 4.1.3 - Planejar as necessidades do ambiente.	
	<b>Produto de saída:</b> P4.2 - Instalação do jogo digital	
	Instalação	Instalação do jogo digital, sendo informado os passos para instalar o jogo.

<b>Atividade</b>	<b>A4.3 Executar o jogo</b>
<b>Descrição</b>	Esta atividade tem por objetivo a execução do jogo em sala de aula, laboratório ou extraclasse, levando em

	consideração o planejamento do Professor, dos aprendizes e do ambiente.
<b>Tarefa</b>	<p><b>4.3 Executar o jogo</b></p> <p>Nesta tarefa deve-se aplicar o jogo em sala de aula, laboratório ou extraclasse. Por exemplo, para aplicação do jogo em sala de aula ou em laboratório o Professor utiliza o computador e o <i>Datashow</i> definido em <b>4.1.1 Planejar as necessidades do Professor</b> para fazer a apresentação do jogo.</p> <p>Após o Professor apresentar o jogo aos alunos, deve-se pedir para começarem a jogar o jogo, caso seja uma aplicação em laboratório ou em sala de aula conforme planejado nas tarefas <b>P4.1.2 - Planejar as necessidades dos aprendizes; 4.1.3 - Planejar as necessidades do ambiente.</b></p> <p>Logo após a realização desta tarefa, realiza-se a atividade <b>A5.1 Conduzir avaliação do jogo.</b></p>
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P2.4.1 – Modelo de avaliação revisado; P4.1.1 - Planejar as necessidades do Professor; P4.1.2 - Planejar as necessidades dos aprendizes; 4.1.3 - Planejar as necessidades do ambiente; P4.2 - Instalação do jogo digital; P5.1 – Resultado da avaliação.
	<b>Produto de saída:</b> P4.3 - Execução do jogo

	Execução	<p>Deve-se descrever como foi aplicação do jogo, do ponto de vista do que foi planejando. Informando se os recursos tecnológicos planejados funcionaram corretamente.</p> <p>Caso o jogo tenha sido rodado nos computadores do laboratório da universidade, deve-se descrever se houveram problemas, como computador que não rodou o jogo.</p> <p>Caso o jogo seja um não-digital, deve-se descrever se possíveis problemas com a sala de aula, por exemplo, as mesas disponíveis eram pequenas para o tabuleiro utilizado no jogo.</p>
--	----------	---

## F5. AVALIAÇÃO DO JOGO

<b>Atividade</b>	<b>A5.1 Conduzir avaliação do jogo</b>
<b>Descrição</b>	<p>Esta atividade tem por objetivo realizar a avaliação do jogo educacional. Esta atividade é uma sequência da atividade <b>A2.4 Revisar o modelo MEEGA</b> da fase <b>F2 Projeto da Unidade Instrucional</b>.</p> <p>Busca-se nesta atividade realizar a coleta de dados após a execução do jogo educacional realizado na atividade <b>A4.3 Executar o jogo</b>.</p>
<b>Tarefa</b>	<p><b>5.1 Conduzir avaliação do jogo</b></p> <p>Nesta tarefa é realizada a avaliação do jogo. Considerando como modelo padrão para o processo ENgAGED o modelo MEEGA (SAVI, 2011), utiliza-se o questionário apresentado no produto de saída <b>P2.4.1 – Modelo de avaliação revisado</b>.</p>



Portanto, para realizar a avaliação do jogo, é entregue aos alunos uma cópia do questionário de avaliação do MEEGA (SAVI, 2011). Depois os dados são tabulados na planilha eletrônica definida pelo MEEGA, acessível no link: <http://www.gqs.ufsc.br/avaliacao-de-jogos-educacionais>.

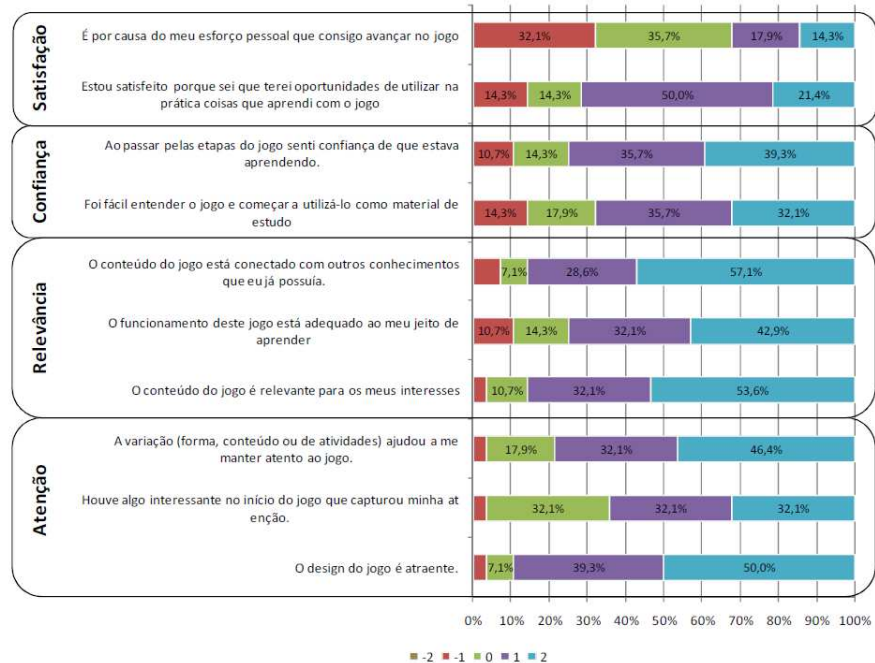
A figura 16 apresenta um exemplo de dados coletados da avaliação de um jogo educacional, inserido na planilha proposta pelo modelo MEEGA (SAVI, 2011).

		JOGO 3 – QUESTÕES																	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16		
Respostas dos alunos	Turma 1	1	2	1	2	0	-1	2	0	2	1	1	2	2	1	2	2	2	
		2	-1	0	1	2	1	1	1	1	1	2	2	1	0	2	2	2	
		3	2	2	2	2	1	2	2	2	2	2	2	2	2	2	2	2	
		4	1	2	2	1	2	1	1	2	2	2	2	2	1	1	2	2	2
		5	1	1	1	2	1	1	1	2	2	1	2	2	2	2	2	1	2
		6	2	2	1	1	2	0	1	0	0	1	2	2	1	2	1	2	2
		7	0	1	1	1	2	2	2	1	1	0	1	1	0	2	2	2	2
		8	1	0	2	2	2	2	2	1	1	2	2	2	2	2	2	2	2
		9	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
		10	1	2	2	2	-1	1	2	-1	-1	1	2	2	2	2	2	2	2
		11	1	0	1	-1	0	0	1	1	0	1	2	1	-2	1	2	2	2
		12	0	0	1	2	1	1	1	2	2	0	2	2	2	2	2	1	2
		13	0	1	1	2	1	2	2	1	1	1	2	2	2	2	2	2	2
		14	0	-1	1	1	1	0	-1	1	1	1	1	0	0	1	1	1	1
		15	2	1	2	2	1	1	2	1	1	0	2	2	1	2	2	2	2
		16	2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	17	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
	18	2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
	19	0	0	1	1	2	1	2	2	0	2	2	2	2	2	2	2	2	
	20	1	2	0	1	0	0	-1	-2	-1	0	2	2	1	2	2	2	2	
	21	1	1	1	1	2	2	1	1	1	1	2	2	2	2	2	2	2	
	22	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
	23	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
	24	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
	25	-2	-2	-1	2	0	1	1	0	-1	0	2	2	0	2	2	2	2	
	26	2	2	2	1	2	1	1	2	1	1	2	2	2	2	2	2	2	
	27	1	1	1	1	1	0	0	1	0	0	0	1	1	1	0	0	0	
	28	0	2	1	2	2	2	1	1	1	-1	1	1	2	0	2	2	2	
		ARCS										Imersão			Social				

<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P2.4.1 – Modelo de avaliação revisado; P4.3 - Execução do jogo;	
	<b>Produto de saída:</b> P5.1.1 – Resultado da avaliação; P5.1.2 – Planilha para coleta de dados do modelo MEEGA.	
	Data da avaliação	Data em que a avaliação foi realizada.
	Local	Local em que a avaliação foi realizada, tipicamente sendo sala de aula, laboratório de informática ou extraclasse.
	Quantidade de alunos	Quantidade de alunos que realizaram a avaliação.
Dados coletados	Dados coletados, sendo apresentado como a figura 41.	

<b>Atividade</b>	<b>A5.2 Analisar dados da avaliação do jogo</b>
<b>Descrição</b>	Esta atividade tem por objetivo realizar a análise da avaliação do jogo. Na análise da avaliação utiliza-se os dados coletados na atividade <b>A5.1 Conduzir avaliação do jogo</b> .
<b>Tarefa</b>	<p><b>5.2 Analisar dados da avaliação do jogo</b></p> <p>Nesta tarefa realiza-se a análise da avaliação do jogo. Para realizar a análise utiliza-se as informações coletadas e apresentadas no produto de trabalho <b>P5.1.1 – Resultado da avaliação</b> e <b>P5.1.2 – Planilha para coleta de dados do modelo MEEGA</b>.</p> <p>A planilha de coleta de dados do MEEGA auxiliar na análise dos dados da avaliação, por permitir a visualização dos dados agrupados, como apresentado no gráfico da figura 42.</p>

### Motivação - Jogo 1



A análise da avaliação do jogo, no contexto do processo ENgAGED é realizada de duas formas. Na primeira deve-se inserir no produto de saída **P5.2 – Avaliação do jogo analisados** os gráficos de

motivação, experiência do usuário e aprendizagem. Na segunda, deve-se identificar os pontos fortes ou fracos para cada dimensão dos gráficos.

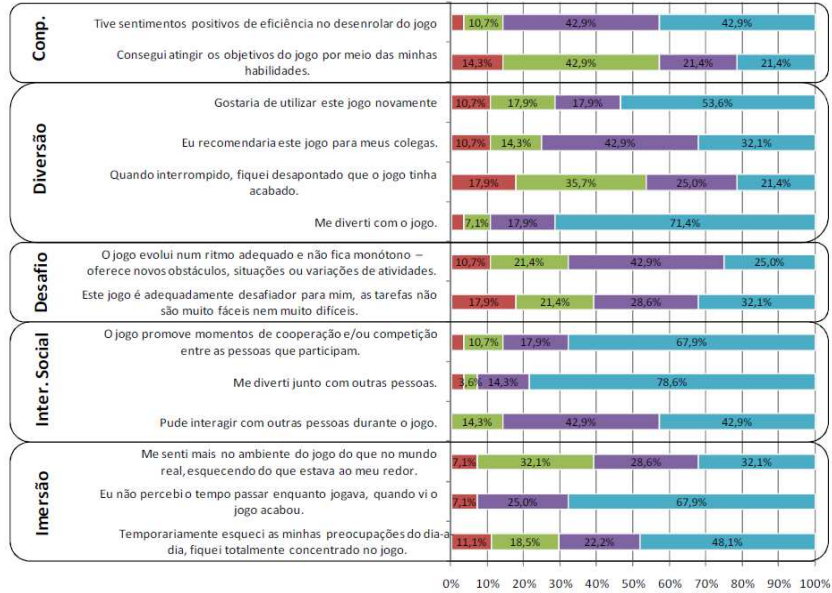
O gráfico Motivação apresenta as seguintes dimensões: atenção, confiança, relevância e satisfação.

O gráfico Experiência do Usuário apresenta as seguintes dimensões: Desafio, Competência, Interação Social e Imersão.

O gráfico de Aprendizagem apresenta as seguintes dimensões: Aprendizagem de curto prazo e Aprendizagem de longo prazo.

A figura 43 apresenta um exemplo do gráfico de Experiência do Usuário e a figura 44 apresenta o gráfico de Aprendizagem.

## Experiência do Usuário - Jogo 1



		Aprendizagem - Jogo 1																									
		<p>Gráfico de barras empilhadas mostrando a percepção dos alunos sobre o jogo em três áreas. O eixo horizontal representa a porcentagem de respostas (0% a 100%). O eixo vertical apresenta as perguntas. A legenda indica as pontuações: -2 (verde), -1 (vermelho), 0 (verde claro), 1 (roxo) e 2 (azul).</p> <table border="1"> <thead> <tr> <th>Pergunta</th> <th>-2</th> <th>-1</th> <th>0</th> <th>1</th> <th>2</th> </tr> </thead> <tbody> <tr> <td>Você acha que a experiência com o jogo vai contribuir para seu desempenho na vida profissional?</td> <td>0%</td> <td>10,7%</td> <td>17,9%</td> <td>53,6%</td> <td>17,9%</td> </tr> <tr> <td>Quanto eficiente o jogo foi para sua aprendizagem, comparando-o com outras atividades da disciplina?</td> <td>0%</td> <td>14,3%</td> <td>39,3%</td> <td>42,9%</td> <td>42,9%</td> </tr> <tr> <td>Quanto você acha que o jogo contribuiu para sua aprendizagem na disciplina?</td> <td>0%</td> <td>10,7%</td> <td>42,9%</td> <td>35,7%</td> <td>42,9%</td> </tr> </tbody> </table>		Pergunta	-2	-1	0	1	2	Você acha que a experiência com o jogo vai contribuir para seu desempenho na vida profissional?	0%	10,7%	17,9%	53,6%	17,9%	Quanto eficiente o jogo foi para sua aprendizagem, comparando-o com outras atividades da disciplina?	0%	14,3%	39,3%	42,9%	42,9%	Quanto você acha que o jogo contribuiu para sua aprendizagem na disciplina?	0%	10,7%	42,9%	35,7%	42,9%
Pergunta	-2	-1	0	1	2																						
Você acha que a experiência com o jogo vai contribuir para seu desempenho na vida profissional?	0%	10,7%	17,9%	53,6%	17,9%																						
Quanto eficiente o jogo foi para sua aprendizagem, comparando-o com outras atividades da disciplina?	0%	14,3%	39,3%	42,9%	42,9%																						
Quanto você acha que o jogo contribuiu para sua aprendizagem na disciplina?	0%	10,7%	42,9%	35,7%	42,9%																						
		<p>Após análise da motivação, experiência do usuário e aprendizagem, realiza-se a análise das perguntas que avaliam o conhecimento dos alunos sobre os conteúdos do jogo.</p> <p>Esta tarefa finaliza com a identificação dos pontos fortes e fracos definidos pelos alunos que avaliaram o jogo.</p>																									
<b>Produtos de trabalho</b>	<b>Produto de entrada:</b> P5.1.1 – Resultado da avaliação; P5.1.2 – Planilha para coleta de dados do modelo MEEGA.																										
	<b>Produto de saída:</b> P5.2 – Dados da avaliação analisados.																										
	Análise da Motivação																										
	Motivação	Gráfico apresentando os dados da motivação, como apresentado na figura 42.																									
	Atenção	Análise da dimensão de <b>atenção</b> da avaliação da motivação.																									

Confiança	Análise da dimensão de <b>confiança</b> da avaliação da motivação.
Relevância	Análise da dimensão de <b>relevância</b> da avaliação da motivação.
Satisfação	Análise da dimensão de <b>satisfação</b> da avaliação da motivação.
Análise da Experiência do Usuário	
Experiência do usuário	Gráfico apresentando os dados da experiência do usuário, como apresentado na figura 43.
Desafio	Análise da dimensão de <b>desafio</b> da avaliação da experiência do usuário.
Diversão	Análise da dimensão de <b>diversão</b> da avaliação da experiência do usuário.
Competência	Análise da dimensão de <b>competência</b> da avaliação da experiência do usuário.
Interação Social	Análise da dimensão de <b>interação social</b> da avaliação da experiência do usuário.
Imersão	Análise da dimensão de <b>imersão</b> da avaliação da experiência do usuário.
Análise da Aprendizagem	
Aprendizagem	Gráfico apresentando os dados da aprendizagem, como apresentado na figura 44.
Contribuição profissional	Análise da questão de contribuição do jogo para o desenvolvimento profissional;
Eficiência do jogo	Análise da questão de eficiência do jogo no aprendizado em relação a outras atividades.
Contribuição no aprendizado na disciplina	Análise da questão de contribuição do jogo para o aprendizado na disciplina.

Avaliação das perguntas sobre o conhecimento dos conteúdos do jogo	
Avaliação do conteúdo	Gráfico apresentando os dados sobre o conhecimento dos conteúdos do jogo.
Análise	Análise sobre as perguntas que avaliam o conhecimento dos alunos sobre os conteúdos abordados no jogo.
Geral	
Comentários	Comentários sobre os pontos fortes e fracos feitos pelos alunos.



## **APÊNDICE F – Questionário para Avaliação do Processo ENgAGED.**

Você está sendo convidado(a) a participar da avaliação do processo ENgAGED. Ele é um processo para desenvolvimento de jogos educacionais para ensino em computação, composto pelas fases de Análise, Projeto, Desenvolvimento, Execução e Avaliação. O objetivo do estudo é avaliar a qualidade do processo ENgAGED, do ponto de vista do criador de jogos educacionais.

O estudo faz parte de um projeto de pesquisa do GQS – Grupo de Qualidade de Software do INE – Departamento de Informática e Estatística da UFSC – Universidade Federal de Santa Catarina, do doutorando Paulo Eduardo Battistella, sob a orientação da professora Dr. Christiane Gresse von Wangenheim.

A sua participação nessa avaliação consistirá em ler o processo ENgAGED, e responder o questionário a seguir com base na sua experiência em desenvolver jogos educacionais.

Todos os dados coletados são confidenciais, de modo a assegurar a sua privacidade. Todos os resultados que serão divulgados em congressos ou revistas científicas serão apresentados de modo a não identificar você.

A qualquer momento você pode desistir da sua participação e retirar o seu consentimento sem qualquer tipo de prejuízo.

Qualquer dúvida, por favor, entre em contato conosco: GQS – Grupo de Qualidade de Software, Paulo Battistella (e-mail: [paulo@incod.ufsc.br](mailto:paulo@incod.ufsc.br) ou telefone: 48-3721-4715).

Mais informações: [www.gqs.ufsc.br](http://www.gqs.ufsc.br)

A sua participação é muito importante para nós!

<b>Dados de identificação do pesquisador</b>	
<b>1) Qual o seu nome?</b>	Nome do pesquisador.
<b>2) Quais jogos já desenvolveu?</b>	Jogos desenvolvidos.
<b>3) Quais jogos educacionais já desenvolveu?</b>	Jogos educacionais desenvolvidos.
<b>4) Qual é o nível da sua formação acadêmica?</b>	<ul style="list-style-type: none"> <li>• Técnico (completo/incompleto).</li> <li>• Tecnólogo (completo/incompleto).</li> <li>• Graduação (completo/incompleto).</li> <li>• Especialização (completo/incompleto).</li> <li>• Mestrado (completo/incompleto).</li> <li>• Doutorado (completo/incompleto).</li> </ul>

	<ul style="list-style-type: none"> <li>• Pós-doutorado (completo/incompleto).</li> <li>• Outro</li> </ul>
5) Qual é a área da sua formação acadêmica?	<ul style="list-style-type: none"> <li>• Computação</li> <li>• Design gráfico</li> <li>• Design de jogos</li> <li>• Pedagogia</li> <li>• Outro</li> </ul>
<b>Feedback do Processo ENgAGED</b>	
6) Você identificou algo ambíguo no processo?	Não (___), pule para 8. Sim (___), explique sua resposta na questão 7.
7) Quais fases, atividades e produtos de trabalho você considerou ambíguos?	Fases:
	Atividades:
	Produtos de Trabalho:
8) Você identificou fases, atividades e produtos de trabalho inconsistentes?	Não (___), pule para 10. Sim (___), explique sua resposta na questão 9.
9) Quais fases, atividades e produtos de trabalho considerados inconsistentes?	Fases:
	Atividades:
	Produtos de Trabalho:
10) Existem fases, atividades e produtos de trabalho faltantes que tornam o processo incompleto?	Não (___), pule para 12. Sim (___), explique sua resposta na questão 11.
11) Quais fases, atividades e produtos de trabalho são incompletos?	Fases:
	Atividades:
	Produtos de Trabalho:
12) Você identifica fases, atividades e produtos de trabalho são incompreensíveis?	Não (___), pule para 14. Sim (___), explique sua resposta na questão 13.
13) Quais fases, atividades e produtos de trabalho são considerados incompreensíveis?	Fases:
	Atividades:
	Produtos de Trabalho:
14) Você identificou erros no processo?	Não (___), pule para 16. Sim (___), explique sua resposta na questão 15.
15) Quais atividades e produtos de trabalho você identificou erros?	Fase:
	Atividades:
	Produtos de Trabalho:
16) Você desenvolveu jogo digital com o processo ENgAGED?	Não (___) Sim (___), nome do jogo:
17) Você desenvolveu jogo não-digital com o processo ENgAGED?	Não (___), pule para 19. Sim (___), nome do jogo:
18) Caso tenha desenvolvido um jogo com ENgAGED, qual foi gênero?	<ul style="list-style-type: none"> <li>• Ação-aventura: Um jogo que exigem que os jogadores utilizem reflexos rápidos, com precisão para superar</li> </ul>

	<p>obstáculos, resolver desafios ou para responder perguntas.</p> <ul style="list-style-type: none"> <li>• Adivinhação: Um jogo em que o objetivo é identificar algum tipo de informação, como uma palavra, a partir de desenhos ou imitação de outros jogadores.</li> <li>• Corrida: Um jogo em que o jogador comanda um veículo ou participa de uma corrida, tentando se mover mais rápido do que os oponentes para alcançar um objetivo específico ou ser mais rápido que um determinado tempo.</li> <li>• Estratégia: Um jogo que é focado na jogabilidade e exige pensamentos táticos e cuidados, a fim de conseguir a vitória.</li> <li>• Puzzle: Um jogo que envolve personagens que controlam um conjunto de comandos, para navegar através de labirintos ou para organizar os objetos de maneira correta.</li> <li>• Quiz: Um jogo onde o jogador precisa responder perguntas para uma determinada área de conhecimento.</li> <li>• Role-playing game (RPG): Um jogo onde o jogador controla ações de um protagonista e com este personagem vive imerso em um mundo fictício. Em jogo de role-playing os personagens interagem com este mundo e ficam mais fortes.</li> <li>• Roll-and-move: Jogos de tabuleiro em que as fichas são movidas com base nos resultados mostrados em um ou mais dados (por exemplo: <i>Snake and Ladder</i> e <i>Monopoly</i>).</li> <li>• Simulação: Um jogo desenvolvido para colocar o jogador no controle de um determinado ambiente ou atividade, o qual busca ser o mais realista possível.</li> <li>• Outro</li> </ul>
<b>19) Quais fases, atividades e produtos</b>	Fases:

<b>de trabalho você modificou no desenvolvimento do seu jogo?</b>	Atividades:			
	Produtos de Trabalho:			
<b>Qualidade do processo ENgAGED</b>				
<b>20) Acho que consigo facilmente adaptar o processo para o desenvolvimento de um determinado tipo de jogo?</b>	Não (___) Sim (___), explique sua resposta:			
<b>21) Acho que consigo desenvolver um jogo mais rapidamente quando utilizo o processo ENgAGED?</b> <i>(Por favor, marque com um "X" a sua opção)</i>				
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente
<b>22) Acho que consigo melhorar o meu desempenho no desenvolvimento de jogos, utilizando o processo ENgAGED?</b> <i>(Por favor, marque com um "X" a sua opção)</i>				
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente
<b>23) Acho que consigo aumentar a minha produtividade no desenvolvimento do jogo, utilizando o processo ENgAGED?</b> <i>(Por favor, marque com um "X" a sua opção)</i>				
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente
<b>24) Acho que consigo aumentar a eficácia no desenvolvimento de jogos, utilizando o processo ENgAGED?</b> <i>(Por favor, marque com um "X" a sua opção)</i>				
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente
<b>25) Acho que consigo tornar mais fácil o desenvolvimento de jogos, utilizando o processo ENgAGED?</b> <i>(Por favor, marque com um "X" a sua opção)</i>				
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente
<b>26) Acho que o processo ENgAGED é realmente útil para o desenvolvimento de jogos?</b> <i>(Por favor, marque com um "X" a sua opção)</i>				
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente
<b>27) Acho que é fácil aprender a usar o processo ENgAGED?</b> <i>(Por favor, marque com um "X" a sua opção)</i>				

	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente	
<b>28) Acho que é fácil obter informações do processo ENgAGED para fazer o que desejo?</b>					
<i>(Por favor, marque com um "X" a sua opção)</i>					
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente	
<b>29) Acho que consigo interagir com o processo ENgAGED de modo claro e compreensível?</b>					
<i>(Por favor, marque com um "X" a sua opção)</i>					
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente	
<b>30) Acho que o processo ENgAGED é flexível?</b>					
<i>(Por favor, marque com um "X" a sua opção)</i>					
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente	
<b>31) Acho que poderia me tornar habilidoso usando o processo ENgAGED?</b>					
<i>(Por favor, marque com um "X" a sua opção)</i>					
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente	
<b>32) Acho que o processo ENgAGED é realmente fácil de usar para o desenvolvimento de jogos?</b>					
<i>(Por favor, marque com um "X" a sua opção)</i>					
	Concordo Fortemente	Concordo	Discordo	Discordo Fortemente	

- O que mais gostou do processo ENgAGED:

---



---



---



---

- O que você acha de deve melhorar:

---



---



---



---

- Mais algum comentário:

---

---

---

Obrigado por sua participação neste estudo!

Qualquer dúvida, por favor, entre em contato conosco:

GQS – Grupo de Qualidade de Software, Paulo Battistella (e-mail:  
[paulo@incod.ufsc.br](mailto:paulo@incod.ufsc.br) ou telefone: 48-3721-4715).

Mais informações: [www.gqs.ufsc.br](http://www.gqs.ufsc.br)

## APÊNDICE G – Questionário para Avaliação do Jogo Educacional.

Nome do jogo:

Gostaríamos que você respondesse as questões abaixo para nos ajudar a melhorar este jogo. Todos os dados são coletados anonimamente e somente serão utilizados no contexto desta pesquisa. Algumas fotografias poderão ser feitas como registro desta atividade, mas não serão publicadas em nenhum local sem autorização.

Disciplina:

Por favor, circule um número de acordo com o quanto você concorda ou discorda de cada afirmação abaixo.

Afirmações	Sua avaliação			Comentários sobre a questão
	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O design do jogo é atraente (interface ou objetos, como cartas ou tabuleiros).	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Houve algo interessante no início do jogo que capturou minha atenção.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
A variação (de forma, conteúdo ou de atividades) ajudou a me manter atento ao jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O conteúdo do jogo é relevante para os meus interesses.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O funcionamento deste jogo está adequado ao meu jeito de aprender.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
O conteúdo do jogo está conectado com outros conhecimentos que eu já possuía.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Foi fácil entender o jogo e começar a utilizá-lo como material de estudo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Ao passar pelas etapas do jogo senti confiança de que estava aprendendo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Estou satisfeito porque sei que terei oportunidades de utilizar na prática coisas que aprendi com o jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
É por causa do meu esforço pessoal que consigo avançar no jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Temporariamente esqueci das minhas preocupações do dia-a-dia, fiquei totalmente concentrado no jogo.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Eu não percebi o tempo passar enquanto jogava, quando vi o jogo acabou.	Discordo Fortemente	-2-1 0 +1 +2	Concordo Fortemente	
Me senti mais no ambiente do jogo do que no mundo real, esquecendo do	Discordo Fortemente	-2-1 0	Concordo Fortemente	

que estava ao meu redor.		+1 +2		
Pude interagir com outras pessoas durante o jogo	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Me diverti junto com outras pessoas	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
O jogo promove momentos de cooperação e/ou competição entre as pessoas que participam.	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Este jogo é adequadamente desafiador para mim, as tarefas não são muito fáceis nem muito difíceis.	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
O jogo evolui num ritmo adequado e não fica monótono – oferece novos obstáculos, situações ou variações de atividades.	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Me diverti com o jogo.	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Quando interrompido, fiquei desapontado que o jogo tinha acabado (gostaria de jogar mais).	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Eu recomendaria este jogo para meus colegas.	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Gostaria de utilizar este jogo novamente	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Consegui atingir os objetivos do jogo por meio das minhas habilidades.	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	
Tive sentimentos positivos de eficiência no desenrolar do jogo	Discordo Fortemente	-2 -1 0 +1 +2	Concordo Fortemente	

– O jogo contribuiu para a minha aprendizagem na disciplina  
(Discordo fortemente)-2 -1 0 +1 +2(Concordo fortemente)

– O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina.  
(Discordo fortemente)-2 -1 0 +1 +2(Concordo fortemente)

– A experiência com o jogo vai contribuir para meu desempenho na vida profissional.  
(Discordo fortemente)-2 -1 0 +1 +2(Concordo fortemente)

– Atribua uma nota de 1,0 a 5,0 para seu nível de conhecimento antes e depois do jogo aos conceitos listados na tabela abaixo (1,0 – pouco;5,0 – muito).

Conceitos	Lembrar o que é		Compreender como funciona		Aplicar na prática	
	Antes	Depois	Antes	Depois	Antes	Depois
Conhecimento						



sobre o conteúdo do jogo						
-----------------------------	--	--	--	--	--	--

– Cite 3 pontos fortes do  
jogo: \_\_\_\_\_

---

---

---

– Por favor, dê 3 sugestões para a melhoria do  
jogo: \_\_\_\_\_

---

---

---

Muito obrigado pela sua contribuição!

## **APÊNDICE H – TCLE - Termo de Consentimento Livre e Esclarecido.**

Você está sendo convidado(a) a participar da avaliação do processo ENgAGED. Ele é um processo para desenvolvimento de jogos educacionais para ensino em computação, composto pelas fases de Análise, Projeto, Desenvolvimento, Execução e Avaliação. O objetivo do estudo é avaliar a qualidade do processo ENgAGED, do ponto de vista do criador de jogos educacionais.

O estudo faz parte de um projeto de pesquisa do GQS – Grupo de Qualidade de Software do INE – Departamento de Informática e Estatística da UFSC – Universidade Federal de Santa Catarina, do doutorando Paulo Eduardo Battistella, sob a orientação da professora Dr. Christiane Gresse von Wangenheim.

A sua participação nessa avaliação consistirá em ler o processo ENgAGED ,e responder o questionário a seguir com base na sua experiência em desenvolver jogos educacionais.

Todos os dados coletados são confidenciais, de modo a assegurar a sua privacidade. Todos os resultados que serão divulgados em congressos ou revistas científicas serão apresentados de modo a não identificar você.

A qualquer momento você pode desistir da sua participação e retirar o seu consentimento sem qualquer tipo de prejuízo.

A participação nesta pesquisa não traz complicações, talvez, apenas, um pequeno sentimento de timidez que algumas pessoas podem sentir durante a observação e/ou no momento da reflexão. Com o objetivo de minimizar qualquer risco serão apresentados claramente o objetivo e execução da pesquisa seguindo a ética em pesquisa como também as instruções referente ao jogo educacional.

A participação é gratuita e voluntária. Não será disponível nenhuma compensação financeira adicional. Não haverá qualquer procedimento que possa incorrer em danos físicos ou financeiros aos participantes e, portanto, não haveria necessidade de indenização por parte dos pesquisadores.

O participante tem como benefício na participação desta avaliação, a aplicação do seu conhecimento em desenvolvimento de jogos, com objetivo de ampliar a qualidade do processo ENgAGED.

Caso você aceite participar da pesquisa, o TCLE precisa ser assinado por você e pelo pesquisador responsável em duas vias, sendo que uma das vias ficará com você e a outra será arquivada pelos pesquisadores.

A pesquisadora responsável explicitamente declara o Termo de Consentimento Livre e Esclarecido está em conformidade com as exigências contidas no item IV.3 da Resolução 466/12. Em caso de dúvidas ou notificação de acontecimentos não previstos entrar em contato com CEPESH - Comitê de Ética em Pesquisa com Seres Humanos da Universidade Federal de Santa Catarina, Campus Universitário - Trindade - Florianópolis/SC, CEP 88040-900 Caixa Postal 476 no telefone (48) 3721-4715, pelo qual o projeto de pesquisa foi aprovado.

---

Pesquisadora responsável

Prof<sup>ª</sup>. Dr. rer. nat. Christiane A. Gresse von Wangenheim, PMP  
GQS-Grupo de Qualidade de Software/INE – Departamento de Informática e Estatística/UFSC - Universidade Federal de Santa Catarina, Campus Universitário - Trindade - Florianópolis/SC, CEP 88040-900 no telefone (48) 3721-4715 ou via email: gresse@inf.ufsc.br

#### CONSENTIMENTO DO PARTICIPANTE

Eu, \_\_\_\_\_  
\_\_\_\_\_, RG \_\_\_\_\_, abaixo assinado, concordo a participar do estudo: “Avaliação da qualidade do processo ENgAGED”.

Fui devidamente informado e esclarecido sobre a pesquisa, os procedimentos nela envolvidos, assim como os possíveis riscos e benefícios decorrentes da minha participação. Foi-me garantido que posso retirar meu consentimento a qualquer momento, sem que isto leve a qualquer penalidade.

Florianópolis, \_\_\_ de \_\_\_\_\_ de 2015.

Assinatura do participante:

---

## APÊNDICE I – APLICAÇÕES DO PROCESSO ENGAGED DETALHADOS

A seguir são apresentados os 4 estudos de caso nos quais foram desenvolvidos jogos educacionais utilizando o processo ENgAGED. Os jogos desenvolvidos são *SCRUM-Scape* (CAMARGO, 2013), *SCRUM'ed* (SCHNEIDER, 2015), *PM Quiz* (CASSETTARI, 2015), e o *Fuga de Gambix* (JUNCKES, 2016).

### JOGO SCRUM-SCAPE

O primeiro jogo foi aplicado no segundo semestre de 2013, no qual o jogo SCRUM-Scape (CAMARGO, 2013) foi desenvolvido utilizando uma primeira versão do processo ENgAGED. Nesta etapa da pesquisa o processo apresentava apenas as fases e atividades descritas em alto nível.

#### F1 – Análise da Unidade Instrucional

Na fase de identificação das metas instrucionais, foi definido como meta principal, ensinar SCRUM para os alunos da disciplina de gerência de projetos do curso de graduação em Ciências da Computação. Foi definido como contexto da unidade instrucional que seria aplicada aos alunos da sexta fase do curso de graduação de Ciência da Computação, na disciplina de Gerência de Projetos. Considerando que a disciplina possui 72 horas/aula, foi criada uma aula de 1 hora e 40 minutos para aplicar o jogo em sala de aula.

Os objetivos de desempenho da unidade instrucional são:

Lembrar os nomes dos principais artefatos, cerimônias e papéis do SCRUM.

Entender o objetivo dos artefatos e suas relações com as cerimônias.

Lembrar as responsabilidades dos papéis do SCRUM Master, *Development Team* e *Product Owner* e as suas relações com os artefatos.

Entender os objetivos das cerimônias durante um projeto.

## **F2 – Projeto da Unidade Instrucional**

Como estratégia instrucional foi aplicada em sala de aula o jogo SCRUM-Scape, tendo por objetivo avaliar o conhecimento dos alunos nos seguintes conteúdos:

Responsabilidades de Papéis: *SCRUM Master, Development Team e Product Owner.*

Artefatos: *Product Backlog, Sprint Backlog e Taskboard.*

Cerimônias: *Daily SCRUM, Sprint Retrospective, Sprint Review e Sprint Planning.*

O jogo foi apresentado aos alunos após 4 aulas teóricas e um exercício prático sobre SCRUM.

## **F3 – Desenvolvimento do jogo educacional**

### ***F3.1 – Análise do jogo***

Os requisitos do jogo são: a aplicação do jogo não pode ultrapassar o tempo de um encontro em sala de aula, o qual possui aproximadamente 2 horas; ter como gênero RPG; ter como plataforma computador pessoal e sistema operacional Windows; ser jogado de modo individual; a medida que o jogador avança precisa avançar no conteúdo.

Como principal requisito do jogo, foi definido que o personagem principal precisa solucionar diversos desafios relativos a metodologia ágil SCRUM e avançar nos cenários do jogo até responder as perguntas sobre responsabilidades de papéis, artefatos e cerimônias. Para definir as ideias do jogo, foi definido que a estória do jogo se passaria em uma prisão, na qual o personagem principal teria que responder perguntas para avançar no jogo e escapar da prisão. Na medida em que o personagem principal avança, encontra monstros e conta com ajuda de personagens aliados que encontra ao longo do jogo para vencer os monstros.

### ***F3.2 – Concepção do jogo***

Na fase de concepção foram criadas narrativas de acordo com o perfil dos alunos, ou público-alvo, definido na fase de análise. Em

relação ao layout do jogo foi definido a utilização de um cenário baseado em uma prisão formada por 3 blocos. Tendo como desafio do jogador passar pelos 3 blocos para vencer o jogo. No primeiro bloco liberta os personagens aliados das celas da prisão e nos blocos subsequentes luta contra monstros.

O jogo foi concebido em 3 níveis, sendo abordado em cada nível um conteúdo do SCRUM. No primeiro bloco são abordados os papéis, no segundo as cerimônias e no último bloco os artefatos. Na definição dos desafios do jogo, foram incluídas questões do SCRUM.

### ***F3.3 – Design do jogo***

As *storyboards* foram criadas a partir da ferramenta de desenvolvimento de jogos no estilo RPG chamado de *RPG Maker*<sup>7</sup>. Neste contexto as animações e áudio são definidas pela própria ferramenta. A seleção desta *engine* de jogo foi definida com base em reuniões e principalmente com base na experiência do professor.

### ***F3.4 – Implementação do jogo***

A implementação do jogo continuou sendo realizada na *game engine* *RPG Maker*, não sendo necessária a codificação das telas, uma vez que a *engine* permite a compilação do jogo em um arquivo executável. A partir do *RPG Maker* o professor escolhe os mapas, personagens, cenários e todas as animações do jogo. A forma de criação dos cenários é intuitiva, o professor seleciona e move os itens para cada cenário que o jogo terá.

---

<sup>7</sup>RPGMaker: [www.rpgmakerweb.com](http://www.rpgmakerweb.com)



### ***F3.5 – Teste do jogo***

A medida que os cenários foram construídos, o professor realizava testes para verificar a funcionalidade do jogo e também identificar possíveis erros na sequência lógica dos desafios, erros de português nas mensagens trocadas entre os personagens do jogo. Estes testes foram realizados pelo próprio professor do jogo.

Como a ferramenta para implementação do jogo não permite a inserção de scripts ou código, a existência de erros de compilação é baixa.

### **F4 – Execução da Unidade Instrucional**

A preparação para execução do jogo foi realizada dias antes do jogo ser aplicado em sala de aula, sendo necessário apenas salvar o arquivo executável do jogo nos computadores do laboratório onde seria jogado pelos alunos.

O jogo foi aplicado em uma disciplina de Gerência de Projetos na Universidade Federal de Santa Catarina (UFSC) durante uma aula e com um grupo de profissionais de TI sendo realizado à distância. A execução

ocorreu em novembro de 2013 envolvendo um total de 17 participantes. Durante a execução, foi primeiramente explicada as regras do jogo e em seguida os participantes jogaram o jogo individualmente.



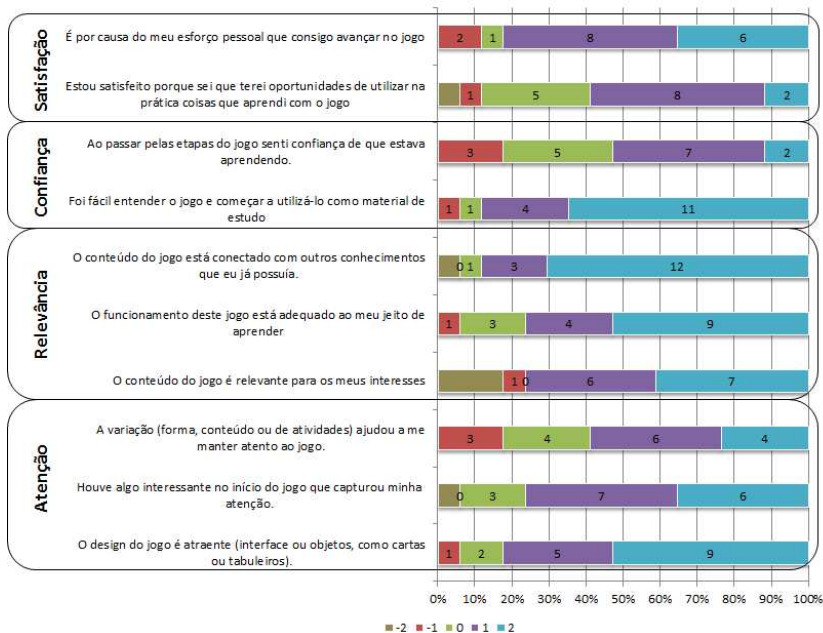
### **F5 – Avaliação da Unidade Instrucional**

Seguindo o modelo MEEGA (Savi, Wangenheim e Borgatto, 2011) foi definida a avaliação em termos de motivação dos alunos, experiência dos usuários e aprendizagem. O planejamento da avaliação foi realizado em paralelo com o planejamento da execução do jogo. A coleta de dados ocorreu ao final da execução por meio de um questionário online. Os dados foram analisados conforme definido no MEEGA (Savi, Wangenheim e Borgatto, 2011) e as sugestões de melhoria foram utilizadas para melhorar o SCRUM-Scape.

A partir dos dados coletados foram gerados gráficos que representam a frequência das respostas considerando a amplitude dos valores (-2, -1, 0, 1, 2) para cada uma das perguntas do questionário. Sendo que, o valor '-2' representa total discordância e o valor '2' representa total concordância com a questão. Além disso, as perguntas foram agrupadas em dimensões dentro de dois componentes analisados: motivação e experiência de usuário.

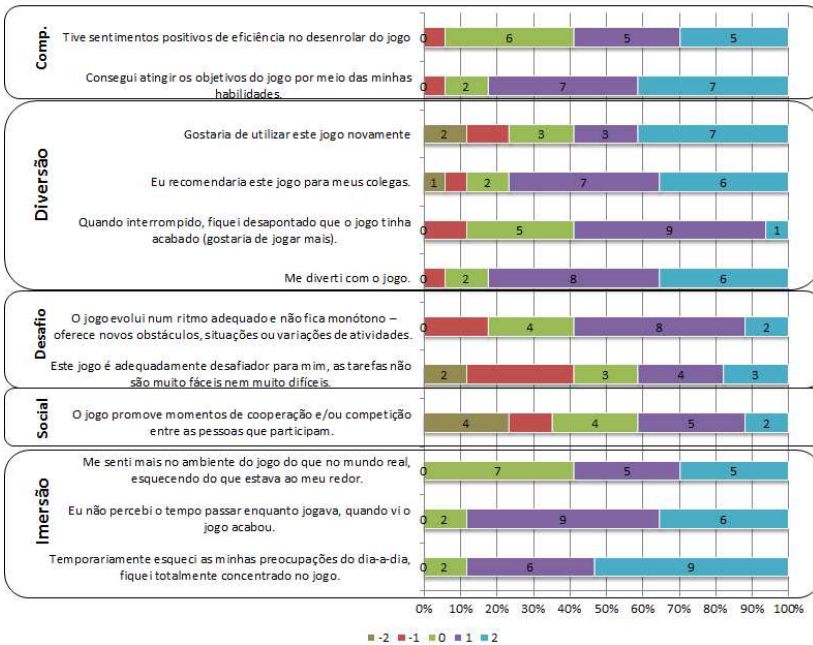


## Avaliação da Motivação.



Os resultados obtidos apresentam valores positivos em todas as dimensões avaliadas acerca da motivação dos alunos, ou seja, boa parte das respostas está situada nos valores de total concordância. As dimensões com maior destaque são a 'relevância' e 'atenção' que apresentaram em mais de 60% das respostas os valores 1 e 2. Outro componente avaliado foi a experiência de usuário.

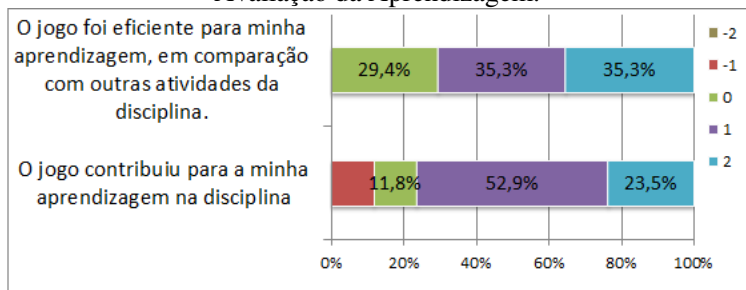
## Avaliação da Experiência do Usuário.



A avaliação acerca da experiência de usuário também demonstrou resultados positivos em todas as dimensões. As dimensões com maior destaque foram ‘diversão’ e ‘imersão’. Tal fato pode ser atribuído à boa evolução proporcionada pela narrativa bem como a mudança nos cenários ao longo do jogo, o que permitiu aos jogadores a interação com novos cenários e personagens.

Adicionalmente foram feitas questões sobre o aprendizado e o nível de conhecimento adquirido pelos alunos. Para esses dados também foram gerados gráficos de frequência.

### Avaliação da Aprendizagem.



## JOGO SCRUM'ED

O segundo jogo foi aplicado no primeiro semestre de 2015, no qual o jogo SCRUM'ed (SCHNEIDER, 2015) foi desenvolvido utilizando a versão completa do processo ENgAGED, contendo descrições detalhadas das fases e atividades.

Uma versão detalhada deste processo de desenvolvimento está apresentada no apêndice E, e também no relatório técnico *Aplicação do Processo ENgAGED: Desenvolvimento do Jogo Educacional SCRUM'ed* descrito por Battistella, Schneider e Wangenheim (2015).

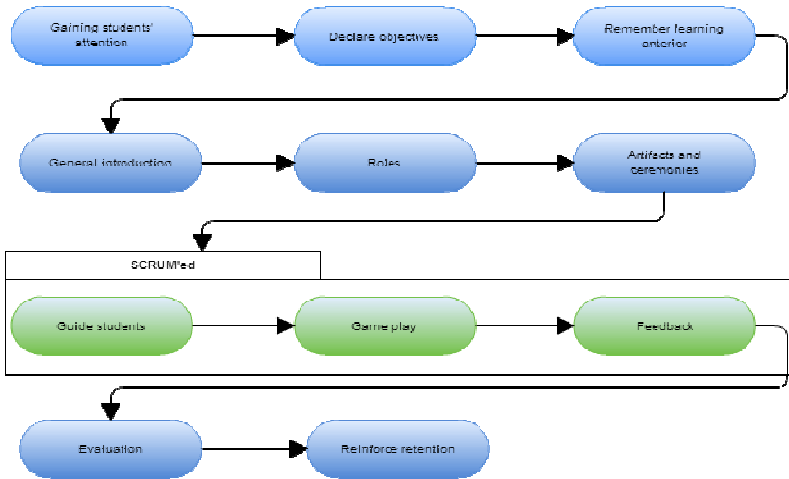
### F1 – Análise da Unidade Instrucional

A primeira fase do processo ENgAGED tem por objetivo realizar a análise da unidade instrucional. Na análise levante-se o público-alvo, ambiente onde o jogo é aplicado e os objetivos de desempenho.

O contexto do jogo é o curso de graduação em Ciências da Computação, da Universidade Federal de Santa Catarina (UFSC), em Florianópolis Brasil. O jogo foi desenvolvido para aplicação na disciplina de Planejamento e Gestão de Projetos de Software, tipicamente realizado pelos alunos do 6º ano do curso de computação. Como pré-requisito da disciplina, espera-se que os alunos tenham cursado disciplinas introdutórias de Engenharia de Software.

Para aplicação do jogo, foi estimado um total de 25 alunos, entre os 16 e 30 anos. A duração da aplicação do jogo é de 2 horas/aula, sendo desenvolvido para aplicação após aulas expositivas de SCRUM.

### Contexto de aplicação do jogo.



Ao final da aplicação do jogo, espera-se que os alunos sejam capazes de:

**Comprender os Papeis:** *SCRUM Master*.

**Comprender os Artefatos:** *Sprint Planning, Daily Scrum, Sprint Review*.

**Comprender as Cerimônias:** *Product Backlog, Sprint Backlog, Task-board*.

## F2 – Projeto da Unidade Instrucional

Nesta fase de design da unidade instrucional o conteúdo deve ser apresentado e organizado para ser utilizado no jogo.

O conteúdo abordado no jogo é o SCRUM, que é uma metodologia ágil de desenvolvimento de software. Esta metodologia visa simplificar o desenvolvimento de softwares com base na transparência de informação entre os membros da equipe, inspeção frequente do que está sendo desenvolvido e ser adaptável para mudanças que ocorrem ao longo do projeto (Larman, 2003), (Schwaber e Beedle, 2001), (Schwaber e Sutherland, 2013). O SCRUM apresenta papeis, artefatos e cerimônias, os quais são apresentados a seguir.

Os papéis que tipicamente fazem parte do SCRUM são (Schwaber e Sutherland, 2013):

**Product Owner:** É quem define os requisitos e as prioridades, representando o cliente. Deve ser uma única pessoa e sua responsabilidade é o gerenciamento do *Product Backlog* durante todo o processo - desde a especificação dos requisitos até a inserção de mudanças durante o projeto.

**SCRUM Master:** Auxilia o SCRUM Team a transformar o *Product Backlog* em funcionalidades. É de sua responsabilidade ajudar na organização da equipe e remover impedimentos ao progresso, e atuar como facilitador durante as cerimônias.

**SCRUM Team:** São todos os demais envolvidos no desenvolvimento do projeto (programadores, designers, testadores), formando uma equipe dinâmica capaz de gerenciar e organizar suas próprias atividades. Deve ser uma equipe multifuncional com todas as habilidades necessárias para o sucesso do projeto sem a intervenção de pessoas de fora da equipe.

Os artefatos que integram o SCRUM são (Schwaber e Sutherland, 2013):

**Product Backlog:** É a lista completa e priorizada de tudo o que será necessário no produto final, sendo a única fonte de requisitos disponível para consulta. É um artefato dinâmico, sempre sob atualização e deve ser composto por funcionalidades, requisitos, funções, melhorias. Cada item geralmente contém uma descrição, uma prioridade e uma estimativa de tamanho ou complexidade. A ordenação pode ser por risco, prioridade ou necessidade.

**Sprint Backlog:** É a lista com os itens selecionados do *Product Backlog* para o Sprint atual, junto com o plano para a entrega da versão incremental do produto e o sucesso do *Sprint Goal*. Define, portanto, o trabalho que a equipe de desenvolvimento executará durante o *Sprint* em curso e decompõe esse trabalho em atividades. Os itens devem estar refinados por prioridade e estimativa de esforço.

**Burndown Chart:** É um gráfico que mostra o progresso diário dos trabalhos dentro de um *Sprint* e permite detectar possíveis desvios. Ao final de cada dia o gráfico deve ser atualizado (manual ou automaticamente) e a equipe deve monitorar o andamento do projeto a cada iteração Cohn (2010).

**Taskboard:** É um painel utilizado como ferramenta no auxílio do acompanhamento do Sprint, servindo principalmente para acompanhar o

andamento das atividades. É atualizado durante o *Daily SCRUM* e deve estar sempre acessível a todos os envolvidos no projeto, pois agrega mais transparência e visibilidade ao processo de desenvolvimento.

As cerimônias do SCRUM são (Schwaber e Sutherland, 2013):

**Sprint Planning Meeting:** Tem como objetivo definir, colaborativamente, os objetivos para o próximo Sprint e deve contar com a participação de toda a equipe. É dividida em duas etapas, cada uma durando aproximadamente metade da reunião: Na primeira é definida a composição da entrega incremental do próximo Sprint, e na segunda é estimada a quantidade de trabalho necessária para se atingir tal objetivo. Utiliza o Product Backlog para escolher tarefas a serem transferidas para o Sprint Backlog.

**Daily SCRUM:** Tem como objetivo sincronizar as atividades da equipe e planejar as atividades para as próximas 24 horas. É uma cerimônia diária que deve ter até quinze minutos: Deve-se monitorar o trabalho desenvolvido desde a última reunião, sendo que cada membro da equipe de desenvolvimento deve responder três perguntas: o que foi produzido, o que será feito após a reunião e quais obstáculos estão sendo enfrentados. Nesta cerimônia a equipe deve medir o progresso do Sprint Goal, e o artefato Taskboard é consultado e atualizado.

**Sprint Review Meeting:** Tem como objetivo apresentar os resultados produzidos ao *Product Owner*, monitorar o processo e caso necessário realizar alterações. Ocorre ao final do *Sprint* e o tempo gasto na reunião deve ser proporcional à duração do *Sprint*, por exemplo, 2 horas para um *Sprint* de 2 semanas. Nessa reunião o Product Owner aprova ou rejeita os resultados apresentados.

**Sprint Retrospective:** Tem como objetivo auto-avaliar e planejar melhorias para a equipe. É uma reunião de até 3 horas, onde são inspecionados os principais aspectos do *Sprint* que foi encerrado em relação às pessoas, processos e ferramentas. Devem ser identificados os principais aspectos de sucesso e sugeridas melhorias, que serão incluídas em um plano para execução no próximo *Sprint*.

### F3 – Desenvolvimento do jogo educacional

A fase de desenvolvimento é composta pelas seguintes fases: análise do jogo, concepção do jogo, design do jogo, implementação do jogo e teste do jogo.

### ***F3.1 – Análise do jogo***

A fase de análise do jogo tem por objetivo realizar o levantamento dos requisitos funcionais e não funcionais do jogo. Os requisitos funcionais definem explicitamente as funcionalidades do jogo e os requisitos não funcionais definem propriedades e restrições do jogo (Swebok, 2004).

Entre os principais requisitos funcionais levantados no desenvolvimento do jogo SCRUM'ed, destaca-se: (a) conter ao longo do jogo os conceitos básicos dos papéis, artefatos e cenários do SCRUM; (b) avaliar o jogador em relação ao domínio do conteúdo do jogo; (c) o jogador deve ser penalizado quando realizar ações erradas durante o jogo.

Entre os principais requisitos não funcionais levantados no desenvolvimento do jogo, destaca-se: (a) o jogo deve ter duração de até 100 minutos; (b) deve ser um jogo digital e em 3D; (c) o jogo deve rodar no sistema operacional Windows 7 ou versões superiores; (d) O jogo não deve apresentar conteúdo difamatório, obsceno, ofensivo, odioso ou violento.

### ***F3.2 – Fase: Concepção do jogo***

A fase de concepção do jogo tem por objetivo modelar o jogo identificando, por exemplo, os objetivos do jogo, regras, narrativas, elementos do jogo, *feedback* educacional.

O objetivo do jogo SCRUM'ed é liderar uma equipe SCRUM para resolver desafios em um cenário medieval. O gênero do jogo é *Role-playing-game* (RPG), sendo desenvolvido para computadores *stand-alone* e modo de interação *single player*.

Entre as principais regras do jogo, destaca-se: *o jogador principal (avatar) deve interagir com os personagens e executar as atividades solicitadas pelo cliente. Para cada atividade resolvida com sucesso e interações bem sucedidas com os personagens do jogo, o jogador ganha cenouras.*

O jogo apresenta a seguinte narrativa: *o personagem principal comanda a equipe SCRUM em um cenário medieval. Desde o primeiro nível o jogador deve interagir com os personagens para conhecer os membros da equipe. Em seguida o cliente entra no cenário do castelo e apresenta as atividades que a equipe deve executar. Após uma reunião, a equipe parte para o jardim do palácio para realizar as atividades*

*solicitadas pelo cliente. No final do dia, os membros da equipe retornam ao castelo e avaliam o resultado do trabalho.*

O jogo apresenta como elementos do jogo, personagens, cenários e artefatos.

#### Personagens do jogo.

Ator	Papel	Tipo	Função
Jogador	SCRUM <i>Master</i>	<i>Avatar</i>	Resolve problemas para os Membros da Equipe; mostra o planejamento; modera Daily Meetings
Guru SCRUM		No-play character (NPC)	Fornecer dicas do jogo
Sir John	Cliente	No-play character (NPC)	Fornecer as tarefas ao grupo
Tristan	<i>Product Owner</i>	No-play character (NPC)	Aceita/Nega as tarefas do cliente e avalia os resultados
Homer	Membro da Equipe	No-play character (NPC)	Realiza as tarefas para o cliente

O jogo apresenta artefatos, que são componentes do SCRUM nos quais os personagens devem interagir.

#### Artefatos do jogo.

Artefatos	Função
<i>Product Backlog</i>	Lista acessível na interface contendo todas as missões existentes, sua prioridade e estimativa.
<i>Sprint Backlog</i>	Lista acessível na interface contendo os desafios a serem realizados para que se possa concluir o Sprint atual, com sua prioridade, estimativa e personagem responsável.
<i>Taskboard</i>	Lista acessível na interface contendo os desafios a serem realizados para que se possa concluir o Sprint atual, organizados por estado ( <i>to do, in progress, done</i> ).

O jogo apresenta dois cenários principais, o primeiro é o castelo na qual a equipe SCRUM trabalha. O segundo é o jardim do palácio, na qual a equipe SCRUM executa as atividades solicitadas pelo cliente.

O jogo apresenta feedbacks educacionais simultâneos, deste modo, o jogo está sempre reforçando os conceitos de papel, artefato ou cerimônia do SCRUM.

### ***F3.3 – Fase: Design do jogo***




A fase de design do jogo tem por objetivo definir a linguagem de programação ou game engine, produzir os elementos do jogo e modelar o fluxograma dos níveis do jogo, *feedback* educacional e o diálogo do jogo.



O jogo SCRUM'ed foi desenvolvido utilizando game engine *Unity 3D*. Esta engine foi escolhida por apresentar diversos *assets* (elementos de jogo) disponíveis para baixar gratuitamente e também por apresentar uma ampla gama de comunidades e fóruns na internet para tirar dúvidas de desenvolvimento.

Nesta fase os personagens foram desenhados, como os exemplos apresentados na tabela a seguir. O jogador terá o papel do SCRUM Master e comandará a equipe. O Tristan que é o *Product Owner*, responsável pelo gerenciamento do *Product Backlog*. O Homer, que é um dos membros da equipe SCRUM.

Esboço dos personagens do jogo.

Ator	Papel	Tipo	Ilustração
Jogador	SCRUM Master	Avatar	
Tristan	<i>Product Owner</i>	NPC	
Homer	Membro da equipe	NPC	

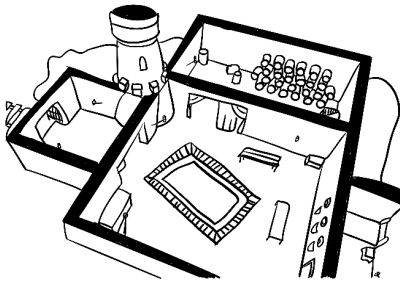
O jogo apresenta dois níveis, no primeiro é o castelo onde a equipe SCRUM trabalha e no segundo é o jardim do palácio. Como apresentado na fase de concepção do jogo, o jogo inicia no castelo, após cada membro da equipe assumir suas atividades, eles se dirigem ao jardim do palácio. No final do dia, a equipe retorna ao castelo para avaliar o resultado das atividades.

Durante o jogo, sempre que ocorre a mudança de nível apresenta-se uma imagem de uma montanha com um castelo no topo. A seguir apresenta-se o esboço desta imagem. Para exemplificação dos cenários do jogo.

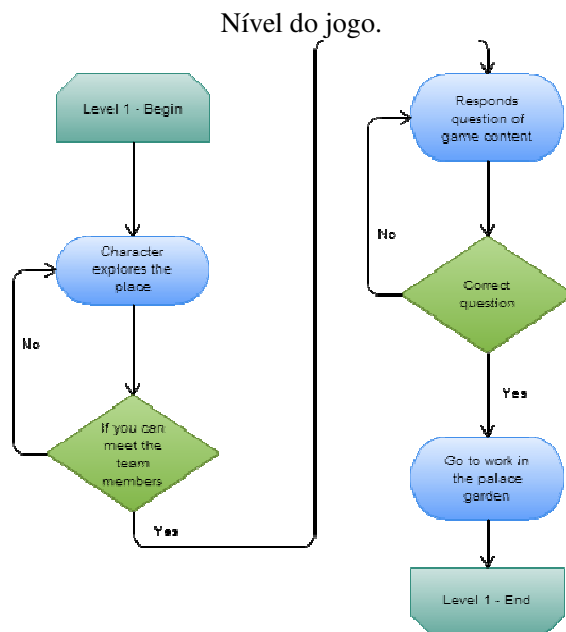
## Esboço de tela de abertura do jogo



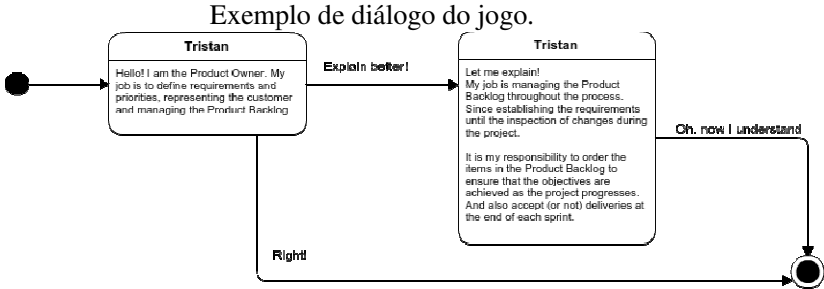
## Esboço de Castelo.



Em cada nível do jogo é criado um fluxograma com as sequências de atividades que o jogador deve realizar durante o jogo. O objetivo da criação deste fluxograma é organizar o conteúdo que será abordado em cada etapa do jogo e também criar uma sequência de ações que o jogador deve tomar para vencer o jogo.



Em um jogo o diálogo precisa estar bem inserido na narrativa do jogo, para permitir que o jogador sintá-se imerso. Para que o diálogo do jogo seja criado de modo mais estruturado, o processo ENgAGED orienta a modelagem dele em forma de diagrama de estados (OMG, 2015). No caso do jogo SCRUM'ed o feedback educacional foi incluído no próprio diálogo que ocorre entre os personagens.






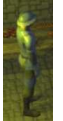


### ***F3.4 – Fase: Implementação do jogo***

Nesta fase os elementos do jogo, personagens, cenários, artefatos e outros objetos são implementados. No caso do jogo SCRUM<sup>ed</sup> os elementos são implementados utilizando a ferramenta *Unity 3D*.

Alguns personagens utilizados no jogo foram adquiridos a partir do *Unity Store*, que permite baixar ou comprar diversos personagens de jogo prontos.

Personagens do jogo implementados.

Nome	Esboço	Imagem
Jogador		
Tristan		
Homer		

Nas figuras a seguir são apresentados os cenários criados para o jogo. O primeiro é o castelo onde a equipe SCRUM trabalha e no segundo é o jardim do palácio onde a equipe SCRUM vai realizar as atividades solicitadas pelo cliente.

Exemplo de diálogo do jogo.



Exemplo de diálogo do jogo.



### ***F3.5 – Fase: Teste do jogo***

Na fase de testes foram realizados dois tipos de testes. O primeiro foi o teste de sistema, sendo realizado pelos autores do jogo, principalmente pelo autor que implementou o jogo. Neste busca-se falhas e erros no jogo, simulando o usuário final.

O segundo foi o teste de aceitação, sendo realizado por alguns componentes do laboratório GQS (Grupo de Qualidade de Software) da Universidade Federal de Santa Catarina (UFSC). Nestes testes são realizadas simulações de operações rotineiras no jogo, para verificar se o comportamento do jogo está de acordo com o solicitado.

Após a realização dos testes de sistema e de aceitação, foram realizados novos ajustes no jogo, utilizando as informações levantadas nos testes. A partir dos testes realizados, foi criada a última versão do jogo SCRUM<sup>ed</sup>.

#### **F4 – Execução da Unidade Instrucional**

A aplicação do jogo foi realizada em duas etapas. Na primeira, foi realizada a aplicação do jogo com os membros do GQS – Grupo de Qualidade de Software da Universidade Federal de Santa Catarina (UFSC). A aplicação do jogo ocorreu em maio de 2015, contando com a participação de 10 alunos de graduação e pós-graduação dos cursos de computação.

A segunda aplicação foi realizada na disciplina INE5427 - Planejamento e Gestão de Projetos, do curso de Bacharelado em Ciências da Computação da UFSC em junho de 2015, com 13 alunos em um laboratório de informática do Departamento de Informática e Estatística (INE) da UFSC.

Ao final da aplicação do jogo, cada aluno assinou o TCLE e a autorização do uso da sua imagem. Em seguida os alunos avaliaram o jogo. A pesquisa foi aprovada pelo CEPSH - Comissão de Ética em Pesquisa com Seres Humanos da UFSC sob parecer de número 1.021.539.

### Aplicação do jogo em sala de aula.



#### **F5 - Avaliação da Unidade Instrucional**

Para a avaliação do jogo SCRUM'ed foi utilizado o modelo de avaliação de jogos educacionais MEEGA (Savi, Wangenheim e Borgatto, 2011), o qual avalia a qualidade do jogo educacional do ponto de vista da motivação dos alunos, experiência dos usuários e aprendizagem.

A **motivação** dos alunos é formada pelos seguintes fatores de qualidade: atenção, confiança, relevância e satisfação.

A **experiência dos usuários** é formada pelos seguintes fatores de qualidade: competência, controle, desafio, divertimento, imersão e interação social.

A **aprendizagem** é formada pelos seguintes fatores de qualidade: aplicação, aprendizado de curto prazo, aprendizado de longo prazo, compreensão e conhecimento.

A avaliação do jogo SCRUM'ed tem por propósito avaliar a qualidade do jogo, nos fatores de motivação dos alunos, experiência dos usuários e aprendizagem, sob o ponto de vista da percepção dos alunos da disciplina Planejamento e Gestão de Projetos.

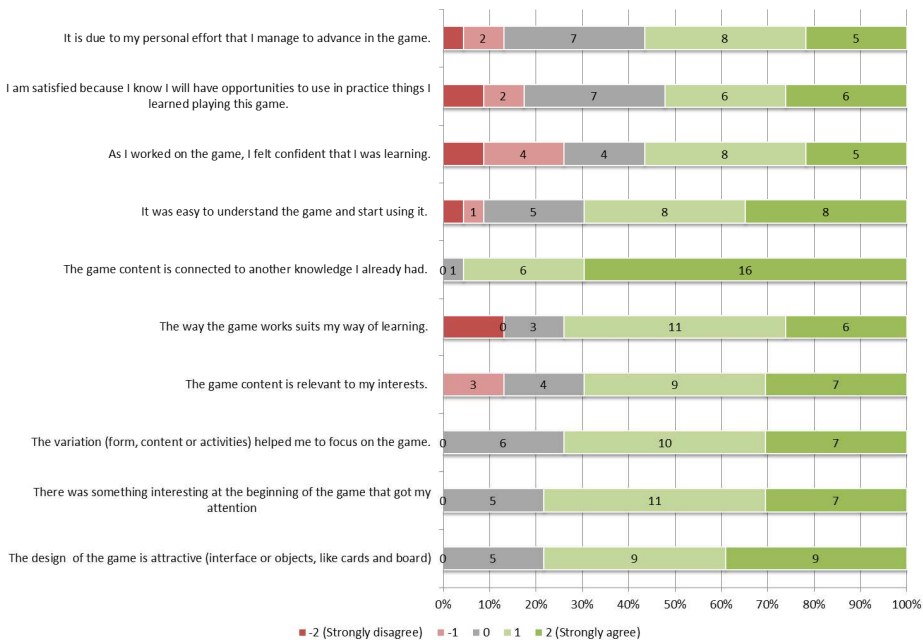
O design de pesquisa é não-experimental. Deste modo os alunos jogam o jogo e em seguida avaliam-no. O instrumento para coletar dados é o questionário do modelo MEEGA (Savi, Wangenheim e Borgatto, 2011). Por meio do questionário é possível produzir os

gráficos sobre a motivação dos alunos, experiência dos usuários e aprendizagem.

A avaliação do jogo foi realizada em maio e junho de 2015, após a aplicação do jogo. Preencheram o questionário 23 alunos dos cursos de computação da universidade.

Para a avaliação do fator de motivação dos alunos, grande parte dos alunos se consideraram focados durante a aplicação do jogo. Grande parte dos alunos sentiram-se confiantes ao jogar o jogo e relataram que o jogo é fácil de jogar. Grande parte dos alunos consideraram o jogo relevante para o aprendizado e sentiram-se satisfeitos com o jogo.

### Avaliação da motivação dos alunos.

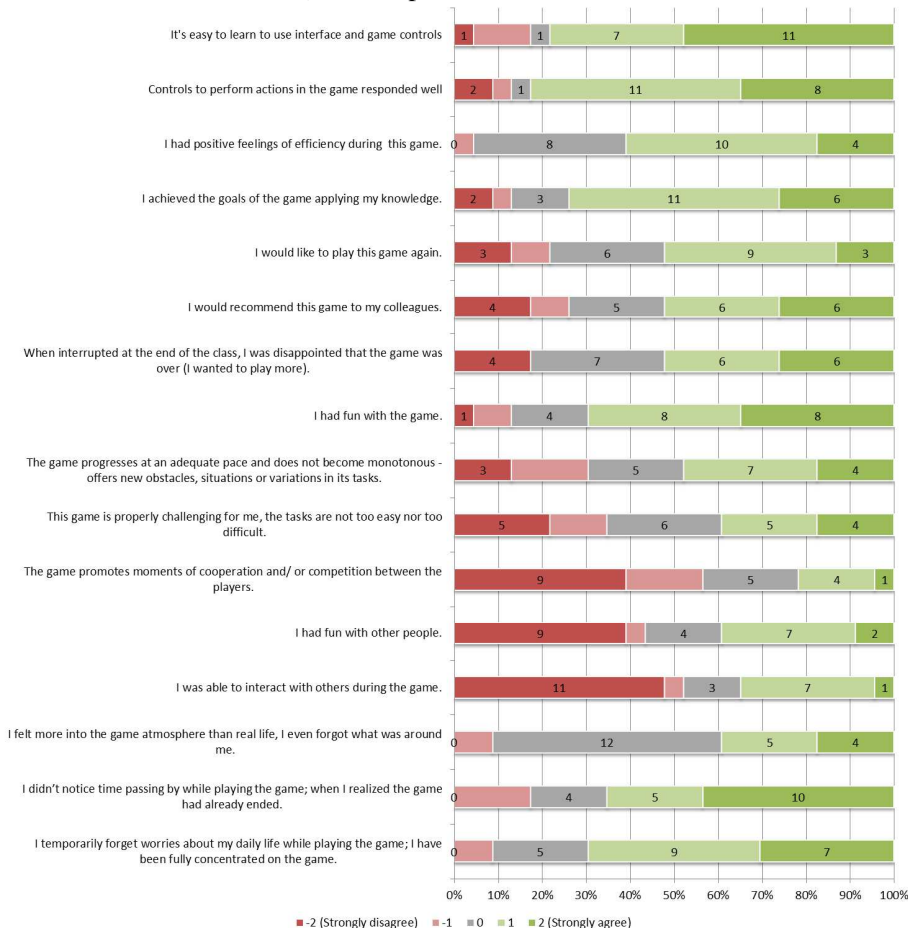


Para a avaliação do fator experiência do usuário foi identificado que a metade dos alunos consideraram o jogo desafiante. Mais da metade dos alunos utilizariam o jogo novamente e recomendariam para seus colegas. Grande parte dos alunos consideraram que o jogo desenvolve competências em relação ao conteúdo da disciplina e consideraram o jogo eficiente. Por tratar de um jogo single player, os alunos avaliaram negativamente a interação social do jogo. Entretanto, a metade dos alunos não identificaram o tempo passar enquanto jogavam



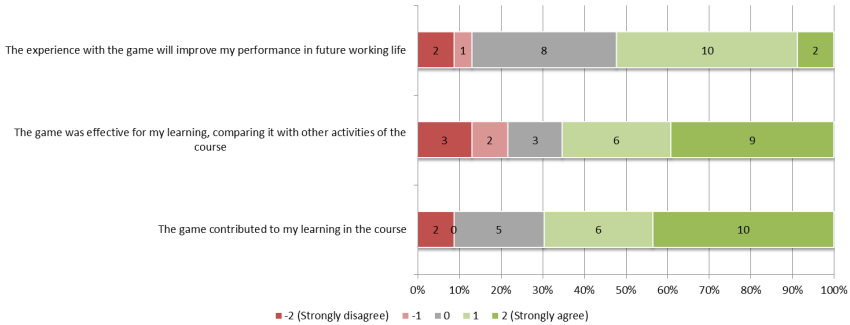
o jogo, demonstrando um resultado positivo em termos de imersão ao jogo.

### Avaliação da experiência dos usuários.



Para a avaliação do fator de aprendizagem, a metade dos alunos consideraram que o jogo contribui com sua formação pessoal. Mais da metade dos alunos consideraram que o jogo é uma estratégia instrucional eficiente, considerando outros tipos de estratégias, como aulas expositivas, leituras ou exercícios. Grande parte dos alunos consideraram que o jogo contribui significativamente para o aprendizado dos conhecimentos abordados no jogo.

### Avaliação de aprendizagem.



## JOGO PM (Project Management) Quiz

A terceira aplicação do processo ENgAGED foi a customização de um *quiz game* para ensinar conceitos de gerência de projetos, sendo criado o jogo *PM Quiz* (CASSETTARI, 2015). Nesta customização foi utilizada a ferramenta online *Kahoot*<sup>8</sup> e foi utilizada a versão mais atual do ENgAGED apresentada no apêndice E.

### F1 – Análise da Unidade Instrucional

A análise instrucional foi feita através de dois *survey*. O primeiro está disponível em *Customização de um Jogo Educacional para Revisão de Conhecimento em Gerenciamento de Projetos* (CASSETTARI, 2015). Ele teve como objetivo identificar o perfil de uso de dispositivos móveis dos alunos de graduação em computação. Os dados foram coletados por meio de um formulário de pesquisa do *google*. O *survey* foi realizado durante o mês de abril de 2015 e obteve 186 respostas.

O segundo *survey* (BATTISTELLA e WANGENHEIM, 2016b) procurou saber sobre os perfis de jogadores de jogos na área de computação. Esta pesquisa ficou online entre novembro de 2014 e abril de 2015, obtendo 406 respostas.

<sup>8</sup> <https://getkahoot.com/>

Com base nestes dados coletados foi possível definir as especificações da unidade instrucional. Em relação a contexto ou ambiente onde a unidade instrucional será aplicada foi identificado: (a) aplicação na disciplina INE5427 Planejamento e Gestão de Projetos/INE5617 Gerência de Projetos, (b) tem o objetivo de “proporcionar ao aluno uma compreensão dos principais conceitos de gerenciamento de projetos seguindo o PMBOK”,(c) tem uma quantidade estimada de 25 alunos, tendo uma variação tipicamente entre 15 à 30 alunos, (d) a unidade instrucional terá a duração de 1 hora e 40 minutos, e (e) apresenta como conteúdo programático: (1) visão geral dos grupos de processos: iniciação, planejamento, execução, monitoramento e encerramento e (2) visão geral de uma das áreas de conhecimento: tempo, custo, escopo, riscos, partes interessadas, qualidade, recursos humanos, integração, comunicações ou aquisições.

### Caracterização dos aprendizes.

<b>Caracterização dos aprendizes</b>	
<b>Faixa etária</b>	A maior parte dos alunos (75%) estão entre 19 e 30 anos.
<b>Gênero</b>	85% dos alunos são do sexo masculino, enquanto 15% do sexo feminino.
<b>Preferência de gênero de jogo</b>	A preferência de gênero de jogos dos alunos, segundo o <i>survey</i> , é ação-aventura e RPG, ambas com 26%. O gênero de <i>quiz games</i> é a preferência de somente 4% dos alunos.
<b>Preferência de plataformas de jogo</b>	Os alunos tipicamente preferem jogos de computadores ou consoles (80%), em comparação com jogos <i>mobile</i> (13%) e não-digitais (5%).
<b>Modo de interação preferido</b>	A maioria dos alunos (42%) prefere jogos com modo de interação <i>single-player</i> . Outros 23% preferem jogos <i>multiplayers</i> competitivos e 17% <i>multiplayers</i> cooperativos. Já 18% dos alunos preferem jogos multigrupos.
<b>Frequência ao jogar jogos digitais</b>	33% dos alunos joga pelo menos uma vez por semana, 26% dos alunos jogam diariamente. Os outros alunos jogam mais raramente.
<b>Cenários favoritos</b>	Os cenários favoritos dos alunos são: Medieval (51%), <i>Dungeons&amp;Dragons</i> (49%), ficção científica (49%), e esportes (26%).
<b>Persona</b>	Jailson tem 22 anos e está na segunda metade do curso de Sistemas de informação na UFSC. Natural de Xanxerê (SC), veio para Florianópolis há dois anos e meio para estudar. Atualmente, mora no bairro Trindade junto com mais dois colegas que fazem outros cursos. Desde cedo Jailson teve contato com computadores: seu pai fazia manutenções dos computadores de sua cidade, assim, desde novo, já sabia fazer reparos que demandassem um pouco mais de conhecimento em <i>software</i> e <i>hardware</i> . Apesar de sentir falta da família, Jailson pretende terminar a faculdade e permanecer na cidade, aproveitando o fato de que a empresa em que estagia já sinalizou interesse de contratação. Atualmente para passar o tempo Jailson gosta de jogar jogos <i>stand-alone</i> em seu <i>notebook</i> , já que a internet no local onde mora não é muito boa, logo, não é uma experiência agradável participar de jogos <i>online</i> .

	Apesar de seu notebook ser sua principal ferramenta de estudo durante as aulas, Jailson não gosta de levá-lo para sala de aula. Jailson prefere levar caderno e caneta para aula, e se possível, baixar no seu <i>smartphone</i> os slides das aulas de Engenharia de Software, Engenharia de Usabilidade, Gerenciamento de Projetos e Informática e Sociedade. Jailson gosta de acompanhar a aula pelo <i>smartphone</i> , mesmo que o professor não goste.	
<b>Análise de contexto</b>		
<b>Local de aplicação</b>	Laboratório de informática, para a aplicação do jogo em horário de aula. Está dividido em duas colunas: uma com os recursos do laboratório da UFSC, outra com os recursos dos alunos, baseados no resultado do survey dos perfis de uso de dispositivos móveis realizado (ANEXO II).	
	<b>Recursos do laboratório</b>	<b>Recursos dos alunos</b>
<b>Recursos</b>	A sala possui datashow e de 10 a 20 computadores.	Segundo o <i>survey</i> 160 (90%) alunos de computação levam seus dispositivos móveis para sala de aula sempre. Outros 10 (6%) levam quase sempre.
<b>Tipo de dispositivo</b>	Os computadores <i>desktop</i> do laboratório são equipados com monitores de resolução 1440x900	De acordo com o <i>survey</i> , 172 (93%) dos alunos possuem <i>smartphones</i> com tela <i>touchscreen</i> ou <i>tablets</i> , e eles os levam para sala de aula. Já outros 12 (7%) levam celulares de teclado físico ou não levam celulares.
<b>Internet</b>	No laboratório de informática a banda de conexão é de 100 (Megabits) Mbps.	Dos alunos que levam os seus dispositivos móveis para a sala de aula, 64 (36%) conseguem conectar na internet da universidade sempre. Outros 101 (57%) relataram conseguir somente às vezes.
<b>Navegador de internet</b>	Os computadores possuem os seguintes <i>browsers</i> : para Windows: Internet Explorer 9 e Chrome 36. Para o Ubuntu: Firefox 22.	Os navegadores que os alunos possuem em sua maioria são: - Google Chrome 39+, com 131 alunos (74%); - Firefox 30+, com 33 alunos (19%); - Safari 7+, com 23 alunos (13%); - Android Browser 4+, com 40 alunos (23%); Já 23 alunos (13%) possuem outros navegadores, enquanto 6 (3%) não possuem algum navegador.
<b>Recursos financeiros disponíveis</b>	Para a implementação, aplicação e avaliação deste jogo são necessários somente recursos humanos e internet.	

## F2 – Projeto da Unidade Instrucional

A **estratégia instrucional**, como parte do projeto da unidade instrucional, baseando-se na aprendizagem experiencial, será a aplicação de um jogo educacional adaptando um *quiz game* para revisar o aprendizado em gerenciamento de projetos.

Este *quiz game* será aplicado na unidade instrucional assim que os conteúdos tenham sido introduzidos. No caso das disciplinas INE5427 (Planejamento e Gestão de Projetos) e INE5617 (Gerência de Projetos), o jogo será aplicado após o conteúdo da área de conhecimento de escopo de gerenciamento de projetos tenha sido exposto.

Através do uso do laboratório, *notebooks ou smartphones* pessoais, o jogo pode ser aplicado e os alunos poderão revisar efetivamente o conteúdo, respondendo às perguntas.

Com a quantidade de respostas corretas e incorretas, os alunos poderão obter um *feedback* de conhecimento na matéria, e então consultar o professor antes do exame caso haja dúvidas, assim como o professor pode notar alguma dificuldade dos alunos, e então conseguir sanar estas dificuldades a tempo.

## F3 – Desenvolvimento do jogo educacional

### F3.1 – Análise do jogo

Levantamento dos requisitos.

Levantamento de requisitos			
Colunas: ID; Descrição; (T)Tipo – (NF) Não funcional, (F) Funcional; (P)Prioridade – (E)Essencial, (I)Importante, (D)Desejável.			
ID	Descrição	T	P
REQ-F1	Conter perguntas referentes às áreas de conhecimento e grupos de processos alinhados ao PMBOK	F	E
REQ-F2	Após o acontecimento de uma rodada, o professor pode manter o jogo parado para falar sobre a pergunta que acabou de ser respondida pelos alunos.	F	E
REQ-F3	Após responder a sequência de perguntas do jogo, o jogador terá seu relatório final ( <i>feedback</i> ). Poderá saber sua taxa de acertos, taxa de acerto dos outros alunos, classificação.	F	E
REQ-F4	O professor poderá de alguma forma editar/criar perguntas, para aplicação de sua unidade instrucional.	F	E
REQ-F5	O professor poderá alterar que conjunto de perguntas será aplicado no momento do <i>quiz</i> , caso possuir mais de um conjunto de perguntas disponível.	F	E
REQ-F6	Haverá uma contagem regressiva de tempo do início até o fim	F	I

	da pergunta. O tempo começará a contar a partir do momento que o professor iniciar a rodada.		
REQ-F7	O jogador poderá acompanhar estatísticas dos outros jogadores, como classificação, acertos, e respostas mais dadas.	F	I
REQ-NF1	O jogo não deve apresentar conteúdo difamatório, obsceno, ofensivo, odioso ou violento.	NF	E
REQ-NF2	O jogo não deve ter custo com licença.	NF	E
REQ-NF3	O jogo deve ser compatível com os sistemas operacionais: - Windows 7 e versões superiores; - Ubuntu 12.04 e versões superiores; - Android; - IOS;	NF	E
REQ-NF4	O jogo será web e terá suporte completo para os <i>browsers</i> a partir da seguinte versão: Chrome 39+, Firefox 30+, Internet Explorer 10+, Safari 7+ e Android Browser 4+. Para versões dos browsers Chrome 37-8, Firefox 29-30, Internet Explorer 9 e Safari 5-6, o jogo está com suas características limitadas. Para os browsers das versões Chrome < 37, Firefox < 28, Internet Explorer < 9, Safari < 4, Android Browser < 4 e Opera < 24, o jogo não possui suporte.	NF	E
REQ-NF5	O conteúdo deve ser em Português do Brasil (pt-br) ou Inglês.	NF	E
REQ-NF6	O jogo deve ter uma duração de até 100 minutos.	NF	E
REQ-NF7	Deve ser um jogo digital.	NF	E
REQ-NF8	Haverá a necessidade de o professor possuir um <i>datashow</i> .	NF	E
REQ-NF9	Deve existir um módulo aluno e um módulo professor.	NF	E
REQ-NF10	Serão recomendados conexões de 1, 2 ou 3Mbits para jogos entre 10, 20 e 50 pessoas respectivamente.	NF	E

### ***F3.2 – Fase: Concepção do jogo***

O jogador deve responder cada questão dentro de um tempo limite. O jogador que fizer mais pontos respondendo mais perguntas corretas no menor tempo é o vencedor.

O gênero do jogo é *quiz*, onde o jogador precisa responder perguntas para uma determinada área de conhecimento.

A plataforma utilizada é computador online e *smartphone*, sendo utilizada a ferramenta *Kahoot*.

A seguir apresenta-se a concepção dos elementos do jogo, regras, critérios para vencer o jogo, *feedback* educacional.

### Concepção do jogo.

Descrição do elemento do jogo	Tipo
<b>Classificação</b> – Após a execução de cada pergunta e no final do jogo, será mostrado a classificação dos 5 melhores jogadores com as suas respectivas pontuações. Este pacote de itens possui um apelido e uma pontuação	Pacote de itens
<b>Perguntas</b> – Será mostrada durante o jogo. Cada pergunta pode ter o máximo de 4 respostas. Cada pergunta deve ter de 1 a 4 respostas corretas. Este pacote de itens possui de 1 a 4 respostas.	Pacote de itens
<b>Respostas</b> – Cada uma das respostas pertencentes a uma pergunta	Item
<b>Pacote de perguntas</b> – Cada pacote de perguntas deve ter um nome, descrição, linguagem e dificuldade. Cada sessão de jogo será a partir da escolha de um pacote de perguntas. Este pacote de itens possui N perguntas.	Pacote de itens
<b>Game-Pin</b> – Número necessário para que os jogadores acessem a instância de um pacote de perguntas, lançadas pelo professor.	Dado
<b>Apelido</b> – Assim que os jogadores inserirem o <i>game-pin</i> , deve ser informado o apelido a ser usado nesta instância de jogo.	Dado
<b>Pontuação</b> – Ao responder corretamente as perguntas durante o jogo, o jogador receberá uma pontuação cumulativa até o final do pacote de perguntas	Dado
<b>Relógio</b> – Quanto tempo falta para que o jogador possa responder a questão e o resultado seja mostrado	Dado
<b>Questionário de satisfação</b> – Ao final do jogo, os alunos são convidados a responder perguntas de satisfação. Este pacote de itens possui 4 itens de questionário de satisfação.	Pacote de itens
<b>Item do questionário de satisfação</b> – Os itens do questionário são: - Nota para diversão (com valores de 1 a 5); - Recomenda o jogo (sim ou não); - Aprendizado com o jogo (sim ou não); e - Como o jogador se sentiu (com opções em representações de um rosto feliz, sem expressão ou triste).	Item
<b>Gráfico de respostas</b> – Após a execução de uma pergunta, aparecerá na tela do professor um gráfico com a quantidade de respostas dadas pelos alunos.	Dado
<b>Planilha de resultados de um jogo</b> – Após o final do jogo, estará disponível para o professor uma planilha eletrônica com o relatório das respostas dadas pelos alunos.	Planilha
<b>Regras</b>	
Descrição da regra do jogo	
Somente os jogadores que responderem corretamente a uma pergunta receberão pontos.	
A quantidade de pontos que um jogador receberá dependerá de quão rápido este jogador responder à questão, com um valor decrescente de 1000	
Os jogadores devem responder às questões dentro do tempo limite definido para a questão	
Critérios de vitória	
Critérios	<p>Será vencedor do jogo aquele jogador que responder mais perguntas corretas em menos tempo.</p> <p>Quanto mais cedo o aluno responder a pergunta e for a opção correta, mais perto de 1000 serão os pontos recebidos. Os jogadores subsequentes que marcarem a resposta correta terão sua pontuação reduzida de acordo com a demora para marcá-la.</p> <p>Ao final, serão somados todos os pontos do jogador. O jogador que tiver</p>

	a maior quantidade de pontos será o vencedor.
<i>Feedbacks</i> educacionais ao jogador	
<i>Feedback</i>	Após passar o tempo limite ou todos os alunos responderem a pergunta, será mostrado a resposta correta e a quantidade de jogadores que marcaram cada uma das respostas. Neste momento o professor também terá tempo hábil para conversar com os alunos sobre a questão e responder a dúvidas.

### ***F3.3 – Design do jogo***

Durante a análise do estado da arte realizado por Cassettari (2015) foi possível identificar a existência de muitas ferramentas que serviam como modelos de *quiz games*. Porém, destes modelos nenhum havia sido explorado como ferramenta para o ensino de gerenciamento de projetos em disciplinas de graduação em computação.

Estas ferramentas foram analisadas e, dentre elas, a escolhida para ser usada no jogo foi o *Kahoot*<sup>9</sup>. O *Kahoot* se destacou e foi escolhido por sua facilidade de manutenção, *layout* agradável, boa usabilidade, possibilidade de uso online, adaptabilidade para dispositivos móveis, presença dos relatórios dos *quiz* aplicados e interatividade.

O *Kahoot* é uma plataforma de *quis games* web, onde é possível cadastrar grupos de perguntas e respostas, lançar sessões de jogos para os jogadores jogarem e convidar jogadores para estas sessões, onde o jogador poderá responder às perguntas do *quiz game*.

Página inicial do Kahoot.

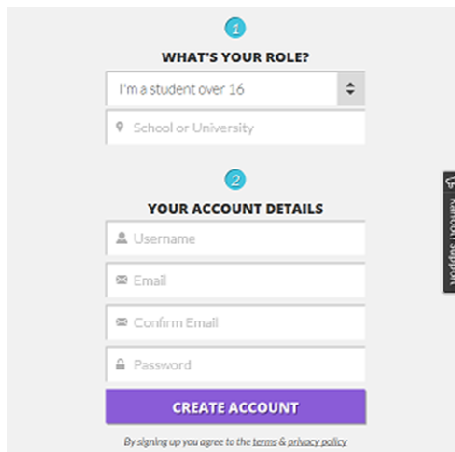


<sup>9</sup> <https://getkahoot.com>



Após a realização do cadastro é possível jogar a qualquer momento qualquer conjunto de perguntas e respostas que estejam criadas em modo público.

### Cadastro de usuários no Kahoot.



The image shows the Kahoot! user registration interface. It is divided into two main sections: 'WHAT'S YOUR ROLE?' and 'YOUR ACCOUNT DETAILS'. In the first section, there is a dropdown menu with 'I'm a student over 16' selected and a text input field for 'School or University'. The second section contains four input fields: 'Username', 'Email', 'Confirm Email', and 'Password'. A purple 'CREATE ACCOUNT' button is at the bottom. A vertical 'Kahoot! Support' button is on the right. At the bottom, a small note reads 'By signing up you agree to the terms & privacy policy'.

Também após o cadastro estará habilitada a criação de múltiplos grupos de perguntas e respostas. Este grupo de perguntas e respostas necessitam serem cadastradas com um certo limite de caracteres: perguntas precisam conter no máximo 95 caracteres, e as respostas, 65 caracteres. Para cada pergunta também há a possibilidade de:

- Inserção de uma imagem.

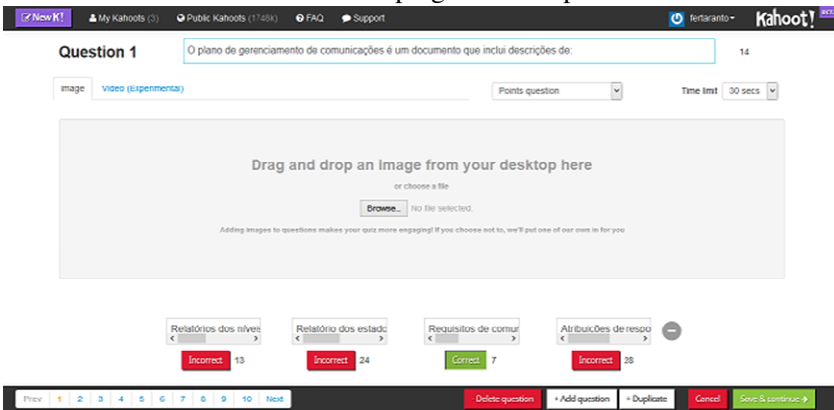
- Determinar a quantidade de tempo que esta pergunta durará em sua rodada.

- Marcar se esta pergunta valerá ou não pontos para respostas corretas.

- Cadastrar de uma a quatro possíveis respostas.

- Marcar de uma a quatro respostas corretas dentre as cadastradas.

## Cadastro de perguntas e respostas.



Não é necessário cadastro para jogá-lo, porém, caso seja de interesse criar grupos de perguntas e respostas, deve ser feito um cadastro sem custos. Apesar de todos os menus do jogo ser em inglês, o jogo não está restrito a criação de perguntas e respostas nesta língua, possibilitando o acesso de jogadores não falantes da língua inglesa.

Após finalizar a criação do grupo de perguntas e respostas é permitido o cadastro de algumas características deste conjunto de perguntas, como: Nome.Linguagem (inglês, português, espanhol), apesar desta opção não alterar a linguagem dos menus durante a execução do jogo. Se deve ser um conjunto de perguntas e respostas que estará sempre disponível (público) ou somente poderá ser respondido caso o dono a ative temporariamente (privado). Que público este conjunto de perguntas está direcionado (para universidades, escolas, negócios, treinamentos, eventos ou social). Adicionar alguma descrição para este conjunto de perguntas.

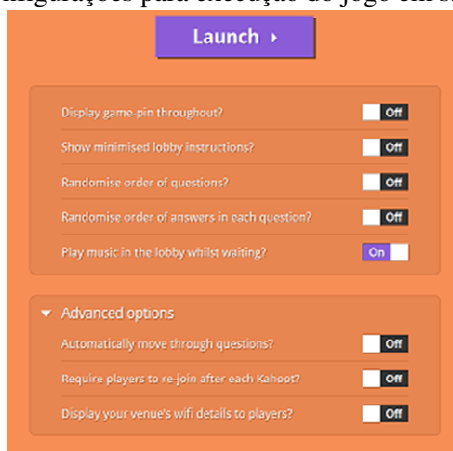
Há a possibilidade de simular o jogo, de forma que a tela fique dividida ao meio: ao lado esquerdo mostrará a visão do professor, com a pergunta, imagem, contador de tempo e cada uma das respostas; e ao lado direito com a apresentação de um dispositivo móvel simulando a tela do aluno, somente com as opções de respostas. Como este jogo somente mostra o texto da pergunta e das respostas na tela do professor, há a necessidade de que ele esteja sendo jogado em um projetor dentro de sala de aula. Há nesta simulação as mesmas características do jogo real, a não ser por esta divisão da tela.

### Simulação pós cadastro das perguntas.



Depois de cadastradas as perguntas e respostas, há algumas configurações em que o professor pode escolher ao lançar o seu jogo para os alunos em sala de aula.

### Configurações para execução do jogo em sala de aula.

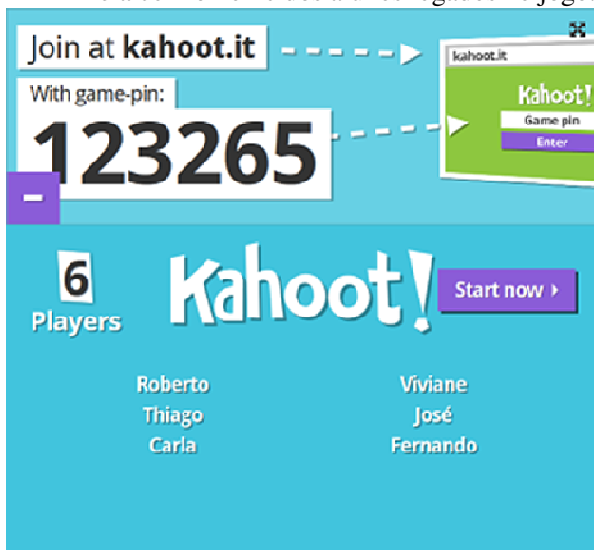


Após marcar as configurações desejadas para esta sessão de jogo, o professor do jogo receberá um código. Este código deverá ser passado

para os jogadores que, ao acessarem o link de jogo (<https://kahoot.it/>) conseguirão acesso. A inserção deste código não requer *login* dos jogadores, somente a inserção de um nome que valerá somente para esta sessão do jogo

Para a execução do jogo, assim que estes alunos colocarem o código e seus nomes no jogo aparecerão seus nomes na tela do professor.

Tela com o nome dos alunos logados no jogo.



Após a confirmação do professor, há o início do jogo. Haverá sucessivas perguntas, e a sistemática dependerá das configurações marcadas nos passos anteriores. A cada final de rodada aparecerá na tela do professor a resposta correta e a quantidade de alunos que responderam a cada alternativa. Além desta tela, será mostrado também o ranking de pontos dos jogadores até o momento.

Quantidade de respostas por alternativa.

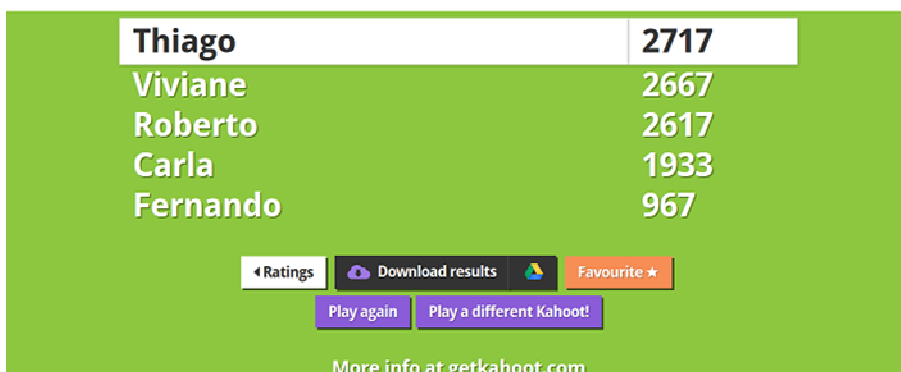
**Q2** Como um time de GP acompanha o trabalho e as ações dos membros do time de um projeto?



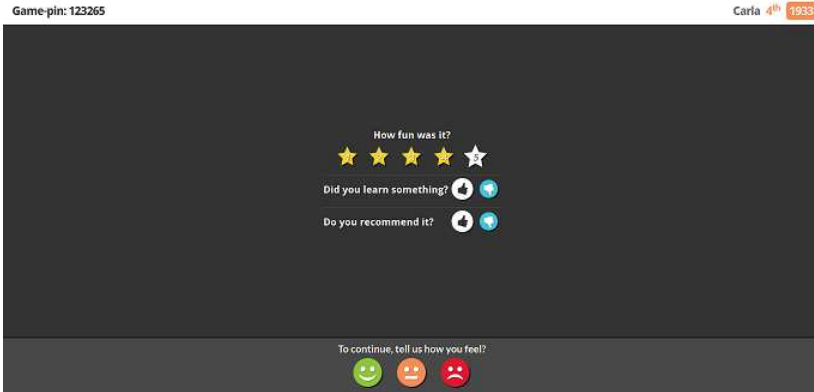
Assim que o *quiz game* acaba aparece na tela do professor a classificação de pontos dos alunos mostrando as pontuações mais altas, e uma tela mostrando os acertos do vencedor.

Classificação final.

## Final scoreboard



## Pesquisa com os alunos sobre o jogo.



### *F3.4 – Implementação do jogo*

A seguir apresenta-se a customização referente ao desenvolvimento do jogo educacional. Para o caso do PM Quiz, esta é a fase em que são desenvolvidas as perguntas e respostas a serem apresentadas durante o jogo, assim como algumas configurações sobre o cadastro destas perguntas.

### Concepção do jogo.

Pergunta		Respostas –* (x) significa resposta correta
<b>Área de conhecimento: Escopo</b>		
1	Qual das alternativas a seguir ocorre durante a criação de uma EAP?	<ul style="list-style-type: none"> <li>- Análise do produto</li> <li>- Seleção de projetos</li> <li>- (x) Decomposição</li> <li>- Identificação de alternativas</li> </ul>
2	O projeto deve ser concluído em um ano. Você deve declarar isto no escopo do projeto como:	<ul style="list-style-type: none"> <li>- Critérios de aceitação</li> <li>- Entregas</li> <li>- Premissa</li> <li>- (x) Restrição</li> </ul>
3	O dicionário da EAP é um documento que descreve:	<ul style="list-style-type: none"> <li>- Termos técnicos usados para o gerenciamento de escopo</li> <li>- (x) Detalhes de cada componente da EAP</li> <li>- Termos essenciais da EAP para equipes globais</li> <li>- Requisitos funcionais do sistema</li> </ul>
4	Que item tipicamente não é uma entrada do processo de definição de escopo?	<ul style="list-style-type: none"> <li>- Plano de gerenciamento do escopo</li> <li>- Termo de abertura do projeto</li> <li>- Contrato do projeto</li> </ul>

		- (x) Matriz de rastreabilidade dos requisitos
5	Até que ponto frequentemente é decomposto o escopo a um pacote de trabalho?	- (x) Até ficar fácil de medir e gerenciar seu custo e duração - Até possibilitar descrição minuciosa do pacote - Até possibilitar a realização por um único recurso - Até o pacote durar uma hora
6	Qual documento possui descrição detalhada do pacote de trabalho?	- (x) Dicionário da Estrutura Analítica do projeto (EAP) - Lista de Atividades - Indicação do Escopo do Projeto Preliminar - Plano de Gerenciamento do Escopo do Projeto
7	Qual das seguintes definições descreve MELHOR o que é o escopo de projeto?	- (x) Somente o trabalho que precisa ser realizado para a entrega - Funções do produto, descritas no termo de abertura - Uma descrição das necessidades do cliente - A lista de requisitos funcionais concordados no contrato.
8	Na gerência de requisitos, a principal causa de fracasso?	- Requisitos funcionais e requisitos financeiros - (x) Requisitos descontrolados e requisitos mal entendidos - Requisitos formais e requisitos jurídicos - Requisitos dos <i>stakeholders</i> e requisitos da equipe
9	Todos os seguintes itens são parte da <i>baseline</i> do escopo, EXCETO:	- Requisitos financeiros - (x) Requisitos descontrolados e mal entendidos - Requisitos formais - Requisitos da gerência sênior
10	Uma entrada para declaração do escopo tipicamente é:	- Estrutura Analítica de Riscos - (x) Termo de abertura de projeto - Estrutura Analítica do Projeto (EAP) - Decomposição
<b>Área de conhecimento: Tempo</b>		
1	A estimativa dos recursos de atividade é uma estimativa...	- quantitativa de materiais de apoio para a equipe - de pessoas e competências - (x) de papéis, materiais e quantidades - de tempo disponível por recurso
2	Para que se usa um organograma na estimativa de recursos de atividades?	- Identificar um especialista caso haja problemas nesta atividade - (x) Identificar papéis capazes de realizar as atividades - Identificar a quem reportar no final de cada atividade - Identificar as pessoas a serem alocadas para realizar as atividades.
3	O que deve ser estimado em "Estimar a duração das	- (x) Tamanho/complexidade, esforço e duração

	atividades”?	<ul style="list-style-type: none"> <li>- Somente a duração da atividade</li> <li>- A quantidade de recursos e de tempo para realizar a atividades</li> <li>- Tempo no calendário</li> </ul>
4	Planning Poker é uma técnica de estimativa baseada em:	<ul style="list-style-type: none"> <li>- Analogia (Dados históricos)</li> <li>- Modelos algorítmicos/paramétricos (Funções matemáticas)</li> <li>- (x) Opinião especializada (experiência e intuição)</li> <li>- Técnica de três pontos (pert)</li> </ul>
5	A unidade de estimativa de 'Dia ideal' MELHOR se define em um dia que:	<ul style="list-style-type: none"> <li>- Todos os recursos estão disponíveis</li> <li>- Não há nenhum recurso ocioso</li> <li>- (x) Há um progresso ideal, de esforço ininterrupto e produtivo</li> <li>- Todos os recursos estão ocupados com atividades</li> </ul>
6	O que é a estimativa de três pontos?	<ul style="list-style-type: none"> <li>- Entrega em três partes: versões básica, avançada e completa</li> <li>- Uso de três pontos: inicial, intermediário e final.</li> <li>- Estimativa p/ três marcos: planejamento, execução e finalização</li> <li>- (x) Uso de três estimativas: otimista, provável e pessimista</li> </ul>
7	Usando a estimativa de três pontos dado $t_p = 5$ , $t_m = 6,5$ e $t_o = 8$ marque a esperada	<ul style="list-style-type: none"> <li>- 6</li> <li>- (x) 6,5</li> <li>- 7</li> <li>- 7,5</li> </ul>
8	Ao estimar as durações das atividades, o que NÃO é indicado fazer?	<ul style="list-style-type: none"> <li>- Se a estimativa é em esforço ou outra unidade, converter em tempo</li> <li>- Levar em consideração competência e produtividade do recurso</li> <li>- Elaborar possíveis datas, evidenciando incertezas</li> <li>- (x) É sempre melhor contar com horas extras ou terceirização</li> </ul>
9	Por que o caminho crítico é tão importante? Porque...	<ul style="list-style-type: none"> <li>- São as atividades mais longas</li> <li>- Estas atividades são chaves para a entrega do projeto</li> <li>- (x) Qualquer atraso nestas atividades resulta no atraso do projeto</li> <li>- São as atividades mais complexas</li> </ul>
10	Qual destas técnicas envolve realização de duas atividades em paralelo?	<ul style="list-style-type: none"> <li>- Crashing</li> <li>- (x) Fasttracking</li> <li>- Nivelamento</li> <li>- Método da corrente crítica</li> </ul>
11	A atribuição de muitos recursos para a mesma atividade resulta em:	<ul style="list-style-type: none"> <li>- (x) Redução da produtividade</li> <li>- Aumento da produtividade</li> <li>- Redução da necessidade de recursos especializados</li> <li>- Aumento de produção proporcionalmente ao num. de recursos</li> </ul>
12	Como se chama a definição do	<ul style="list-style-type: none"> <li>- Alocação</li> </ul>



	cronograma em função da disponibilidade de recursos?	<ul style="list-style-type: none"> <li>- Particionamento</li> <li>- (x) Nivelamento</li> <li>- Quantificação</li> </ul>
13	Para o método do caminho crítico, qual a MELHOR definição para folga?	<ul style="list-style-type: none"> <li>- (x) Atraso total permitido sem interferir no prazo do projeto</li> <li>- Dias em que um recurso humano não trabalha</li> <li>- Dias em que nenhum recurso humano trabalha</li> <li>- Divisão de trabalho para que não haja sobrecarga do recurso</li> </ul>
<b>Área de conhecimento: Custos</b>		
1	No ciclo de vida de custo, onde se concentra a maior quantidade de gastos?	<ul style="list-style-type: none"> <li>- Início do projeto</li> <li>- Planejamento e preparação</li> <li>- (x) Execução do trabalho</li> <li>- Encerramento do projeto</li> </ul>
2	Tipicamente qual é o maior componente de custo de um projeto de software?	<ul style="list-style-type: none"> <li>- (x) Recursos humanos</li> <li>- Software</li> <li>- Redes e comunicação</li> <li>- Computadores</li> </ul>
3	O que é reserva de contingência?	<ul style="list-style-type: none"> <li>- São provisões para uma necessidade confirmada futuramente</li> <li>- (x) São provisões para imprevistos conhecidos</li> <li>- São provisões especificamente para multas</li> <li>- É a entidade onde a equipe do projeto pode requisitar provisões</li> </ul>
4	O que é orçamento?	<ul style="list-style-type: none"> <li>- Todo o recurso financeiro disponível</li> <li>- Somente a soma do custo de todas as atividades</li> <li>- A soma do custo de todas as atividades mais as reservas</li> <li>- (x) Previsão de receitas e despesas futuras</li> </ul>
5	O orçamento de um projeto é determinado pelos custos dos seguintes, MENOS:	<ul style="list-style-type: none"> <li>- Reserva de contingência</li> <li>- Recursos</li> <li>- Reserva gerencial</li> <li>- (x) Reserva de segurança</li> </ul>
6	A reserva gerencial é:	<ul style="list-style-type: none"> <li>- (x) Orçamento reservado para riscos não previstos</li> <li>- Reserva destinada para custos gerenciais</li> <li>- Reserva destinada para serviços</li> <li>- Reserva destinada para produtos</li> </ul>
7	Quanto tipicamente é o valor da reserva gerencial em relação abaseline?	<ul style="list-style-type: none"> <li>- (x) 5%-10%</li> <li>- 10%-15%</li> <li>- 15%-20%</li> <li>- 20%-25%</li> </ul>
8	A baseline (do desempenho) dos custos NÃO mostra:	<ul style="list-style-type: none"> <li>- Controle e desempenho de custos geral do projeto</li> <li>- A reserva de contingência</li> <li>- (x) A reserva de gerenciamento</li> <li>- O fluxo de caixa esperado</li> </ul>
9	Que alternativa tipicamente representa a imprecisão das estimativas de custo?	<ul style="list-style-type: none"> <li>- (x) Iniciação: -50% a +100%</li> <li>- Planejamento: -10% a +10%</li> <li>- Planejamento: -20% a +20%</li> </ul>

		- Execução: -50% a +100%
10	Sobre CUSTOS, marque a FALSA:	- Posso usar unidades como pessoa/h para estimar custos - Preço a cobrar = Custo + Lucro - Geralmente é expressado em unidades de alguma moeda - (x) Preço a cobrar deve ser maior que o custo

### F3.5 – Fase: Teste do jogo

A seguir apresenta-se os testes realizado com o jogo, após a inserção das perguntas no *Kahoot*.

#### Realização de testes do jogo educacional.

Realização de testes		
Características:		
Tipo de teste - u (Unidade), i (Integração), s (Sistema), a (Aceitação), r (Regressão)		
Tipo de bug – c (Crash), g (Crítico/Grave), m (Menor), n (Não se aplica)		
Status – a (Aprovado), r (Reprovado)		
ID	Descrição do teste	Características:
1	Compatibilidade com computador, <i>tablet</i> e celular. Foram usados: - <i>Notebook</i> Acer Aspire intel 2.2GHz 4GB RAM com Windows 7 nos navegadores Mozilla Firefox 36 e Google Chrome 41. - Celular Motorola XT890 “Razr I”, com os navegadores Google Chrome 40 e Mozilla Firefox 36. - Celular Moto X, com os navegadores Google Chrome 40 e Mozilla Firefox 36. - Samsung Galaxy S3, com os navegadores Google Chrome 40 e Mozilla Firefox 36. - Tablet Ipad Mini 2, IOS 8.2, com o navegador Mobile Safari. - Macbookair 13”, OS Yosemite 10.10.2, com o navegador Safari 8.0.4. Este teste está alinhado ao requisito REQ-NF3 e REQ-NF4.	s / n / a
2	Capacidade de uso de conexões de 1MB, alinhado ao requisito REQ-NF10.	s / n / a
3	Obtenção de relatório detalhado ao final do quiz game executado, alinhado ao requisito REQ-F3.	s / n / a
4	Criação e edição de perguntas, alinhado ao requisito REQ-F4.	s / n / a
5	Ausência de conteúdo difamatório, obsceno, ofensivo, odioso ou violento, alinhado ao requisito REQ-NF1.	s / n / a
6	Escolha de diferentes conjuntos de perguntas momentos antes do início do <i>quiz game</i> , alinhado ao requisito REQ-F5.	s / n / a
7	Observação e verificação das estatísticas, alinhado ao requisito REQ-F8.	s / n / a
8	Ausência de conteúdo em línguas fora português ou inglês,	s / n / a

alinhado ao requisito REQ-NF5.
--------------------------------

#### F4 – Execução da Unidade Instrucional

Esta é a fase em que se busca executar a unidade instrucional, após ter sido analisada, projetada e desenvolvida.

#### Planejamento de execução.

<b>Planejamento de execução</b>		
<b>Tipo de aplicação</b>	A aplicação será no laboratório, apesar de ser indicado o uso dos dispositivos móveis pessoais dos alunos. No caso dos alunos não possuírem dispositivos móveis ou eles não estarem dentro dos requisitos, então o uso do computador do laboratório será incentivado	
<b>Equipamentos eletrônicos</b>	<b>Características</b>	<b>Descrição</b>
	Hard Disk (HD)	Não é relevante
	Memória RAM	Não é relevante
	Sistema Operacional	Windows 7 e Ubuntu 12.04 (Linux)
	Velocidade de banda larga	No laboratório de informática a banda de conexão é de 100 (Megabits) Mbps.
	Resolução do monitor	O laboratório é equipado com monitores de resolução 1440x900
	Quantidade	Há de 10 a 20 computadores no laboratório
<b>Equipamentos de suporte</b>	Há a necessidade de um projetor para apresentação do jogo	
<b>Materiais</b>	Não se aplica	
<b>Tempo</b>	Será necessário 40 minutos para aplicação do jogo	
<b>Alunos</b>	O ambiente deve estar preparado para a recepção dos 29 alunos matriculados na disciplina INE5427 – Planejamento e Gestão de Projetos	

#### F5 – Avaliação da Unidade Instrucional

A execução das duas instâncias de jogo aconteceram em Março e Abril de 2015 nos laboratórios do departamento do INE - Informática e Estatística - da Universidade Federal de Santa Catarina, com os alunos da disciplina INE5617 – Gerência de Projetos do curso de Bacharelado em Sistemas de Informação/INE/UFSC e com os alunos da disciplina INE5427 – Planejamento e Gestão de Projetos do curso de Bacharelado em Ciências da Computação/INE/UFSC. Ao todo participaram da avaliação 33 alunos, 13 deles da disciplina INE5617 e 20 na disciplina INE5427.

Alunos da disciplina INE5617 – Gerência de projetos jogando o jogo.



Ambos os jogos ocorreram no meio das aulas de planejamento de projetos de escopo, logo após os alunos já terem sido introduzidos ao assunto. Devido a matéria de escopo ter sido o assunto mais recente passado pelo professor, foi escolhido o grupo de perguntas e respostas de escopo para aplicação com os alunos.

No primeiro momento, foi explicado para os alunos o objetivo da pesquisa e foi requisitado a assinatura dos mesmos com o consentimento do uso do direito de imagem e participação na pesquisa.

A partir da obtenção das assinaturas dos alunos, foi explicado a sistemática do jogo, assim como foi requisitado o *login* dos alunos no jogo com os seus dispositivos móveis. Os alunos que não tinham disponível seus dispositivos móveis foram instruídos a usar os computadores do laboratório. Já que os alunos ainda precisaram preparar seu dispositivo móvel ou computador, entrar no site do jogo, digitar o código da sessão de jogo do *PM Quiz* e colocar seus apelidos, esta etapa durou alguns minutos.

Para a sessão de jogo de escopo aplicada aconteceram dez perguntas. A sequência de passos em cada uma das dez perguntas foi:

Apresentação somente da pergunta durante 5 segundos.

Apresentação da pergunta mais as respostas por 1 minuto na disciplina INE5427 (Posteriormente na turma INE5617 o tempo foi reduzido para 30 segundos já que houve feedback de que 1 minuto era considerado muito tempo).

Ao terminar o tempo ou todos os alunos responderem, foi mostrado a resposta certa e quantos alunos responderam a cada uma das questões (não é apresentado quais alunos).

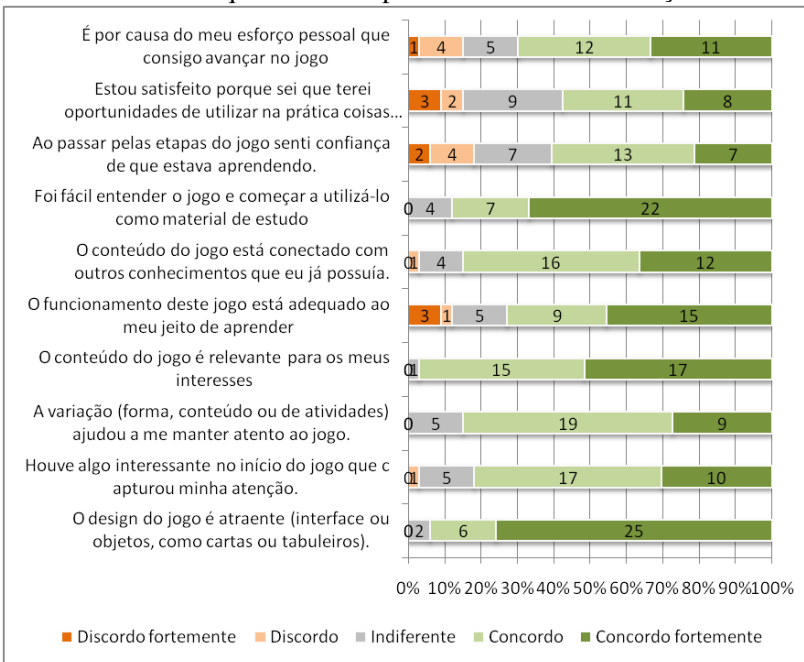
Com a resposta apresentada no projetor, o professor teve a oportunidade de conversar com os alunos sobre a quantidade de acertos, a pergunta, o porquê da resposta certa e porquê das outras respostas estarem erradas.

Após a conversa o professor avançou e foi mostrado a classificação atualizada com somente os 5 melhores alunos.

Ao final das 10 perguntas foi apresentado a classificação dos alunos novamente e qual foi o aluno com maior pontuação. Após a apresentação da classificação final, foi mostrado para os alunos em seus dispositivos móveis uma pesquisa de satisfação para feedback da unidade instrucional. Enquanto os alunos respondiam, as estatísticas das respostas eram apresentadas no projetor em tempo real.

Neste momento, também foi requisitado aos alunos para preencherem o questionário MEEGA (SAVI; WANGENHEIM; BORGATTO, 2011) para uma avaliação mais específica da unidade instrucional, além da proporcionada pelo próprio jogo.

### Frequência de respostas referente a motivação.



A partir das respostas dos alunos é possível notar a *feedback* positivo dos alunos em relação a motivação, levando em conta que 79% das respostas dos alunos foram ‘Concordo’ ou ‘Concordo fortemente’. As seguintes afirmações foram avaliadas muito positivamente:

‘O conteúdo do jogo é relevante para os meus interesses’ com 32 respostas em acordo;

‘O design do jogo é atraente’, com 31 respostas em acordo, sendo destas 25 respostas ‘Concordo fortemente’; e

‘Foi fácil entender o jogo e começar a utilizá-lo como material de estudo’, com 29 respostas em acordo.

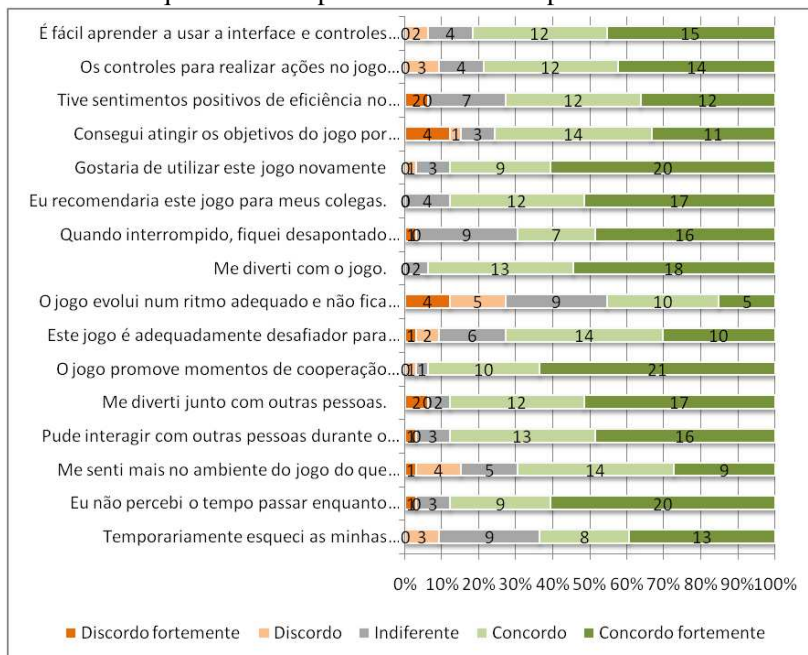
Estes itens foram avaliados positivamente pela facilidade e simplicidade de jogar o *Kahoot*. Além disso, este tipo de unidade instrucional contribui diretamente para a revisão do conteúdo lecionado nos últimos dias de aula.

A afirmação com menos valores favoráveis foi ‘Estou satisfeito porque sei que terei oportunidades de utilizar na prática coisas que aprendi com o jogo’, com 19 respostas positivas. Possivelmente este é um dos itens com menor quantidade de respostas positivas pois nem

sempre os alunos pensam diretamente na aplicação prática do conteúdo aprendido.

Estes valores mostram que o jogo educacional efetivamente atingiu os objetivos de motivação, já que os alunos principalmente relataram que o jogo condiz com seus interesses, é fácil de usá-lo como material de estudo e possui design atraente.

### Frequência de respostas referente a experiência do usuário.



A partir das respostas dos alunos é possível notar o *feedback* positivo dos alunos em relação a experiência do usuário, levando em conta que 78,5% das respostas dos alunos foram 'Concordo' ou 'Concordo fortemente'. As seguintes afirmações foram avaliadas muito positivamente:

‘Eu não percebi o tempo passar enquanto jogava, quando vi o jogo acabou’ com 29 respostas em acordo;

‘O jogo promove momentos de cooperação e/ou competição entre as pessoas que participaram’, com 31 respostas em acordo, sendo destas 21 respostas ‘Concordo fortemente’; e

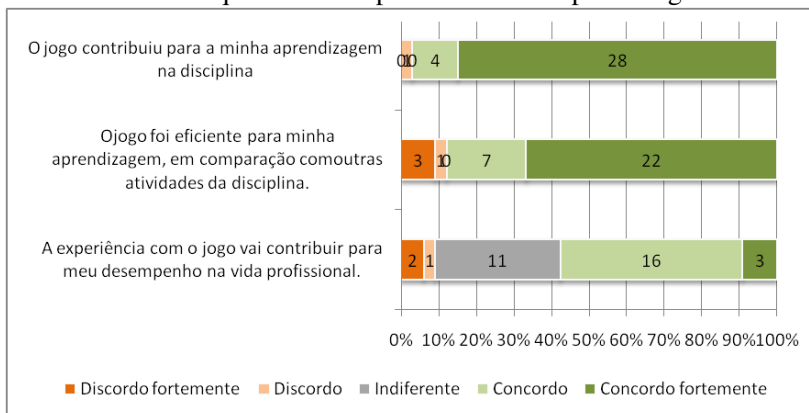
‘Me diverti com o jogo’, com 31 respostas em acordo.

Estes itens foram avaliados muito positivamente por ser um tipo de unidade instrucional fora do comum, mais leve e que estimula interação e debate entre o professor e os alunos.

A afirmação com menos valores favoráveis foi ‘O jogo evolui num ritmo adequado e não fica monótono – oferece novos obstáculos, situações ou variações de atividades’, com 15 respostas em acordo. Possivelmente este item teve menos respostas positivas, pois para os alunos que já sabiam o assunto, ainda precisavam esperar a explicação do professor novamente sobre a resposta da questão. Outro ponto que pode ter levado a menor quantidade de respostas positivas neste item foram as situações em que houve a resposta rápida do aluno e a necessidade do mesmo esperar por todos os colegas responderem ou o tempo da pergunta acabar para dar prosseguimento ao jogo.

Estes valores mostram que o jogo educacional proporcionou aos alunos uma ótima experiência, já que os alunos relataram uma interface agradável, divertimento e boa interação com os colegas de classe.

#### Frequência de respostas referente aprendizagem.



A partir das respostas dos alunos é possível notar o *feedback* positivo dos alunos em relação a aprendizagem, levando em conta que 81% das respostas dos alunos foram ‘Concordo’ ou ‘Concordo fortemente’. As seguintes afirmações foram avaliadas muito positivamente:

‘O jogo contribuiu para a minha aprendizagem na disciplina’, com 32 respostas em acordo, sendo destas 28 respostas ‘Concordo fortemente’.



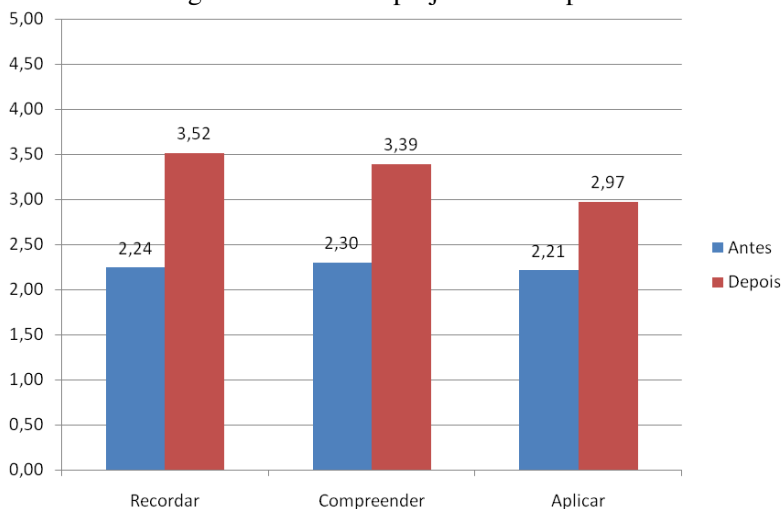
‘O jogo foi eficiente para minha aprendizagem, em comparação com outras atividades da disciplina’, com 28 respostas positivas.

Estes dois itens se destacam, porque esta unidade instrucional revisa diretamente assuntos relevantes da disciplina, e que provavelmente serão assuntos questionados em alguma avaliação.

A afirmação com menos valores favoráveis foi ‘A experiência com o jogo vai contribuir para o meu desempenho na vida profissional’, com 18 respostas em acordo. Possivelmente este é um dos itens com menor quantidade de respostas positivas, pois nem sempre os alunos sonham em ocupar o cargo que levará em conta os assuntos do conteúdo aprendido, mesmo que precise lembrá-lo indiretamente.

Estes valores mostram que o jogo educacional atingiu ótimas pontuações nos quesitos de aprendizagem, já que os alunos consideraram que a aplicação do jogo contribuiu para seu aprendizado.

Nível de conhecimento na área de conhecimento de gerenciamento de projetos – Escopo.



Nos itens de nível de conhecimento os alunos foram convidados a quantificar em uma nota de 0 a 5 para o quanto consideravam recordar, compreender e aplicar suas competências sobre a área de conhecimento de gerenciamento de projetos de escopo, um valor tanto para antes quanto para depois do jogo educacional.

De acordo com os dados fornecidos pelos alunos, eles consideraram que passaram a recordar 1.27 pontos a mais do que

recordavam antes do jogo. Para compreensão, os alunos consideraram compreender 1.09 pontos a mais depois da aplicação do jogo. Já aplicar, os alunos consideraram que sua competência aumentou em 0.76 pontos.

Estes valores mostram que o jogo educacional efetivamente atingiu os objetivos de aprendizagem propostos, já que os alunos consideraram que aprenderam com a aplicação do jogo.

## **JOGO FUGA DE GAMBIX**

A quarta aplicação do processo ENgAGED foi a produção de um jogo não-digital chamado *Fuga de Gambix* (JUNCKES, 2016). Para o desenvolvimento foi utilizado a versão mais atual do ENgAGED apresentada no apêndice E.

### **F1 – Análise da Unidade Instrucional**

A análise foi iniciada pela especificação da UI do jogo, sendo detalhada na tabela a seguir.

#### Caraterísticas da Disciplina.

<b>Informações Gerais</b>	
<b>Universidade</b>	Universidade Federal de Santa Catarina
<b>Curso</b>	Ciência da Computação/Sistemas de Informação
<b>Disciplina</b>	INE5427 Planejamento e Gestão de Projetos INE5617 Gerência de Projetos
<b>Objetivo Geral</b>	Proporcionar ao aluno uma compreensão dos principais conceitos de gerenciamento de projetos.
<b>Quantidade de alunos por disciplina</b>	Variando entre 15 e 30 alunos.
<b>Informações específicas da UI</b>	
<b>Conteúdo programático</b>	Iniciação e Motivação sobre Planejamento de Projetos de Software.

A faixa etária do público-alvo é geralmente entre 19 e 30 anos. Tipicamente, este tipo de público-alvo já possui contato com jogos em geral. Além disso, assume-se que o mesmo tenha recebido uma introdução com relação a parte de planejamento em gerenciamento de projetos antes de ser apresentado o jogo, na própria disciplina no qual o jogo deverá ser aplicado.

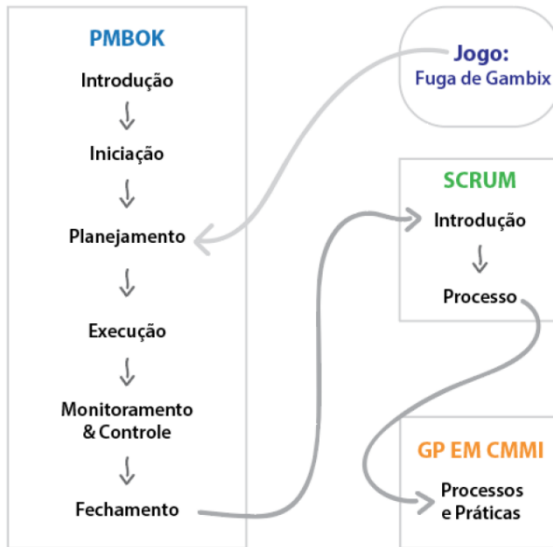
#### Caracterização do Público-Alvo do jogo.

<b>Caracterização dos Aprendizes</b>	
<b>Formação</b>	Cursando Bacharelado em Ciências da Computação, Sistemas de Informação ou algum outro curso na área de tecnologia.
<b>Faixa etária</b>	Grande maioria na faixa entre 19 e 30 anos.

<b>Gênero</b>	Majoritariamente masculino.
<b>Preferência de gêneros de jogo</b>	Ação-aventura, RPG e Estratégia.
<b>Preferência de plataformas de jogo</b>	Computador ou console.
<b>Modo de interação Preferido</b>	Singleplayer.
<b>Frequência de jogar jogos digitais</b>	Semanalmente: ao menos uma vez por semana.
<b>Frequência de jogar jogos não</b>	Raramente: de tempos em tempos.
<b>Cenários favoritos</b>	Medieval, Dungeons & Dragons, ficção científica.
<b>Persona</b>	Nome: Josh Idade: 23 Descrição: Estudante de Ciência da Computação, apesar de ter bastante tarefas para fazer sempre, quase todos os dias Josh consegue reservar um tempo para jogar online, e uma vez por semana ele se reúne com seus amigos para jogar Dungeon & Dragons.

O contexto organizacional onde o jogo é projetado para ser adotado são turmas de disciplinas de Gerenciamento de Projetos em cursos de graduação em computação em universidades públicas. A disciplina acontece tanto em sala de aula (sem computadores) como pontualmente em laboratórios (com computadores). As salas de aula possuem mesas planas e tanto os laboratórios como as salas de aula possuem Datashow. As 72 horas/aula da carga horária da disciplina são distribuídas em encontros com duração de 2 horas/aula. O jogo deve ser aplicável em um período de até 50 minutos (1 hora/aula), incluindo nesse tempo a organização dos grupos que jogarão bem como a finalização e recolhimento do material. As turmas de Gerenciamento de Projetos têm números variados entre 15 e 30 alunos, que devem ser divididos em grupos de 4-6 alunos por jogo.

O objetivo da unidade instrucional é complementar a unidade de Motivação da disciplina de Gerenciamento de Projetos. A sequência de apresentação de conteúdos pode ser observada.



## F2 – Projeto da Unidade Instrucional

A estratégia instrucional escolhida foi a de aprendizado por experiência, através de um jogo educacional. Serão abordados no jogo, como conteúdo a análise de objetivos e o planejamento de ações. Com base no objetivo a ser alcançado, deve-se realizar um planejamento, de modo a conseguir encontrar o melhor meio de alcançar sucesso (vencer).

Os alunos irão interagir principalmente competindo com os outros jogadores, mas realizando a jogada de forma individual, com *feedback* e troca de experiências com os outros participantes ao longo do jogo.

O objetivo de aprendizado do jogo desenvolvido não abrange nenhum aprendizado de conteúdo ou habilidade, mas uma mudança de atitude e em nível afetivo.

## F3 – Desenvolvimento do jogo educacional

### ***F3.1 – Análise do jogo***

Seguindo a atividade levantar requisitos do jogo do processo, são apresentados abaixo os requisitos do jogo:

Como requisito do jogo, foi definido que ele deve ser jogável entre 4-6 pessoas;

O jogo deve também ser aplicável em um tempo aproximado de 50 min, o equivalente há uma hora/aula em média;

Deve ser um jogo não-digital para possibilitar a jogada sem uso de computadores, estimulando também a interação social entre os alunos;

O custo por jogo não deve ser mais do que 150 reais;

Possuir em sua mecânica, regras ou elementos que façam com que quem planeje, chegue ao final com uma chance de vitória maior do que quem não planejou.

### ***F3.2– Concepção do jogo***

Nesta seção são apresentados os principais aspectos do jogo educacional desenvolvido. Na tabela a seguir são apresentadas as principais características do jogo.

Características do jogo.

<b>Características do jogo</b>	
<b>Objetivo do jogo</b>	O jogador vai escolher um personagem e deve completar a missão de coletar a quantidade de recursos atribuída a ele.
<b>Gênero</b>	Estratégia.
<b>Plataforma</b>	Jogo não-digital de tabuleiro.
<b>Modo de interação</b>	Multiplayers competitivo: vários jogadores competem entre si.
<b>Regras</b>	O Jogador se movimentara pelo tabuleiro coletando os recurso sindicados nas casas que ele ocupar. Um D20 será rolado a cada rodada, podendo seu valor modificar a maneira como ele coletara os recursos indicados, ou lhe conceder cartas.As cartas obtidas alteram opções de coleta ou movimentação,e valem apenas para a rodada que for utilizada. O jogo não conta tempo, mas rodadas. Ao final do jogo, ganha quem alcançar o objetivo antes de 14 rodadas. Caso isso não aconteça, um critério de desempate baseado na quantidade de recursos inúteis coletados definira um vencedor.
<b>Mecânica</b>	O jogador se movimentara de uma casa a outra seguindo as linhas que ligam as casas. Ele não pode ir e voltar pela mesma linha. A distância entre uma casa e outra que ele pode ir é de até três casas, cabendo a ele escolher qual dentre as alcançáveis lhe é mais proveitosa.
<b>Narrativa</b>	O jogo começa com cada jogador escolhendo uma carta que contém seu personagem e objetivo.

	<p>Em seguida, alguém lê para todos os jogadores a narrativa do jogo, que explica o que aconteceu e onde eles estão. Durante a narrativa, cada jogador é convidado a apresentar aos outros como chegou na situação atual, e qual seu objetivo no jogo. Concluída esta etapa de familiarização, o jogo é iniciado.</p> <p>A cada rodada, os jogadores escolhem dentre as casas alcançáveis, para qual delas eles desejam ir. Uma casa é dita alcançável, se ela estiver separada por até 3 casas de distância da casa atual do jogador, contando a partir das linhas que o jogador não utilizou para chegar na casa atual. Na casa escolhida se encontra a quantidade de recursos que devem ser coletados, e três números. O jogador usa então o D20, para ver como será feita a coleta dos recursos indicados na casa:</p> <p>Se o dado rolar um dos valores que se encontram na casa, ele recebe adicionalmente aos recursos, uma carta ação, cujo efeito pode ser aplicado a qualquer momento futuro, mediante descarte da carta. Se o valor rolado for 20, ele recebe o valor dos recursos descritos na casa em dobro. Se for 1, ele não recebe recurso nenhum, apenas um token de azar. Caso não se encaixe em nenhum dos três casos, o jogador recebe os recursos e um token de azar, que ao juntar três, podem ser trocados por uma carta bônus.</p> <p>O jogo tem uma duração de até 14 rodadas, podendo se encerrar antes caso alguém alcance seu objetivo. Quando um jogador alcançar o seu objetivo, a rodada vai até o final, para o caso de mais de um jogador alcançar na mesma rodada. Caso isso aconteça, ou no caso de o jogo alcançar 14 rodadas sem vencedor, será aplicada a regra de desempate para definir o vencedor.</p>
<b>Recursos do jogo</b>	<p>Pedra avermelhada, usada para comprar suprimentos.</p> <p>Pedra esverdeada, usada para comprar combustível.</p> <p>Pedra negra, usada para reparos na nave.</p>
<b>Personagens</b>	
<b>John</b>	<p>Pego no campo de asteroídes enquanto levava uma mensagem importante para o planeta Maelstrom, John e sua pequena tripulação foram completamente surpreendidos e não conseguiram evitar que sua super-rápida ZH4000 caísse. A nave sofreu bastantes danos, e perdeu todo o combustível.</p>
<b>Bruce</b>	<p>Enquanto Bruce e sua tripulação composta por funcionários da GalaTur levavam os relatórios da última temporada para sua sede no outro lado da galáxia, um campo de asteroides que não estava nos radares apareceu, forçando uma série de manobras evasivas que quase foram um sucesso. Por fim um asteroíde acertou o sistema de propulsão, obrigando a um pouso forçado no planeta mais próximo.</p>
<b>Srta. Gibbs</b>	<p>Transportando os novos trabalhadores de um projeto grandioso que está sendo feito no planeta Zed-23, Gibbs já estava bem próxima de seu destino quando sem aviso um asteroíde acertou em cheio a parte de traz da nave. O impacto inutilizou um dos motores, fazendo com que fosse necessário um pouso emergencial.</p>
<b>McNally</b>	<p>O conhecido pirata espacial McNally e seu bando viajavam com os escudos da nave ativos quando um pequeno asteroíde encontrou uma brecha e danificou o tanque de combustível. Voltando para seu esconderijo, foram surpreendidos no começo da viagem por um tanque vazio em um planeta desconhecido.</p>
<b>Theodore</b>	<p>A nave que transportava o empresário Theodore para uma reunião de negócios no sistema Gom foi pega completamente de surpresa pelo</p>

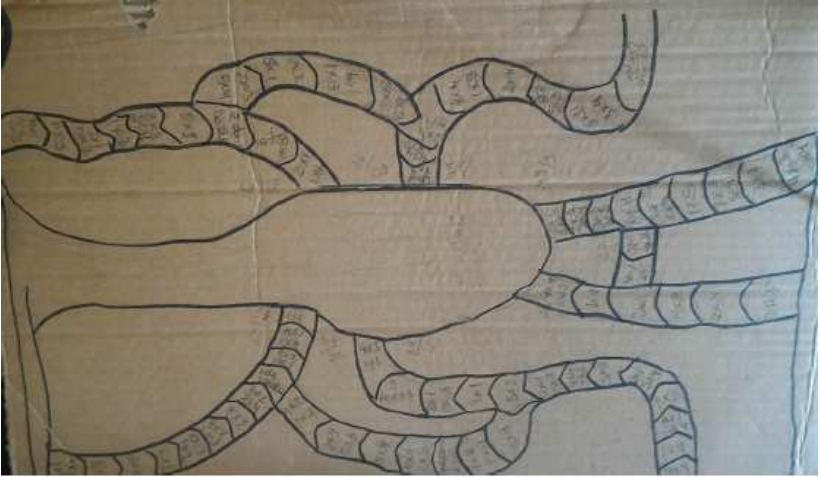
	campo de asteróides não mapeado. O impacto foi frontal, e destruir o sistema de navegação da nave, fazendo com que ela caísse, e Theodore se ferisse levemente.
<b>Capitão Bartholomeu</b>	Os passageiros da nave de transporte da interPlanet foram surpreendidos pela voz do seu capitão "Senhores passageiros, aqui é o Cap. Bartholomeu falando, informo que será necessário um pouso de emergência no planeta Gambix par afazer alguns reparos na nave e reabastece-la. A interPlanet pede desculpas pelo transtorno".
<b>Outros Elementos do jogo</b>	
<b>Tabuleiro</b>	Responsável visual na movimentação do jogo.
<b>Ficha de narrativa</b>	Contém a narrativa que será apresentada ao iniciar o jogo, inserindo os jogadores no contexto.
<b>Carta de objetivo</b>	Contém a descrição do personagem e seu objetivo.
<b>Carta de ação</b>	Obtidas ao rolar um dos números presentes na casa do tabuleiro, ao ser utilizada altera a mecânica do jogo naquela rodada.
<b>Carta bônus</b>	Obtidas ao acumular três tokens de azar, pode ser usada para adicionar seu valor ao dado.
<b>Token de azar</b>	Obtidos ao falhar em rolar um dos números presentes na casa.
<b>Peões</b>	Token de movimentação do jogador.
<b>Token de finalização</b>	Peças obtidas ao finalizar cada um dos objetivos, indicando que você conseguiu.
<b>Dado (D20)</b>	Dado de vinte faces, utilizado na coleta de recursos.
<b>Vitória e feedback educacional</b>	
<b>Critérios de vitória</b>	Será o vencedor do jogo aquele que em no máximo 14 rodadas alcançar o objetivo de seu personagem. Caso ninguém alcance o objetivo em 14 rodadas, ou mais de um jogador alcance na mesma rodada, será aplicado o critério de desempate. Desempate: O vencedor será aquele com menos recursos extras ao objetivo, pois este teve menos desperdício.
<b>Feedback educacionais ao jogador</b>	O jogo não fornece <i>feedback</i> ao jogador durante o jogo. O <i>feedback</i> ocorre ao final do jogo, em uma discussão entre jogadores e professor sobre o resultado do jogo. Pontos sugeridos a serem levantados no <i>feedback</i> incluem o quão longe cada um estava ou não de ganhar, e porque eles não conseguiram alcançar o objetivo.

### ***F3.3– Design do jogo***

Nesta seção é descrita a evolução do desenvolvimento do jogo até sua versão final. O jogo segue desde o início do seu desenvolvimento a ideia de que o jogador deve coletar recursos para vencer.

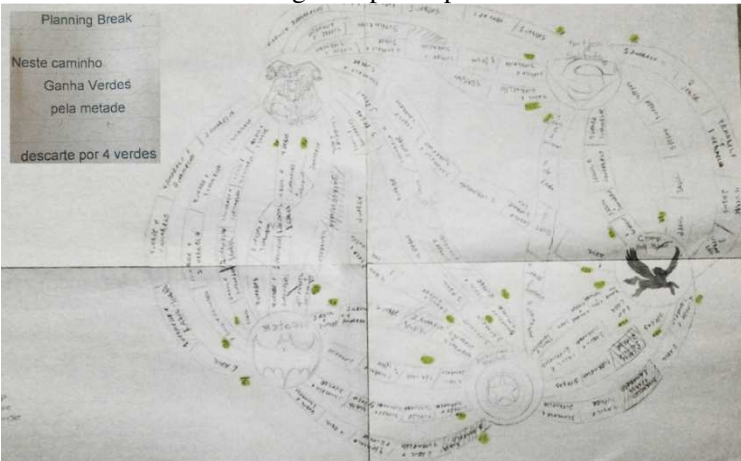
Na primeira versão, o jogo tinha a ideia de caminhos fixos, aonde o jogador ia caminhando, e a cada rodada coletava os recursos. Aplicações desta versão detectaram que esta possuía caminhos muito longos, que não ajudavam no planejamento, e era pouco divertido.

Primeiro protótipo.



Uma nova versão, ainda seguindo a ideia de ter caminhos, que uma vez escolhidos, percorria-se nele coletando recursos a cada rodada, e ao fim do caminho, escolhia-se o próximo.

### Segundo protótipo.

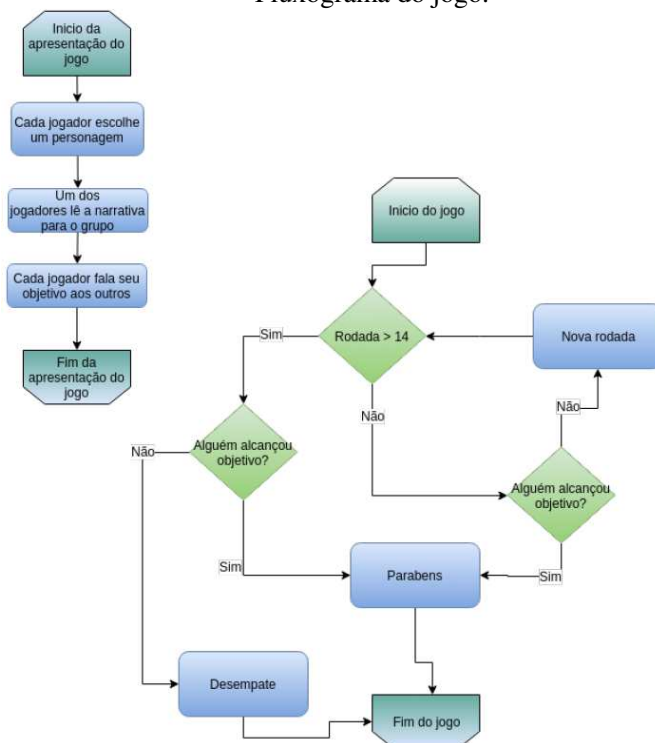




## Modelagem do jogo

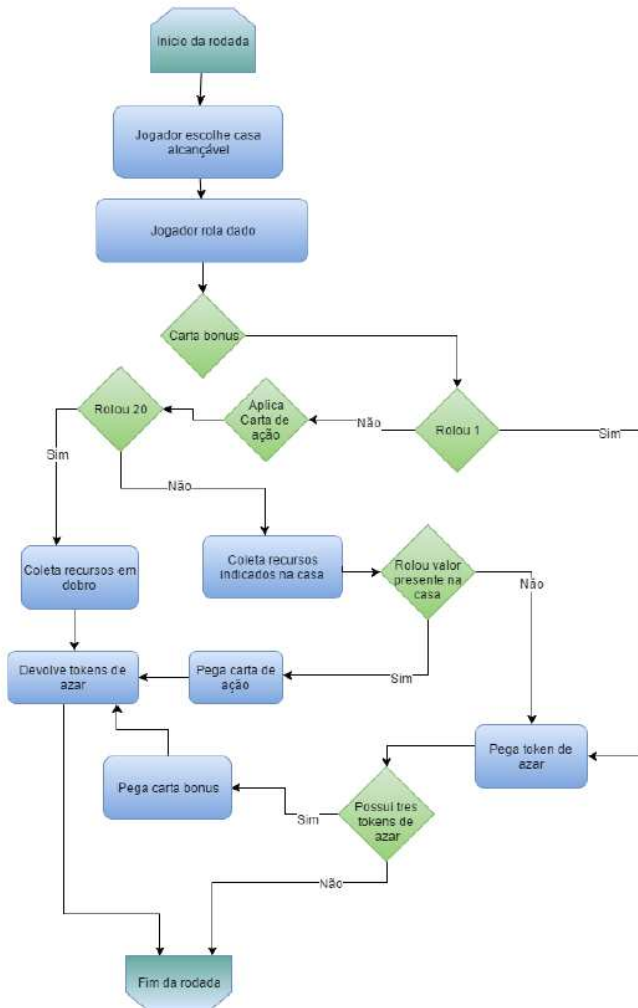
Com objetivo de modelar o jogo, apresenta-se o fluxograma que descreve as etapas de início do jogo e as iterações iniciais até o final do jogo.

Fluxograma do jogo.



A seguir apresenta-se o fluxo de uma rodada habitual, com suas variações, dependendo das ações tomadas pelo jogador, e do estado atual dos artefatos do jogador.







## Fluxograma de uma rodada do jogo.







### F3.4– Implementação do jogo

O jogo possui como elementos que devem ser comprados os peões e dado de 20 números. Os demais elementos (cartas e tabuleiro) devem ser impressos. O tabuleiro deve ser impresso em papel A3.

#### Elementos do jogo.

Elementos do jogo	
Peões	 <p>Peões comprados no site Alibaba (<a href="http://www.alibaba.com/">http://www.alibaba.com/</a>)</p>
Tabuleiro	
Carta de objetivo	<p>A nave que transportava o empresário Theodore para uma reunião de negócios no sistema Gom foi pega completamente de surpresa pelo campo de asteroides não mapeado. O impacto foi frontal, e destruiu o sistema de navegação da nave, fazendo com que ela caísse, e Theodore se ferisse levemente.</p> <p>  1800          800          1400       </p>
Carta de ação	<p><b>DOIS POR UM</b></p> <p>  <b>Colete o dobro do recurso</b> </p>

Carta bônus		
Token de azar		
Token de finalização		
Dado (D20)	 Peões comprados no site Alibaba ( <a href="http://www.alibaba.com/">http://www.alibaba.com/</a> )	

### *F3.5– Teste do jogo*

Os testes do jogo foram realizados com os protótipos e a versão final com alunos de graduação e pós-graduação de computação da UFSC.

## **F4 – Execução da Unidade Instrucional**

A execução do jogo foi realizada em maio de 2016, com um grupo de cinco participantes. Dos cinco participantes, dois são doutorandos, um mestre e dois graduandos, e todos já cursaram a disciplina de gerenciamento de projetos oferecida pelo Departamento de Informática e Estatística da Universidade Federal de Santa Catarina (UFSC).

A aplicação do jogo durou cerca de uma hora, incluindo a montagem do ambiente, a explicação das regras e execução do jogo, e a organização dos materiais após a aplicação.

No início da aplicação os participantes foram convidados a assinar TCLE e a autorização do uso da imagem. Ao final do jogo, os participantes foram convidados a responder o questionário sobre a experiência vivenciada por eles, disponibilizado impresso.

### Aplicação do jogo.

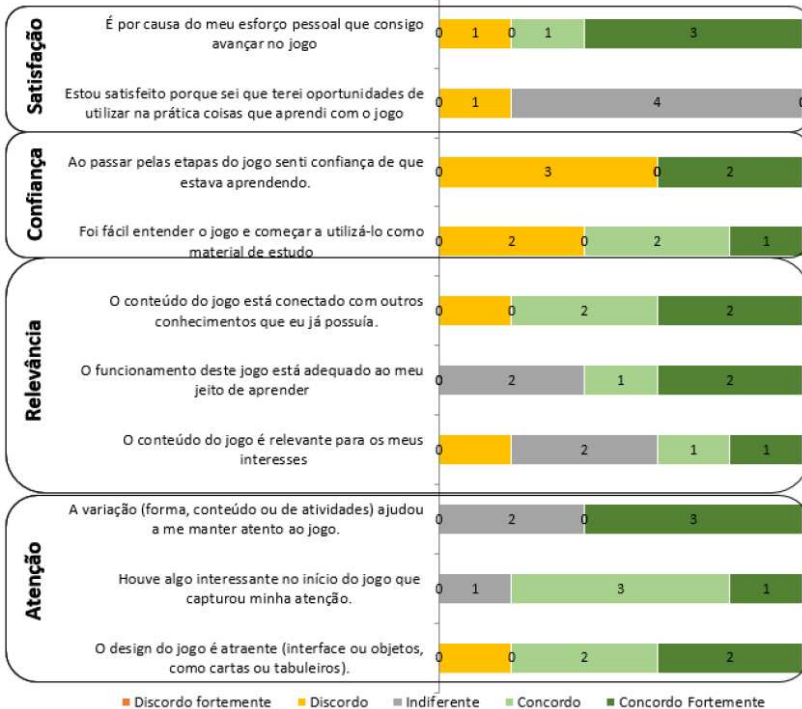


### **F6 – Avaliação da Unidade Instrucional**

Os dados coletados com os questionários foram compilados para a planilha fornecida pelo modelo MEEGA (SAVI; WANGENHEIM; BORGATTO, 2011) utilizado na avaliação do jogo. Com os dados coletados foram gerados gráficos, que permitem uma melhor avaliação quanto aos aspectos positivos e negativos dos componentes avaliados pelo modelo.

No gráfico observa-se os resultados obtidos para a motivação. Nele é possível observar que a maioria dos pontos avaliados tiveram resultados positivos, com pelo menos 3 das 5 das avaliações representando concordância com os pontos avaliados.

## Avaliação da Motivação.



O quesito atenção foi o melhor avaliado, tendo apenas uma das avaliações mostra a discordância na questão "*O design do jogo é atraente*". Nas questões quanto a mecânica e iteração do jogo não houve avaliações apresentando discordância.

O quesito relevância também foi bem avaliado, contando com apenas duas respostas negativas.

A dimensão de confiança apresentou o maior número de avaliações negativas, sendo três delas na questão "*Ao passar pelas etapas do jogo senti confiança de que estava aprendendo*", mas isto já era esperado, pelo objetivo de aprendizado não ser cognitivo, e sim afetivo e não haver um aprendizado visível de maneira concreta.

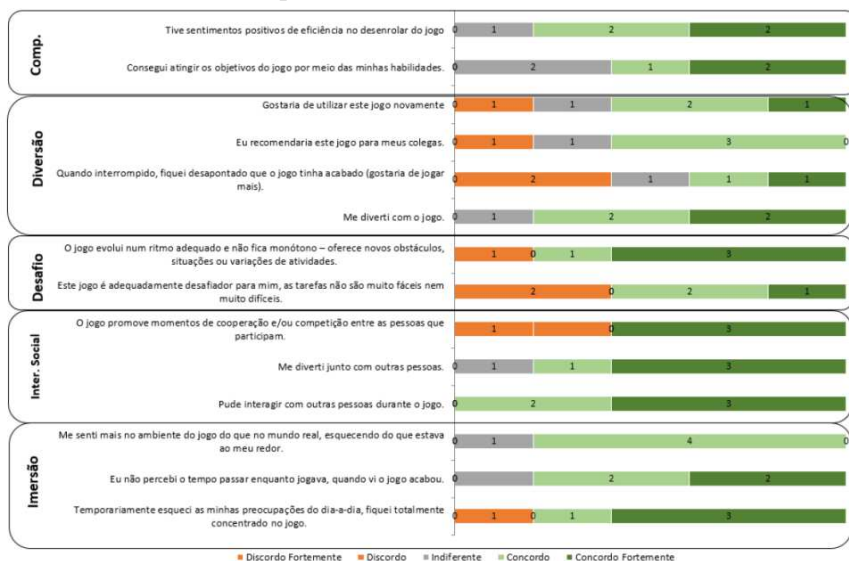
Em termos de experiência de usuário, no gráfico a seguir apresenta-se os resultados obtidos, sendo observado novamente uma avaliação positiva.

Na dimensão desafio houveram algumas discordâncias, corroboradas pelos comentários sobre possíveis melhorias, que dizem que o jogo poderia ser um pouco mais difícil.

Podemos destacar a dimensão de imersão, que teve apenas uma resposta indicando discordância, e quatro das avaliações em cada uma das questões indicando concordância.

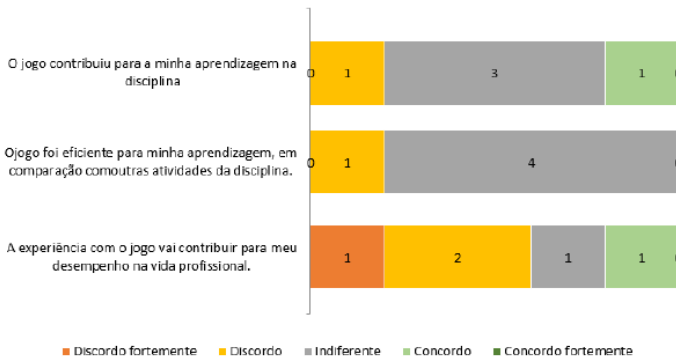
A dimensão de competência não apresentou avaliações com discordância, apenas concordância e algumas respostas neutras.

### Avaliação da Experiência do Usuário.



A seguir apresenta-se o gráfico com os resultados da aprendizagem. Ao analisar o gráfico de aprendizado, observa-se que as respostas ficaram bastante neutras. Isto se deve ao fato de não haver no objetivo de aprendizado um conteúdo a ser aprendido, mas apenas um conceito a ser valorizado.

### Avaliação da Aprendizagem.

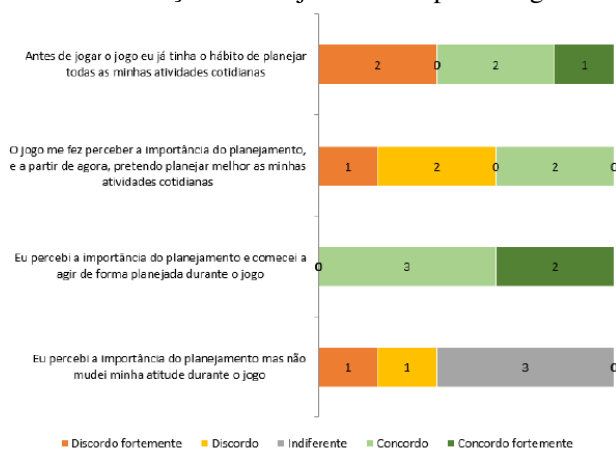


Para o objetivo de aprendizagem, foi avaliado as últimas quatro questões apresentadas, que fazem parte da adaptação do modelo para avaliar o impacto afetivo segundo a taxonomia de Bloom (1956).

As questões foram apresentadas de maneira a avaliar o hábito de planejamento antes do jogo, e avaliar o quão profundamente nos níveis afetivos os participantes seriam atingidos. A primeira das quatro questões visa mensurar o planejamento no cotidiano dos participantes antes do jogo. As três seguintes, cada uma delas mensura um dos três primeiros níveis afetivos avaliados. A segunda questão com valores mais positivos indica que o participante percebeu a importância, mas não houve uma mudança de postura, ele apenas participa de maneira passiva. A terceira questão com valores positivos indica que o aluno não apenas recebe, mas começa a responder ao que está sendo passado, já demonstrando certa mudança de postura. E por fim a última questão corresponde a um nível de valorização mais profunda, aonde o aluno leva o conceito adiante, como um bom hábito a ser praticado diariamente.



## Avaliação dos Objetivos de Aprendizagem.



## APÊNDICE J – FASES E ATIVIDADES DA PRIMEIRA VERSÃO DO PROCESSO ENgAGED

A seguir são apresentadas as fases e atividades em alto nível da primeira versão do processo ENgAGED.

<b>ENgAGED: Processo de desenvolvimento de jogos educacionais</b>	
<b>F1. Análise da unidade instrucional</b>	A1.1 Identificar metas instrucionais A1.2 Analisar aprendizes e contexto A1.3 Conduzir a análise instrucional A1.4 Definir objetivos de desempenhos
<b>F2. Projeto da unidade instrucional</b>	A2.1 Desenvolver instrumento de avaliação A2.2 Desenvolver estratégias instrucionais
<b>F3. Desenvolvimento do jogo educacional</b>	<p><b>F3.1 Análise</b> A3.1.1 Levantar requisitos do jogo A3.1.2 Definir objetivos do jogo A3.1.3 Definir idéias do jogo A3.1.4 Definir conceitos para alcançar os objetivos A3.1.5 Definir linguagem de programação ou <i>game engine</i> A3.1.6 Testar os conceitos com potenciais jogadores</p> <p><b>F3.2 Concepção</b> A3.2.1 Desenhar esboço do jogo A3.2.2 Definir interação entre jogadores A3.2.3 Definir regras do jogo A3.2.4 Definir níveis do jogo A3.2.5 Definir desafios do jogo A3.2.6 Refinar protótipo</p> <p><b>F3.3 Design</b> A3.3.1 Criar <i>storyboards</i> A3.3.2 Criar interfaces A3.3.3 Criar animações A3.3.4 Produzir/editar áudio</p> <p><b>F3.4 Implementação</b> A3.4.1 Desenvolver aplicação A3.4.1.1 Codificar <i>storyboard</i> e animações A3.4.2 Criar integração tecnológica</p> <p><b>F3.5 Teste</b> A3.5.1 Realizar testes do jogo</p> <p><b>F3.6 Implantação</b> A3.6.1 Configurar e instalar jogo A3.6.2 Publicar jogo</p>

<b>F4. Execução da unidade instrucional</b>	A4.1 Planejar a execução do jogo A4.2 Executar o jogo na unidade instrucional
<b>F5. Avaliação da unidade instrucional</b>	A5.1 Definir avaliação A5.2 Planejar avaliação A5.3 Executar avaliação e coleta de dados A5.4 Analisar e interpretar dados