

## PROPAGA: Processo Padrão de Apoio a Games Educativos

Kelven Siqueira, Carlos Portela

Faculdade de Sistemas de Informação

Universidade Federal do Pará

Cametá-PA, Brasil

kelvenchristian18@hotmail.com, csp@ufpa.br

Sandro Oliveira

Programa de Pós-Graduação em Ciência da Computação

Universidade Federal do Pará

Belém-PA, Brasil

srbo@ufpa.br

**Resumo**—Jogos educativos podem ser ferramentas eficientes no processo de ensino-aprendizagem, sendo utilizados nas mais diversas áreas de ensino, como Matemática, Português, Geografia e Ciências Exatas. Contudo, muitos desenvolvedores desconhecem as atividades necessárias para criar um jogo educativo. É comum encontrar relatos de jogos que são criados de forma *ad-hoc*, não seguindo processos formais de desenvolvimento de software. Neste contexto, este artigo apresenta um processo padrão para apoiar o desenvolvimento de jogos educativos. Esse processo, denominado PROPAGA, foi avaliado por um painel de especialistas com experiência em desenvolvimento de jogos. Espera-se que o seguimento das fases deste processo incentive a criação e a adoção de jogos que possam motivar e engajar os alunos no processo de ensino-aprendizagem.

**Palavras-chave:** *processo padrão; modelagem de processos; jogos educativos.*

### I. INTRODUÇÃO

No Brasil, jogos educativos são utilizados nas mais diversas áreas de ensino, como por exemplo, na Matemática, Português, Geografia, Ciências Exatas e Humanas [1]. O desenvolvimento desse tipo de jogos totalizou quase 50% de todos os títulos produzidos pelas empresas brasileiras no ano de 2014, segundo o I Censo da Indústria Brasileira de Jogos Digitais [2].

Jogos educativos se caracterizam como uma abordagem ativa de ensino-aprendizagem que permite o engajamento dos alunos na busca pelo conhecimento [3]. Consequentemente, a adoção desses jogos em sala de aula pode possibilitar a melhoria no aprendizado dos alunos, pois possibilita experiências imersivas dos conteúdos vistos em sala de aula.

Contudo, apesar do seu potencial, o desenvolvimento de jogos voltados para o ensino ainda é pouco atrativo para os desenvolvedores no Brasil. A falta de interesse nessa área, talvez, deve-se ao fato de que desenvolver um jogo educativo é bastante complexo, exigindo muito esforço e pouco retorno financeiro, que é maior para outras áreas de jogos digitais, como por exemplo, a de entretenimento [4].

O processo de desenvolvimento de um jogo digital assemelha-se ao processo de Engenharia de Software, pois é constituído de etapas como concepção do jogo, ideia do jogo e sua narrativa, documentação do design do jogo, escolha da plataforma e tecnologias de apoio, codificação e manutenção [5]. Contudo, o desenvolvimento de jogos educativos possui as suas particularidades. Apesar disto, é comum encontrar na literatura relatos de jogos que são desenvolvidos de forma *ad-hoc*, não seguindo processos formais de desenvolvimento de software [6].

Neste contexto, este artigo apresenta o PROPAGA (acrônimo de PROcesso Padrão de Apoio a GAMES educativos). Este processo objetiva que os desenvolvedores possam conhecer melhor as fases e tarefas envolvidas na criação de jogos e, consequentemente, fomentar a adoção de jogos educativos no Brasil.

Além desta seção introdutória, a Seção II apresenta os conceitos e características de Jogos Educativos considerados nesta pesquisa. A Seção III descreve as principais definições de Processos de Desenvolvimento, bem como apresenta os processos para desenvolvimento de jogos educativos identificados nesta pesquisa. Adicionalmente, apresenta-se uma análise comparativa entre as fases em comum desses processos. Na Seção IV, é apresentado o processo padrão PROPAGA, a principal contribuição desta pesquisa. Essa proposta foi avaliada através de um painel composto por 7 (sete) especialistas, conforme destaca a Seção V. Por fim, as conclusões e trabalhos futuros são apresentados na Seção VI.

### II. JOGOS EDUCATIVOS

Um jogo poder ser conceituado como uma atividade orientada, dentro de um contexto de realidade imaginada, onde os participantes tentam alcançar uma ou várias metas, guiando-se através de regras pré-estabelecidas [7]. Nesse sentido, um jogo educativo pode ser definido como uma situação que exige ações coordenadas, as quais objetivam aquisição de conhecimento ou de habilidades intelectuais adquiridas didaticamente [8].

Através dessa tecnologia, os alunos podem relacionar a teoria abordada a sua aplicação prática. Adicionalmente, o uso de jogos em sala de aula pode motivar e facilitar o aprendizado desses alunos, além de engajá-los nas atividades, tornando-os agentes ativos no processo de ensino-aprendizagem [9]. Neste sentido, os jogos educativos proporcionam aos professores métodos dinamizados de estimular os estudantes no que diz respeito à aprendizagem.

Até mesmo quando implementados para resolver tarefas simples, os jogos educativos podem ser plataformas de aprendizagem bastante ricas e desafiadoras, pois fornecem um mundo imaginário para ser explorado pelos alunos [10]. De maneira complementar, observa-se que jogos são cada vez mais utilizados como uma maneira de estimular os discentes, no que se refere à busca pelo conhecimento, uma vez que possui elementos motivadores [3].

No entanto, o desenvolvimento de jogos educativos envolve diversas tarefas técnicas específicas da área de educação. Estas tarefas, muitas vezes, não são seguidas

pela equipe de desenvolvimento. Assim, se faz necessário que estes desenvolvedores possuam uma melhor base instrucional para assegurar que os jogos atinjam seus objetivos de aprendizagem.

Para tratar este problema, foram encontrados na literatura diversos trabalhos que tratam de processos de produção de jogos educativos [6][11][12]. Cada um destes apresenta um processo diferente, conforme apresenta a Seção III, com fases e tarefas diversas, ainda que todos estejam inseridos no mesmo contexto.

Além disso, esses processos são descritos apenas de forma textual, não apresentando uma modelagem visual e formal que permita um melhor entendimento dos fluxos de tarefas de desenvolvimento. Portanto, essa variedade de processos e fases acaba exigindo uma análise detalhada para que se entenda a visão de diversos autores em relação ao desenvolvimento de jogos educativos como ferramentas que possam auxiliar o processo de ensino-aprendizagem.

### III. PROCESSO DE DESENVOLVIMENTO

#### A. Definição de Processo

Humphrey [13] define o processo de desenvolvimento de software como a transformação de um conjunto de requisitos em um produto de software. Nesse sentido, cada etapa dessa transformação é considerada um refinamento do nível de abstração da solução de um problema.

Cada uma dessas etapas é composta por um conjunto parcialmente ordenado de atividades que interagem entre si e que podem estar associadas a elementos do processo, como artefatos, recursos humanos ou ferramentas [14]. Apesar das particularidades de um projeto de software implicarem em processos distintos, existe um conjunto de elementos comuns presentes em todos os processos [13]. Esse conjunto de elementos é denominado “processo padrão”, pois consiste em um processo fundamental que permite definir e instanciar os demais processos [15].

A partir do seguimento de um processo padrão, é possível [13]:

- Reduzir os problemas relacionados a treinamento e suporte da equipe em relação ao processo;
- Melhorar os processos instanciados a partir das experiências adquiridas reunidas no processo padrão;
- Reduzir o tempo e esforço gasto na definição de novos processos.

A fim de obter essas vantagens, definiu-se um processo padrão para o desenvolvimento de jogos educativos. Inicialmente, realizou-se uma revisão da literatura em teses de doutorado, dissertações de mestrado e artigos científicos. Desta forma, identificaram-se propostas de processos nos trabalhos de Battistella [6], Dambros [11] e Gonçalves e Werner [12]. No entanto, observou-se que esses trabalhos não apresentaram a modelagem de seus processos, o que dificulta a compreensão e aplicação dos mesmos [13].

A subseção a seguir define os objetivos e os elementos necessários para realizar a modelagem de um processo de software específico.

#### B. Modelagem de Processo

A modelagem de processos objetiva o registro, entendimento, avaliação e comunicação de um processo de

software [17]. De maneira específica, a modelagem de processos de software apresenta quatro benefícios [18]:

- A possibilidade de divulgar o processo de software;
- A facilidade para o reuso do processo de software;
- O suporte à evolução do processo de software;
- O apoio ao gerenciamento do processo de software.







O modelo de processo permite uma representação única do processo de software que pode ser compartilhada entre os todos os indivíduos envolvidos nesse processo, como clientes, gerentes, usuários, etc. Um modelo de processo também pode ser utilizado para o compartilhamento de conhecimento e transferência de experiências [17].

A modelagem facilita o reuso do processo, pois, a partir de um modelo, diversos processos de software podem ser instanciados, necessitando apenas de modificações nas estruturas que são únicas em cada processo e reaproveitando toda a estrutura comum compartilhada pelos processos [18].

Buscando obter estes benefícios, optou-se por modelar as fases dos processos identificados na literatura seguindo a linguagem de modelagem SPIDER\_ML [18] e o apoio ferramental da Spider-PM [19]. Essa linguagem trata-se de um perfil do padrão *Software Process Engineering Metamodel* (SPEM) [20], que possui notações formais específicas para representar processos de software.







A Tabela I apresenta os elementos utilizados na modelagem, que representam um processo padrão, com suas respectivas notações, nomes e a descrição de cada elemento.

TABELA I. ELEMENTOS DO PROCESSO PADRÃO [18]

Notação	Nome	Descrição
	<i>Processo Padrão</i>	O conjunto de todos os ativos do processo de software que são reutilizáveis.
	<i>Disciplina</i>	Um conjunto de definições de tarefa que contribuem para o atendimento de um mesmo objetivo do Processo Padrão.
	<i>Definição de Tarefa</i>	Um trabalho reutilizável e que não pode ser decomposto.
	<i>Definição de Papel</i>	Um conjunto de capacidades desejadas para um determinado papel que pode ser reutilizado em diferentes Processos Instanciados.
	<i>Definição de Produto de Trabalho</i>	Um conjunto de características desejadas para um produto de trabalho que pode ser reutilizado em diferentes Processos Instanciados.
	<i>Definição de Ferramenta</i>	Um conjunto de características desejadas para uma ferramenta que pode ser reutilizada em diferentes Processos Instanciados.

De forma similar, a Tabela II apresenta os elementos que representam um processo instanciado, ou seja, um processo padrão aplicado em um projeto específico.

TABELA II. ELEMENTOS DO PROCESSO INSTANCIADO [18]

Notação	Nome	Descrição
	<b>Processo Instanciado</b>	Uma instância do Processo Padrão para um projeto específico.
	<b>Fase</b>	Um período significativo de um Processo Instanciado.
	<b>Tarefa Instanciada</b>	Um trabalho que deve ser realizado no decorrer de um Processo Instanciado e que não pode ser decomposto.
	<b>Papel Instanciado</b>	Um papel desempenhado por um recurso no Processo Instanciado.
	<b>Produto de Trabalho Instanciado</b>	Um produto de trabalho consumido, gerado ou modificado em um Processo Instanciado.
	<b>Ferramenta Instanciada</b>	Uma ferramenta que deve ser utilizada durante a realização de uma tarefa em um Processo Instanciado.

Além dos elementos apresentados nas Tabelas I e II, a linguagem SPIDER\_ML adota 7 relacionamentos da *Unified Modeling Language* (UML) [18]: associação, transição, dependência, agregação, composição, generalização e âncora de nota. A semântica de cada um desses permaneceu a mesma da UML.

As subseções a seguir detalham os 3 (três) processos selecionados [6][11][12] e apresentam a sua modelagem. Após a modelagem, foi verificada a existência de processos que possuem fases em comum. A partir dessas, foi realizada uma análise comparativa na Subseção III-F.

C. *Processo de Battistella*

Battistella [6] define o *Educational Games Development* (ENgAGED) que reuni processos de design instrucional, além de processos de design de jogos, através de análises e comparações sistemáticas. Adotou-se este trabalho como base comparativa com os demais estudos por se tratar de uma tese de doutorado, o que confere maior embasamento científico aos resultados da pesquisa.

Através da análise desse processo, identificaram-se as fases que compõem a produção de um jogo educativo, as quais abrangem o processo como um todo, desde seu planejamento inicial até a sua utilização pelos usuários finais. O processo possui um total de 5 (cinco) fases, conforme destaca a Figura 1.

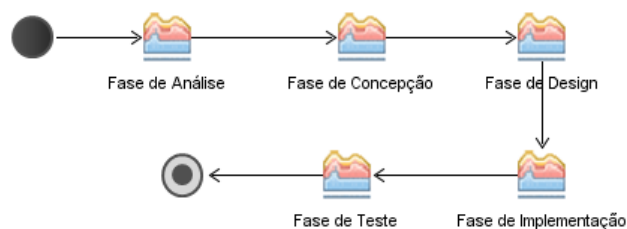


Figura 1. Fases do Processo de Battistella [6].

A Fase de Análise objetiva levantar requisitos, uma vez que é necessário tratar possíveis problemas em relação aos alunos que futuramente irão utilizar o jogo. Já a Fase de

Concepção define as partes constituintes do planejamento de um jogo, pois é nessa fase que se realiza a definição da história e objetivo do *game*, além da plataforma e gênero. Define-se, também, o modo de interação dos jogadores e personagens, bem como os diálogos, artefatos, cenários, menus de interação, formas de *feedback* educacional com o usuário, regras e critérios para se vencer.

Em seguida, a Fase de *Design* diz respeito à área visual (desenho de elementos do jogo e/ou modelagem 3D), mais especificamente a interface do jogo, além da definição das linguagens de programação que serão usadas, do *game engine* e seus componentes, da arquitetura do jogo e do próprio processo de desenvolvimento. Adicionalmente, a Fase de Implementação trata da codificação do jogo.

Por fim, o autor recomenda a realização da Fase de Teste, na qual são realizadas as avaliações do jogo. Os testes são necessários para se encontrar e corrigir possíveis *bugs*, além de analisar o envolvimento dos alunos na utilização da ferramenta de ensino (jogo educativo).

D. *Processo de Gonçalves e Werner*

Goncalves e Werner [12] descrevem noções básicas e conceitos utilizados no desenvolvimento de *games* tecnológicos e como esses podem ser utilizados em âmbito educacional. Os autores apresentam uma significativa contribuição, pois destacam os conceitos de desenvolvimento desta área e aplicam tais conceitos na criação de um protótipo. O protótipo encontra-se em evolução, e faz uso de métodos e ferramentas conhecidas, livres e genéricas. Desta forma, os autores demonstram a própria eficácia de uso do seu processo.

No entanto, encontrou-se dificuldade na identificação das fases do processo, pois as mesmas não foram explicitadas no artigo, apenas indiretamente demonstradas. Sendo assim, analisou-se e modelou-se 6 (seis) fases, conforme ilustrado na Figura 2.

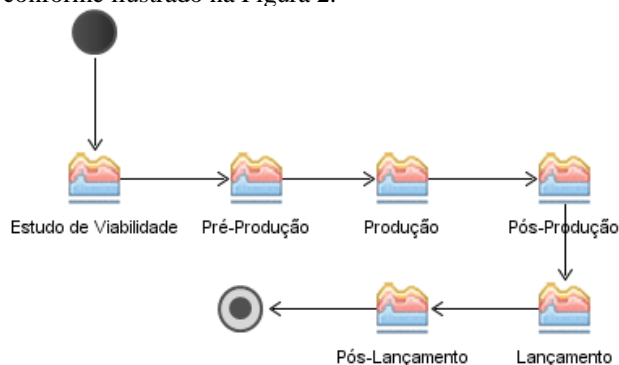


Figura 2. Fases do Processo de Gonçalves e Werner [12].

A fase de Estudo de Viabilidade trata dos métodos e ferramentas que podem ser utilizadas para que se chegue ao resultado desejado, tendo em vista até mesmo os custos para produção. Além da análise da viabilidade técnica (que está relacionada às ferramentas e tecnologias), existe ainda a análise de viabilidade operacional que visa a formação da equipe, contemplando suas habilidades e competências, bem como a análise de viabilidade econômica e a análise do cronograma a ser seguido.

Posteriormente, a fase de Pré-Produção está relacionada à alocação de recursos. É fundamental estabelecer os requisitos de *hardware* e *software*

necessários para a criação do jogo, alocação de recursos humanos, assinatura de termos de confidencialidade e não divulgação, elaboração do *Game Design Document* (GDD) e a elaboração do Documento de *Design Técnico*.

Em seguida, na fase de Produção analisa-se o desempenho do jogo, a procura de *bugs* a fim de evitar que o público-alvo tenha problemas com o uso do *game*.

Adicionalmente, na Fase de Lançamento, o jogo pode ser entregue ao cliente e, também, validado pelo mesmo. Finalmente, ocorre a fase de Pós-Lançamento, onde deve-se monitorar as reações dos usuários finais ao jogo, através do *feedback* dos próprios usuários.

**E. Processo de Dambros**

Dambros [11] propõe, em sua dissertação de mestrado, um jogo digital para auxiliar o ensino de cartografia no nível fundamental. A autora descreve o processo de desenvolvimento adotado de forma bastante detalhada, facilitando a identificação e entendimento das fases. Neste sentido, listam-se as 4 (quatro) fases que foram seguidas em sua experiência prática, conforme destaca a Figura 3.

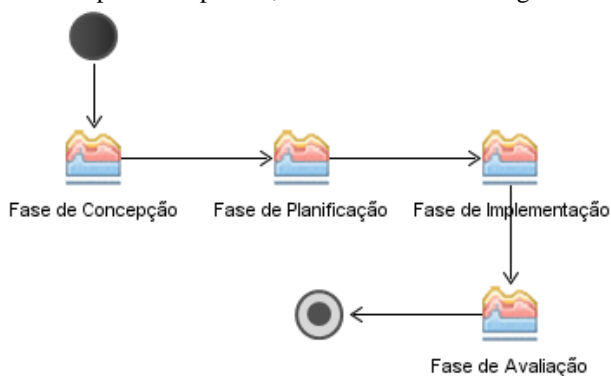


Figura 3. Fases do Processo de Dambros [11].

A Fase de Concepção define os objetivos, o público-alvo e a proposta pedagógica a ser seguida. Assim, determina-se, através desse planejamento, os componentes primários de um jogo, tais como, a definição do tema, formação da equipe e a delimitação dos conteúdos.

Posteriormente, a fase denominada de Planificação determina a pesquisa dos conteúdos que servirão como base para a construção do jogo. Além disso, sugere a criação de um *storyboards* como parâmetro para demonstrar aos usuários finais e equipe de desenvolvimento como de fato o jogo será. É nessa fase que se realiza a criação do *design* de interface, que trata do visual e da interação entre usuário e aplicação [11].

Já na Fase de Implementação, realiza-se a programação do jogo, onde cada membro da equipe exerce uma função específica. A equipe define a linguagem de programação a ser utilizada. É nessa fase que se inicia a criação de um protótipo para testar o funcionamento do *game*.

Por fim, a Fase de Avaliação diz respeito aos testes de validação do jogo educativo, com os alunos, para verificação de problemas de adequação com a aplicação. Portanto, nessa fase deve-se identificar e corrigir possíveis erros, bem como avaliar se o *game* atingiu o objetivo de contribuir com o processo de aprendizagem.

**F. Análise Comparativa**

Diante da análise dos processos apresentados em [6][11][12], foram encontradas fases com objetivos semelhantes, como Planejamento, Produção, Testes, entre outras. A seguir, são apresentados os resultados das análises comparativas entre os 3 (três) trabalhos selecionados, bem como as suas justificativas. Esta análise foi apresentada e discutida inicialmente em [13].

A Tabela III apresenta o resultado da análise comparativa entre as fases similares apresentadas por Battistella [6] e Gonçalves e Werner [12].

TABELA III. COMPARAÇÃO ENTRE OS PROCESSOS DE BATTISTELLA E GONÇALVES E WERNER

Battistella [6]	Gonçalves e Werner [12]	Justificativa
<i>Fase de Design</i>	<i>Pré-Produção</i>	A Fase de <i>Design</i> é similar à Pré-Produção visto que em ambas são estabelecidas as linguagens de programação e o processo de desenvolvimento do jogo. Essas informações devem constar no <i>Game Design Document</i> (GDD).
<i>Fase de Implementação</i>	<i>Produção</i>	Ambas as fases implicam em realizar a codificação, isto é, a tarefa de programar utilizando recursos de multimídia (áudio, vídeo e imagens) para gerar jogos interativos.
<i>Fase de Teste</i>	<i>Pós-Produção</i>	Na Fase de Teste são realizadas avaliações do jogo com o propósito de analisar e corrigir possíveis <i>bugs</i> , antes de seguirem para a Fase de Lançamento, de forma similar à Pós-Produção.

A Tabela IV apresenta o resultado da análise comparativa entre as fases similares apresentadas por Battistella [6] e Dambros [11].

TABELA IV. COMPARAÇÃO ENTRE OS PROCESSOS DE BATTISTELLA E DAMBROS

Battistella [6]	Dambros [11]	Justificativa
<i>Fase de Implementação</i>	<i>Fase de Implementação</i>	Em ambas as fases ocorre a codificação do jogo.
<i>Fase de Design</i>	<i>Fase de Planificação</i>	Nestas fases procura-se definir os componentes essenciais para a criação do jogo, o que envolve a escolha de um processo a ser seguido, definição das ferramentas e a criação do design de interface.
<i>Fase de Teste</i>	<i>Fase de Avaliação</i>	Analisar os elementos do jogo como jogabilidade, termos técnicos e adequação ao público-alvo, como descrito anteriormente, são atividades pertencente à etapa de Teste. Portanto, esse objetivo é similar a Fase de Avaliação.

#### IV. PROPAGA

Inicialmente, a proposta de processo padrão foi publicada em [21]. Posteriormente, essa proposta foi publicada em <http://lafoca.com.br/propaga/>, através de uma interface em HTML e CSS que permite aos desenvolvedores acessar e instanciar esse na criação dos seus jogos educativos. O PROPAGA unifica as características em comum das fases apresentadas em cada processo descrito na Seção III com o propósito de facilitar a compreensão das etapas de desenvolvimento de um jogo educativo.

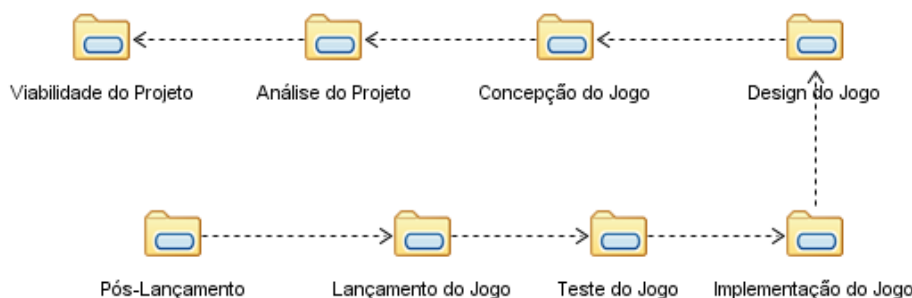


Figura 4. Disciplinas do PROPAGA.

Algumas dessas disciplinas possuem relação de dependência entre si, conforme mostra a Figura 4 através do relacionamento da UML, ou seja, uma é pré-requisito para execução da outra. Exemplo: Antes de realizar os testes do jogo, é necessário realizar a implementação deste.

Inicialmente, a disciplina Viabilidade do Projeto agrupa as tarefas de análise da viabilidade técnica, operacional, econômica e de prazo. Em seguida, a disciplina Análise do Projeto agrupa as tarefas de definição dos requisitos funcionais e não-funcionais.

Após o seguimento dessas tarefas, recomenda-se realizar as tarefas da disciplina Concepção do Jogo, que definem a proposta pedagógica, objetivos e metas de aprendizagem, tema e história do jogo, gênero e plataforma, modos de interação, artefatos e cenários, para conceber os elementos principais dos jogos educativos. Já a disciplina Design do Jogo aborda as tarefas de criação das artes e modelagem (2D ou 3D), bem como definição da linguagem de programação e *game engine*, além da alocação de recursos de *hardware*, software e humanos.

A disciplina Implementação do Jogo trata das tarefas de codificação dos módulos do jogo, assim como a criação de protótipos para testes com os usuários. Esses testes são contemplados na disciplina Teste do Jogo que objetiva avaliar a adequação do jogo ao público-alvo e analisar se os objetivos de aprendizagem foram alcançados.

Finalmente, a disciplina Lançamento do Jogo possui as tarefas que visam apresentar o resultado final do projeto ao público-alvo, sendo seguida da disciplina Pós-Lançamento que consiste no monitoramento das reações dos alunos e professores ao uso do jogo educativo.

As fases do processo instanciado são derivadas da instanciação dessas disciplinas para um projeto específico, conforme mostra a Figura 5.

A nomenclatura dessas fases foi minimamente alterada para representar a junção das fases equivalentes nas propostas analisadas. As tarefas que compõem cada uma dessas fases são descritas nas subseções a seguir.

Nas subseções a seguir, são apresentadas as disciplinas e definições de tarefas do PROPAGA.

##### A. Fases do Processo

O processo padrão é composto por disciplinas, que definem as tarefas que contribuem para o atendimento de um mesmo objetivo. Assim, as disciplinas que compõem o processo padrão proposto nesta pesquisa são apresentadas na Figura 4.

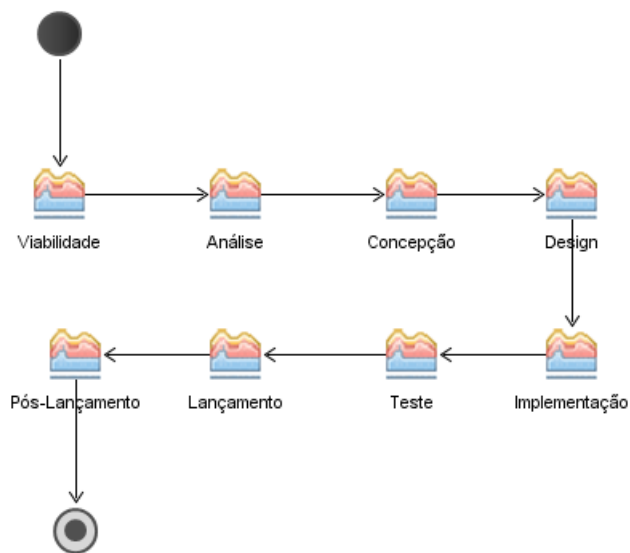


Figura 5. Fases do Processo Instanciado.

##### B. Tarefas de Desenvolvimento

As tarefas do processo padrão não podem ser decompostas e são reutilizáveis [18]. Neste sentido, para cada uma das disciplinas do processo padrão proposto, identificaram-se diversas tarefas (T) necessárias para que o modelo seja utilizado de maneira eficiente no desenvolvimento de jogos educativos.

###### 1) Tarefas de Viabilidade:

A primeira tarefa desta disciplina consiste em identificar as habilidades e competências técnicas (T1) necessárias para o desenvolvimento do projeto. A partir da identificação dessas, a seleção dos membros da equipe (T2) pode ser feita e, em seguida, essa equipe define a infraestrutura de trabalho (T3), ou seja, o local em que trabalhará, as máquinas que serão utilizadas, os requisitos de *hardware* necessários, dentre outras.

Essa infraestrutura é imprescindível para realizar-se a seleção das ferramentas de desenvolvimento (T4). É importante ressaltar que é feito apenas um levantamento, a fim de analisar a viabilidade técnica do projeto. A seleção dessas tecnologias é feita na disciplina *Design do Jogo*.

Adicionalmente, deve-se analisar a viabilidade econômica (T6), a partir do levantamento de todos os custos envolvidos no projeto (T5). Por fim, identifica-se as fases e tarefas técnicas (T7) para definir o cronograma do projeto (T8) e verificar se o mesmo é viável.


Essas tarefas da disciplina Viabilidade do Projeto são apresentadas na Figura 6. Ao se tornarem tarefas instanciadas , é possível estabelecer um fluxo de sequência entre essas tarefas, a partir do uso das notações UML de início, transição e fim para a Fase de Viabilidade, conforme mostra a Figura 7.



Figura 6. Tarefas da Disciplina Viabilidade do Projeto.

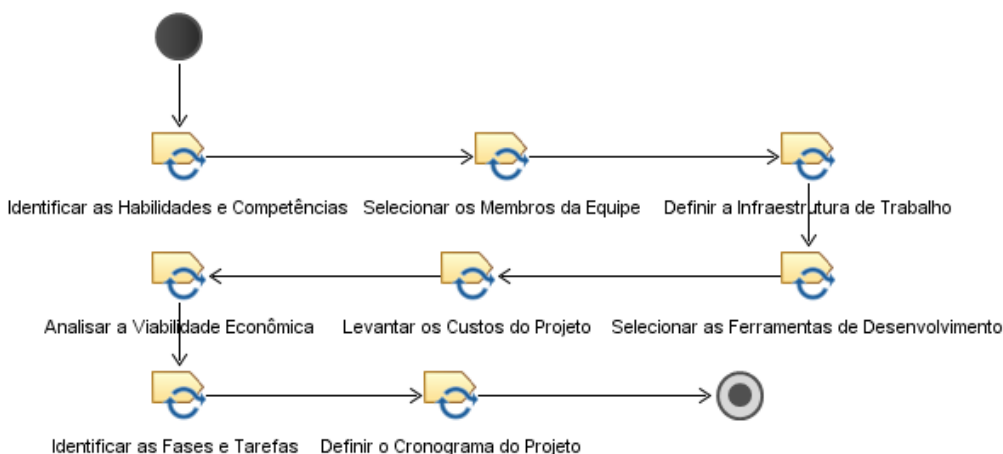


Figura 7. Tarefas Instanciadas da Fase de Viabilidade.

2) *Tarefas de Análise:*

As tarefas da disciplina Análise do Projeto consistem, basicamente, em realizar o levantamento dos requisitos do jogo educativo. Sendo assim, deve-se identificar os requisitos funcionais (T10) e não-funcionais (T11) que dependem diretamente dos objetivos de aprendizagem (T9). Recomenda-se, também, realizar o mapeamento entre esses requisitos (T12), a fim de identificar as dependências entre esses e, consequentemente, o impacto de mudanças durante o desenvolvimento do jogo. Essas tarefas são apresentadas na Figura 8.

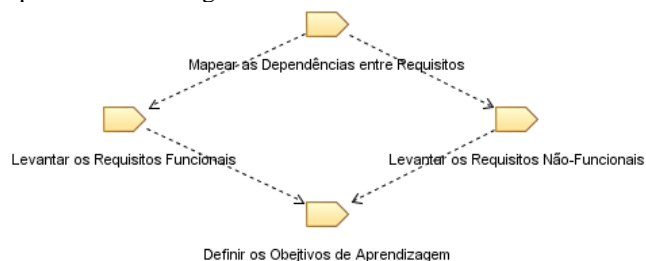








Figura 8. Tarefas da Disciplina Análise do Projeto.

Destaca-se que cada uma dessas pode ser detalhada, através da definição do papel responsável por sua execução, produto de trabalho consumido ou gerado (artefatos de entrada ou saída) e ferramentas utilizadas, respectivamente representadas pelas notações ,  e .

, conforme mostra o exemplo da atividade Levantar os Requisitos Funcionais na Figura 9.



Figura 9. Tarefa Levantar os Requisitos Funcionais.

Esses papéis , produtos de trabalho  e ferramentas  também podem ser instanciados, conforme exemplifica a Figura 10.

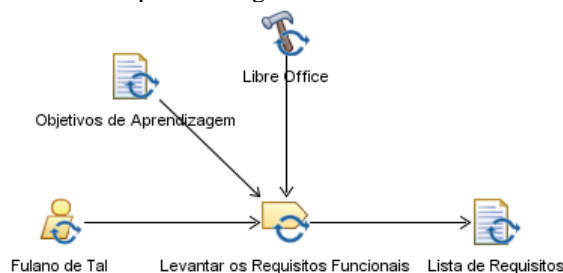


Figura 10. Instanciação da Tarefa Levantar os Requisitos Funcionais.

Neste momento, devem-se utilizar os nomes das pessoas que assumirão os papéis, os artefatos ao invés dos modelos e as ferramentas específicas ao invés da sua categoria.

3) *Tarefas de Concepção:*

A Figura 11 apresenta as tarefas da disciplina Concepção do Jogo. A primeira tarefa consiste na identificação do público-alvo (T13) do jogo, por exemplo, alunos do ensino fundamental, médio, superior ou profissionais da indústria. Essa identificação é necessária

para que se possa definir a proposta pedagógica (T14) de ensino e, a partir dessa, as formas de *feedback* educacional (T15) que o jogo educativo deverá apresentar. A partir dessa proposta pedagógica, também, se estabelecem os objetivos e metas (T16) do jogo. Em seguida, determinam-se o tema e a história do jogo (T17), que devem, de maneira criativa e divertida, permitir que os objetivos e metas sejam alcançados. Esses objetivos e metas influenciam a definição das regras e critérios (T18) para se vencer.

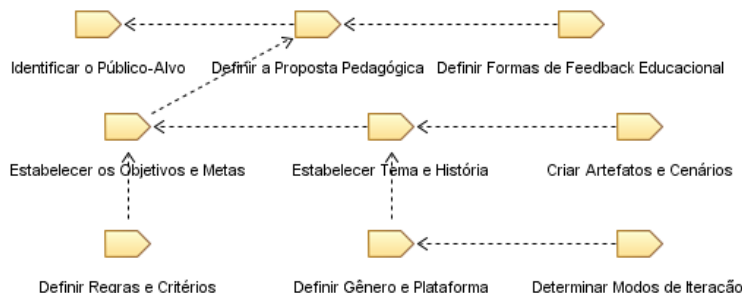


Figura 11. Tarefas da Disciplina Concepção do Jogo.

Adicionalmente, criam-se os artefatos e cenários (T19) que darão vida a essa história. A escolha do gênero e plataforma (T20) deve ser realizada de acordo com a temática do jogo. Esses, por sua vez, determinam os modos de interação (T21). Por exemplo, se a plataforma for *mobile*, o modo de interação poderá ser *touchscreen*.

Os resultados da execução dessas tarefas podem ser registrados através do documento *Game Concept* [11], que delimita o conteúdo do jogo que se pretende desenvolver.

4) *Tarefas de Design:*

As tarefas da disciplina *Design* do Jogo são apresentadas na Figura 12. Inicialmente, os membros da equipe devem assinar um termo de confidencialidade (T22), pois, durante essa etapa, as artes visuais do jogo serão criadas, não podendo ser divulgadas até o lançamento do mesmo. Após a assinatura, alocam-se as tarefas para os membros da equipe (T23) de acordo com o processo de desenvolvimento especificado (T24).

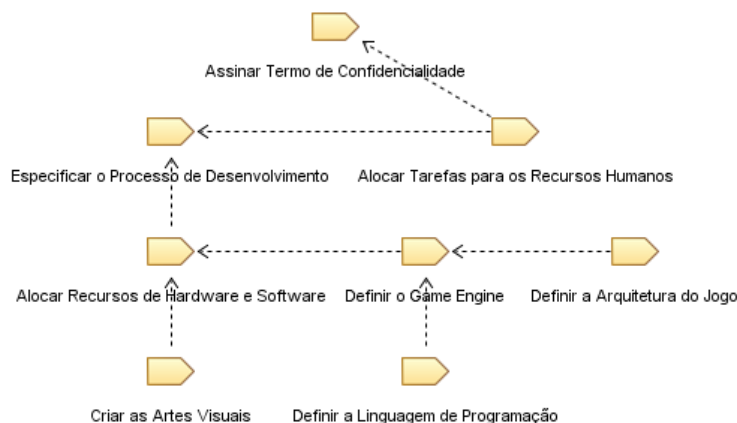


Figura 12. Tarefas da Disciplina Design do Jogo.

As tarefas desta disciplina determinam os recursos de *hardware* e *software* a serem alocados (T25). Esses recursos influenciarão na definição do *game engine* (T26), da linguagem de programação (T27) e arquitetura (T28).

Por fim, esses recursos são necessários para a criação das artes visuais (T29), como os personagens e cenários da história e a modelagem dos demais elementos 2D/3D.

Os resultados dessas tarefas podem ser documentados no *Game Design Document* (GDD) [12]. Quanto aos artefatos técnicos, que apoiarão a codificação do projeto, como os diagramas UML que representam os requisitos, os modelos de banco de dados e o padrão arquitetural, podem ser registrados no Documento de *Design Técnico* [12].

5) *Tarefas de Implementação:*

As tarefas de Implementação do Jogo são bem objetivas, consistindo na codificação dos requisitos (T30) e na criação de protótipos (T31), que permitem testar o funcionamento do jogo (T32), conforme Figura 13.

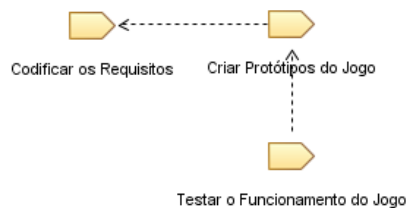


Figura 13. Tarefas da Disciplina Implementação do Jogo.

6) *Tarefas de Teste:*

A disciplina Teste do Jogo consiste na procura e correção de problemas (*bugs*) (T33) e na análise do envolvimento do público-alvo (T34), que permite verificar se o jogo alcançou os seus objetivos de ensino (T35). Essas tarefas são apresentadas na Figura 14.

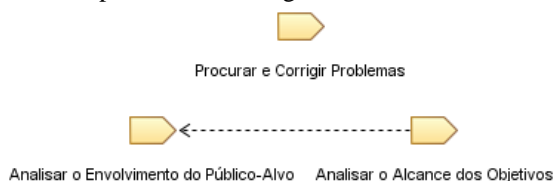


Figura 14. Tarefas da Disciplina Teste do Jogo.

7) *Tarefas de Lançamento:*

A disciplina Lançamento do Jogo, conforme mostra a Figura 15, possui as tarefas de entrega do produto final (T36) e a sua validação pelo cliente (T37).

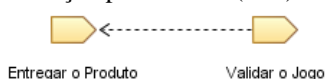


Figura 15. Tarefas da Disciplina Lançamento do Jogo.

8) *Tarefas de Pós-Lançamento:*

Na disciplina Pós-Lançamento, os desenvolvedores devem monitorar as reações do público-alvo (T38), a fim de obter *feedback* dos usuários (T39) em relação ao apoio do jogo ao ensino de um conteúdo específico. Essas duas atividades são representadas na Figura 16.



Figura 16. Tarefas da Disciplina Pós-Lançamento.

V. AVALIAÇÃO POR PAINEL DE ESPECIALISTAS

A avaliação do processo PROPAGA foi feita através de um painel de especialistas [22]. Esse método, baseado na opinião de especialistas, é utilizado ou quando não existe unanimidade de opinião sobre determinado assunto, ou para resolver as discordâncias. Assim, é possível estabelecer um consenso sob o objeto-alvo da avaliação.

Os critérios para a seleção dos especialistas foram: i) ter experiência acadêmica ou profissional na área de jogos educativos; ii) possuir titulação, no mínimo graduação.

Para obter as percepções destes desenvolvedores, solicitou-se que esses analisassem integralmente o modelo, a fim de fornecer suas contribuições e recomendações. Durante essa avaliação, foram instruídos a considerar não somente as fases que seguem no desenvolvimento dos seus jogos, mas sim a relevância das fases apresentadas.

A. *Design*

Inicialmente, houve uma análise e avaliação individual da adoção das fases dos processos identificados na revisão da literatura apresentada na Seção III, no período de 15 de janeiro a 14 de maio de 2018. Em seguida, foi feita uma avaliação de consenso entre os especialistas, a partir das fases selecionadas na primeira etapa e de acordo com o resultado do mapeamento apresentado na Subseção III-F, no período de 21 de maio a 04 de junho de 2018. Essa segunda etapa objetivou resolver as discordâncias,

possibilitando o desenvolvimento de um consenso final. Neste sentido, para cada uma das fases, os especialistas justificaram as suas respostas em relação à relevância das mesmas. Como resultado, foi emitido um parecer conclusivo da avaliação. Ambas as etapas foram realizadas via questionários, disponível em <https://bit.ly/2jWPr4N>.

Assim, na primeira etapa, perguntou-se “*Quais destas fases foram adotadas no processo de desenvolvimento do seu jogo educacional?*”. As opções de respostas eram 13 (treze) fases: Estudo de Viabilidade; Fase de Análise; Fase de Planificação; Fase de *Design*; Fase de Concepção; Pré-Produção; Fase de Implementação; Produção; Fase de Avaliação; Fase de Teste; Pós-Produção; Lançamento; Pós-Lançamento. Nessa etapa, os especialistas sinalizavam a adoção marcando uma caixa de seleção (*checkbox*).

Obteve-se como resultado que todas as fases foram seguidas pelo menos por um especialista. Assim, a partir da análise comparativa entre essas fases, foi possível identificar 8 (oito) fases: Estudo de Viabilidade; Fase de Análise; Fase de *Design*; Fase de Concepção; Fase de Implementação; Fase de Teste; Lançamento; Pós-Lançamento. Então, na avaliação de consenso, perguntou-se para cada uma dessas: “*Você considera essa fase necessária?*”. O especialista podia responder Sim ou Não, tendo que justificar sua resposta em ambas as opções.

B. *Participantes*

Foram convidados, via e-mail, 7 (sete) especialistas, com média de atuação na área de jogos educativos de 3,7 anos. Esses desenvolvedores são da região Norte e Sudeste do Brasil e possuem média de idade igual a 26 anos.

Quanto à experiência, esses já desenvolveram jogos dos gêneros Ação e Aventura, Simulação e Quiz, para as plataformas Android, iOS, PC e navegador Chrome. As principais linguagens utilizadas por esses são Java, C#, Objective-C e Javascript.

C. *Resultados*

A partir da análise dos resultados da primeira etapa, obteve-se como resultado a quantidade de adoção das fases pelos especialistas, apresentada na Figura 17. Desta forma, todas essas fases foram consideradas relevantes, pois pelos menos um especialista as seguiu na criação de seu jogo.

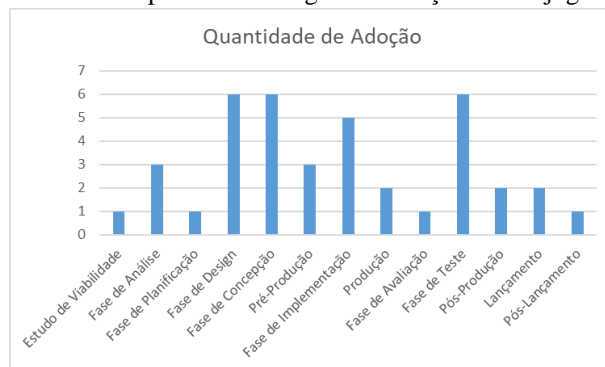


Figura 17. Quantidade de Adoção das Fases pelos Especialistas.

Na avaliação de consenso, todas as oito fases, apresentadas na Figura 5, foram consideradas relevantes. Apenas um especialista não achou necessária a fase de Lançamento, pois considera que o cliente deve participar da fase de Teste e, conseqüentemente, receber o jogo durante essa etapa.



Desta forma, de acordo com o consenso e a análise do parecer final, os especialistas consideraram que essas oito fases são necessárias e devem ser seguidas no desenvolvimento de jogos educativos. As considerações emitidas no parecer são discutidas na subseção a seguir.

#### D. Análise e Discussões

##### 1) Sobre a Fase de Viabilidade

Em relação à Fase de Viabilidade, os especialistas destacaram que deve-se analisar se o desenvolvimento do jogo será viável em relação ao tempo de duração do projeto (T8), se o mesmo possui infraestrutura suficiente (T3), equipe (T2), entre outros. Dessa forma, a análise da viabilidade irá confirmar se o projeto tem recursos e planejamento para ser finalizado dentro dos prazos e se o orçamento (T5) irá suprir todas as necessidades.

Um especialista enfatizou que a equipe possuir conhecimento das ferramentas utilizadas (T4) é essencial para o sucesso do projeto. A questão de licenças e custos de tais ferramentas é outro aspecto destacado por esse.

##### 2) Sobre a Fase de Análise

Para a Fase de Análise, destacaram que é necessário considerar o público-alvo e sua demanda específica (T9). Assim, analisar os requisitos se torna fundamental para melhor abstração do objetivo do projeto. O levantamento correto deles, logo no início do projeto, aumenta a possibilidade de sucesso e, possivelmente, diminui a probabilidade de futuras correções.

Os requisitos devem ser muito bem documentados (T12), sendo necessário especificar os requisitos funcionais (T10) e não funcionais (T11). Em relação aos requisitos não funcionais, um especialista destacou que o jogo tem que se adequar ao seu público (ser ergonômico).

##### 3) Sobre a Fase de Concepção

Na Fase de Concepção ocorre o detalhamento do jogo, a partir da definição do estilo de arte (T19), *level design* (T18), mecânicas do jogo (T20), etc. Nela, ocorre a divisão das tarefas entre programadores, artistas, *designers*, etc.

Um especialista destaca que o jogo educativo precisa incentivar o seu jogador (T21) a continuar jogando de alguma forma. Assim, na Fase de Concepção, serão especificados elementos lúdicos (T15) (T19). Na opinião dele, esse é o maior gargalo em jogos educativos: “quase não há o sentimento de diversão” (T17).

Por fim, destacam que o documento *game concept* é muito importante, já que nele estará todas as definições do jogo (T16), o que é essencial para o seu desenvolvimento.

##### 4) Sobre a Fase de Design

Durante a Fase de *Design* é designada uma estrutura visual (T29) e como esses elementos vão se comportar no jogo, o que pode ser mudado constantemente durante o desenvolvimento. Esse *design* torna o jogo divertido e engajador para o público, na opinião dos especialistas.

Do ponto de vista do usuário (jogador), o visual é uma das primeiras coisas a serem avaliadas, e acaba sendo decisivo para determinar se o jogador vai continuar o jogo ou não.

Em relação à definição da linguagem de programação (T27), *engine* (T26) e arquitetura (T28), os especialistas destacaram que é necessário que a equipe saiba trabalhar com seus limites e falta de conhecimento técnico na área (T23). Relataram que produzir jogos é bastante difícil e desgastante. Pessoas sem experiência técnica na área de

desenvolvimento não entendem isso e acabam achando que criar jogos é tão divertido quanto jogá-los. Assim, consideram extremamente necessário dosar o escopo do *design* juntamente com o tempo e esforço de implementação (T24).

##### 5) Sobre a Fase de Implementação

Os especialistas consideram a Fase de Implementação extremamente complexa, pois, nela, deve-se cumprir os objetivos estabelecidos, a fim de desenvolver um produto de qualidade no tempo definido (T30).

Recomendam que o jogo seja implementado através de protótipos (T31) e que passe por testes (T32) até chegar em um resultado mais próximo ao produto final planejado. Para tal, uma boa estrutura (arquitetura) é essencial.

Por fim, destacam que a criação da interface, mundo do jogo e itens dos personagens irá gerar um protótipo que permitirá obter *feedback* dos jogadores para futuras melhorias na versão final.

##### 6) Sobre a Fase de Teste

A Fase de Teste objetiva validar tudo que foi desenvolvido, pois, caso passe por esses testes, o jogo poderá ser considerado útil para atender os objetivos especificados (T35). Sendo assim, os especialistas consideram a realização dessa fase necessária antes que seja lançada uma versão.

Nessa fase, a equipe e jogadores validam o jogo, em todos os seus módulos, à procura de defeitos e melhorias (T33). Deve-se verificar se o jogo está atendendo os requisitos e seguindo o *game concept*.

Os especialistas destacam que não adianta ter implementado algo extremamente complexo se o público-alvo não tem vontade de jogar/testar (T34). Entretanto, consideram normal ter que voltar para a Fase de Implementação para corrigir algo ou até mesmo criar outra funcionalidade. Se isso não acontecer, recomendam desconfiar dos resultados dos testes realizados.

##### 7) Sobre a Fase de Lançamento

A maioria dos especialistas considera a Fase de Lançamento necessária, pois será o primeiro contato dos jogadores com o produto criado (T36) e, por isso, deve-se atentar para o comportamento dos usuários ao utilizarem o jogo. É uma oportunidade para se avaliar se o projeto do jogo foi concluído conforme o planejado (T37).

No entanto, um especialista considerou essa fase desnecessária, pois essa entrega aos clientes poderia ser realizada após a conclusão da Fase de Teste. Sendo assim, na opinião dele, só seria útil caso o projeto tenha sido projetado para a indústria de entretenimento.

##### 8) Sobre a Fase de Pós-Lançamento

Segundo os especialistas, é necessário continuar coletando informações do jogo, na Fase Pós-Lançamento, para gerar futuras atualizações e correções. Através do monitoramento do *feedback* dos jogadores (T39), os desenvolvedores podem visualizar pontos de melhoria e corrigir erros, sempre buscando atender às necessidades dos usuários.

Durante a Fase de Pós-Lançamento, também, devem ser disponibilizadas as atualizações ou expansões do jogo. Os especialistas recomendam o uso de ferramentas de *Key Performance Indicators* (KPI) para medir indicadores de uso do jogo com o público real (T38) e, com base nisso, inferir métricas que podem ser analisadas de diversas formas.

### E. Ameaças à Validade

Dentre as ameaças, tem-se o fato do painel ter envolvido apenas 7 especialistas, o que afeta à validade de conclusão. No entanto, destaca-se a média de experiência (mais de 3 anos) e o conhecimento destes especialistas em diversas plataformas e linguagens de programação.

Adicionalmente, há uma ameaça relacionada ao fato da análise ter considerado apenas o modelo do processo, o que pode ter limitado o entendimento dos especialistas. Nesse sentido, utilizou-se uma descrição textual para cada fase, de acordo com os autores que propuseram essas. Por fim, realizou-se uma avaliação de consenso a fim de discutir eventuais equívocos em relação ao modelo.

## VI. CONCLUSÃO

A área de jogos educativos representa uma parcela satisfatória do mercado nacional, no que diz respeito à criação dos mesmos. Contudo, mesmo com esse cenário favorável, continua sendo uma área pouco atrativa para desenvolvedores, devido à falta de conhecimento sobre as etapas básicas necessárias para criar esse tipo de jogo e ao baixo custo-benefício de produção do mesmo.

Através do processo padrão PROPAGA, os desenvolvedores podem conhecer e seguir as etapas necessárias para criar jogos educativos de maneira satisfatória, conforme validado por 7 especialistas. Para iniciantes na área de desenvolvimentos de jogos, este pode ser um ponto de partida. Já para desenvolvedores experientes, esse processo poderá ser instanciado e modificado de acordo com as particularidades do seu projeto. Como benefícios, estes terão uma representação visual das atividades dos seus projetos, o que pode facilitar a comunicação interna e treinamento de novos membros da equipe. Consequentemente, esse processo poderá ser melhor gerenciado e reutilizado em outros projetos. Isso permite uma evolução e melhoria contínua do processo seguido pelas empresas de desenvolvimento de jogos.

Quanto ao baixo custo-benefício da área de jogos educativos, observa-se que faltam incentivos financeiros ou políticas públicas para a criação desses jogos. Assim, espera-se que o seguimento desse processo permita o desenvolvimento e a adoção de jogos em sala de aula. Dessa forma, poder-se-á, indiretamente, melhorar o ensino no Brasil, que carece desses tipos de abordagens.

Como trabalhos futuros, o processo será disponibilizado no formato XML, permitindo assim a modelagem de processos instanciados a partir do apoio ferramental da Spider-PM [19]. Por fim, pretende-se realizar um estudo de caso da aplicação do PROPAGA com alunos da disciplina “Desenvolvimento de Jogos” do curso de Sistemas de Informação da UFPA que abordará a criação desses tipos de jogos. Assim, o processo poderá ser avaliado quanto ao apoio aos desenvolvedores iniciantes.

## REFERÊNCIAS

- [1] J. Silva e A. Pinto, “Geração C: Conectados em Novos Modelos de Aprendizagem,” Anais do Simpósio Brasileiro de Jogos de Computador e Entretenimento Digital (SBGAMES 09), Rio de Janeiro, 2009, pp. 48-51.
- [2] A. Fleury, L. Sakuda e J. Cordeiro, “I Censo da Indústria Brasileira de Jogos Digitais,” [http://www.abragames.org/uploads/5/6/8/0/56805537/i\\_censo\\_da\\_industria\\_brasileira\\_de\\_jogos\\_digitais.pdf](http://www.abragames.org/uploads/5/6/8/0/56805537/i_censo_da_industria_brasileira_de_jogos_digitais.pdf) (acessado em 13 de Jun. 2019).
- [3] E. Monsalve, V. Werneck e J. Leite, “SimulES-W: Um Jogo para o Ensino de Engenharia de Software,” Anais do III Fórum de Ensino de Engenharia de Software (FEES 10), Salvador, 2010, pp. 17-26.
- [4] L. Campos, “O que os Jogos de Entretenimento têm que os Jogos Educativos não têm,” Anais do Simpósio Brasileiro de Jogos de Computador e Entretenimento Digital (SBGAMES 09), Rio de Janeiro, 2009, pp. 1-20.
- [5] L. Correia, V. Cayres e R. Ramos, “Fundamentos de Jogos em Abordagens Educacionais: Um Retrato da Motivação do Aluno Jogador,” Anais do Simpósio Brasileiro de Jogos de Computador e Entretenimento Digital (SBGAMES 16), São Paulo, 2016, pp. 1202-1204.
- [6] P. Battistella, ENGAGED: Um Processo de Desenvolvimento de Jogos para Ensino em Computação. Tese de Doutorado do Programa de Pós-Graduação em Ciência da Computação, Universidade Federal de Santa Catarina. UFSC, 2016.
- [7] R. Savi, Avaliação de Jogos voltados para a Disseminação do Conhecimento. Tese de Doutorado do Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento, Universidade Federal de Santa Catarina. UFSC, 2011.
- [8] M. Kishimoto, “O Jogo e a Educação Infantil,” em *Pioneira Thomson Learning*, São Paulo, 2003.
- [9] A. Barboza e I. Silveira, “PerMotivE: Um Modelo conceitual de Persuasão, Motivação e Engajamento para Jogos Educacionais,” Anais do Simpósio Brasileiro de Jogos de Computador e Entretenimento Digital (SBGAMES 16), São Paulo, 2016, pp. 920-929.
- [10] E. Clua e J. Bittencourt, “Desenvolvimento de Jogos 3D: Concepção, Design e Programação,” Anais da XXIV Jornada de Atualização em Informática do Congresso da Sociedade Brasileira de Computação, São Leopoldo, 2005, pp. 1313-1356.
- [11] G. Dambros, Por uma Cartografia Escolar Interativa: Jogo Digital para a Alfabetização Cartográfica no Ensino Fundamental. Dissertação de Mestrado do Programa de Pós-Graduação em Geografia e Geociências, Universidade Federal de Santa Maria. UFSM, 2014.
- [12] S. Gonçalves e C. Werner, “Conceitos e Desenvolvimento de Jogos Digitais Educativos,” Anais da XIII Semana de Informática e XV Mostra de Trabalhos de Iniciação Científica, Paranavaí, 2016.
- [13] W. Humphrey, “Managing the Software Process,” in *The SEI Series in Software Engineering*. Addison-Wesley, 1989.
- [14] J. Lomcham, “A Structured Conceptual and Terminological Framework for Software Process Engineering,” Proc. The Second International Conference on the Software Process: Continuous Software Process Improvement (ICSP 93), Berlin, 1993, pp.41-53.
- [15] International Organization for Standardization/International Electrotechnical Commission. ISO/IEC 15504-2: Information Technology - Process Assessment – Part 2 – Performing an Assessment. Geneva: ISO, 2003.
- [16] L. Bezerra, L. Silva, I. Silveira, T. Kuma e C. Junior, “Experiências no Ensino de Projeto de Jogos Digitais,” Anais do Simpósio Brasileiro de Jogos de Computador e Entretenimento Digital (SBGAMES 06), Recife, 2006.
- [17] M. Kellner e G. Hansen, “Software Process Modeling,” <http://www.sei.cmu.edu/reports/88tr009.pdf> (acessado em 18 de Jun. 2018).
- [18] R. Barros e S. Oliveira, “SPIDER\_ML: Uma Linguagem de Linguagem de Modelagem de Processos de Software,” Anais da Escola Regional de Informática (ERIN 10), Manaus, 2010.
- [19] R. Barros e S. Oliveira, “Spider-PM: Uma Ferramenta de Apoio à Modelagem de Processos de Software,” Anais do Encontro Anual de Computação, Manaus, 2010.
- [20] OMG, “Software & Systems Process Engineering Meta-Model Specification,” <http://www.omg.org/spec/SPEM/2.0/PDF> (acessado em 14 de Jun. 2018).
- [21] K. Siqueira, E. Nascimento e C. Portela, “Uma Proposta de Processo Padrão para Apoiar o Desenvolvimento de Jogos Educativos”, Anais do Simpósio Brasileiro de Jogos e Entretenimento Digital, Foz do Iguaçu, 2018.
- [22] S. Beecham, T. Hall, C. Britton, M. Cottee, A. Rainer, “Using an Expert Panel to Validate a Requirements Process Improvement Model,” *The Journal of Systems and Software*, v. 76, 2005.