

CBBIE 2018
CONGRESSO BRASILEIRO DE
INFORMÁTICA NA EDUCAÇÃO

ANAIS

**VII JORNADA DE ATUALIZAÇÃO EM
INFORMÁTICA NA EDUCAÇÃO
(JAIE 2018)**

Coordenador:
Henrique Schneider (UFS)

29 de Outubro a 01 de Novembro de 2018
Fortaleza, CE

VII Congresso Brasileiro de Informática na Educação

ANAIS DA VII JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA NA EDUCAÇÃO (JAIE 2018)

<http://cbie2018.virtual.ufc.br>
<http://www.br-ie.org/pub/index.php/pie>

ISBN: 978-85-7669-462-5

Fortaleza, CE, Brasil
29 de outubro a 01 de novembro de 2018

Editora

Sociedade Brasileira de Computação – SBC

Organizadores

Henrique Schneider (UFS)

Organização e Execução

Universidade Federal do Ceará (UFC)

Promoção

Comissão Especial de Informática na Educação – CEIE

Realização

Sociedade Brasileira de Computação – SBC

Este livro contém quatro capítulos correspondentes aos minicursos conduzidos na VII Jornada de Atualização em Informática na Educação (JAIE 2018) do Congresso Brasileiro de Informática na Educação (CBIE 2018). A inclusão nesta publicação não necessariamente constitui endosso pelos editores e/ou organizadores.

A fonte e os direitos da SBC devem ser devidamente referenciados. As cópias não devem ser utilizadas de nenhum modo que implique o endosso da SBC. Cópias da obra não podem ser colocadas à venda sem a autorização expressa da SBC.

Permissão para fazer cópias impressas ou digitais de todo ou parte deste trabalho para uso pessoal ou acadêmico é concedido sem taxas desde que cópias não sejam feitas ou distribuídas para renda ou vantagem comercial e que cópias contenham esta observação e citação completa na primeira página.

Sociedade Brasileira de Computação
CNPJ no. 29.532.264/0001-78
Inscrição Estadual isenta
CCM nº 18115128
<http://www.sbc.org.br>

Av. Bento Gonçalves, 9500 – Setor 4 – Sala 116 – Prédio 43424 – Agronomia
CEP 91501-970 – Porto Alegre – RS, Brasil

Produzido em Fortaleza, CE, Brasil.

C749a

Congresso Brasileiro de Informática na Educação (CBIE); Jornada de Atualização em Informática na Educação (JAIE 2018) (7.:2018: Fortaleza, CE)

Anais do VII Congresso Brasileiro de Informática na Educação; Jornada de Atualização em Informática na Educação (JAIE 2018) [Recurso eletrônico], 29 de outubro a 01 de novembro de 2018./ Organizado por Universidade Federal do Ceará e Henrique Schneider. – Fortaleza: SBC, 2018.

93p. Recurso Eletrônico disponível em: <http://br-ie.org/pub/index.php/pie/issue/view/177>

ISBN 978-85-7669-462-5

1. Informática na Educação - Anais. 2. Informática aplicada - Congresso. 3. Informática na Educação – Jornada. 4. Informática Educativa - Atualização. I. Título. II. Organizador (Org.).

CDD: 371.334

Sobre o Congresso Brasileiro de Informática na Educação

O **Congresso Brasileiro de Informática na Educação (CBIE)** é um evento anual da SBC, de caráter internacional, que busca promover e incentivar as trocas de experiências entre as comunidades científica, acadêmica, profissional, governamental e empresarial na área de Informática na Educação. Em 2018, o CBIE foi realizado em Fortaleza/CE sob a coordenação dos professores Herbert Lima e José Aires de Castro Filho. O tema do CBIE foi “Informática na Educação: Educação e Empreendedorismo”. Em sua sétima edição, o CBIE 2018 engloba o 29º Simpósio Brasileiro de Informática na Educação (SBIE), o 24º Workshop de Informática na Escola (WIE), as Jornadas de Atualização em Informática na Educação (JAIE), a Mostra de Práticas de Informática na Educação (MPIE), o Concurso de Teses, Dissertações e Trabalhos de Conclusão de Curso (CTD-IE) e diversos workshops satélites, painéis e mesas redondas. O evento contou com cerca de 800 participantes.

Prefácio

Este ano, nove propostas de minicurso foram submetidas através do sistema JEMS/SBC. Após vencido o prazo para submissões, foram realizadas as análises balizadas no barema de avaliação disposto no JEMS pelos onze pesquisadores que aceitaram o nosso convite para avaliarem, cada um, até duas propostas. Depois de resolvidos os casos de empate por meio de mais uma rodada de avaliação, foram aceitas quatro propostas de minicursos de três horas, os quais serão desenvolvidos ao longo do CBIE e cujos artigos estão reunidos nestes anais, refletindo a preocupação que sempre norteou a comunidade brasileira de Informática na Educação, com a seriedade e a qualidade dos seus eventos científicos.

Antes de trazer um breve bosquejo sobre cada artigo, quero externar os meus agradecimentos aos colegas que compuseram a Comissão Científica do JAIE 2018 e àqueles que deram suporte ao CBIE na Universidade Federal do Ceará, especialmente ao professores Sérgio Crespo Coelho da Silva Pinto e José Aires de Castro Filho, respectivamente Coordenador da Comissão Especial de Informática na Educação da Sociedade Brasileira de Computação e do Comitê de Organização do CBIE 2018, pela confiança e apoio recebido.

O artigo **“Geração Procedural de Conteúdo Aplicada a Jogos Digitais Educacionais”** discute a problemática de balancear o aspecto de entretenimento dos jogos com as demandas cognitivas da aprendizagem, apresentando a geração procedural de conteúdo como uma estratégia interessante para o desenvolvimento de jogos digitais educacionais.

O artigo **“Planejamento e Execução de Estudos Secundários em Informática na Educação: Um Guia Prático Baseado em Experiência”** apresenta os métodos “Revisão Sistemática da Literatura” e “Mapeamento Sistemático da Literatura” e ferramentas informáticas associadas, como meios eficazes de levantar o estado da arte sobre temas de pesquisas em informática educativa.

Finalmente, o artigo **“Desenvolvimento do Pensamento Computacional através do Uso de Tutoriais Interativos”** discute a relevância do Pensamento Computacional na contemporaneidade e apresenta as ferramentas “Tutorial Interativo” e a linguagem de programação em blocos “Scratch” como possíveis instrumentos para o desenvolvimento do Pensamento Computacional.

Dessa forma, pelo que foi apresentado, o JAIE 2018 reúne temáticas diversificadas e atuais e, por isso, importantes para a atualização dos pesquisadores em Informática Educativa. Portanto, indicamos a leitura dos artigos e, se possível, a realização dos minicursos, a fim de desfrutar da interação com os ministrantes e com os colegas, momentos ricos de aprendizado.

Fortaleza/CE, outubro de 2018.

Henrique Nou Schneider (UFS-IFS)
Coordenador JAIE 2018

Comitê de Programa

- Coordenador:

Henrique Nou Schneider | Universidade Federal de Sergipe (UFS)

- Membros do Comitê de Programa:

Alex Sandro Gomes | Universidade Federal de Pernambuco (UFPE)

Daniel Costa de Paiva | Universidade Federal Fluminense (UFF)

Edilson Ferneda | Universidade Católica de Brasília (UCB)

Edison Pimentel | Universidade Federal do ABC (UFABC)

Elaine dos Reis Soeira | Instituto Federal de Alagoas (IFAL)

Fábio Ferrentini Sampaio | Universidade Federal do Rio de Janeiro (UFRJ)

Henrique Nou Schneider | Universidade Federal de Sergipe (UFS)

José Francisco Magalhães Netto | Universidade Federal do Amazonas (UFAM)

Sérgio Crespo C. S. Pinto | Universidade Federal Fluminense (UFF)

Sheilla Silva Conceição | Universidade Federal de Sergipe (UFS)

Vinicius Silva Santos | Universidade Federal de Sergipe (UFS)

Outras Coordenações Relacionadas

- **Coordenação Geral do Congresso Brasileiro de Informática na Educação (CBIE 2018):**
José Aires de Castro Filho | Universidade Federal do Ceará (UFC)
Francisco Herbert Lima Vasconcelos | Universidade Federal do Ceará (UFC)
- **Coordenação do XXIX Simpósio Brasileiro de Informática na Educação (SBIE 2018):**
Thais Helena Chaves de Castro | Universidade Federal do Amazonas (UFAM)
Maria de Fátima Costa de Souza | Universidade Federal do Ceará (UFC)
- **Coordenação do XXIV Workshop de Informática na Escola (WIE 2018):**
Dennys Leite Maia | Universidade Federal do Rio Grande do Norte (UFRN)
- **Coordenação da VII Jornada de Atualização em Informática em Educação (JAIE 2018):**
Henrique Schneider | Universidade Federal de Sergipe (UFS)
- **Coordenação do Concurso Alexandre Direne de Teses, Dissertações e Trabalhos de Conclusão de Curso em Informática na Educação (CTD-IE 2018):**
Patricia A. Jaques Universidade do Vale do Rio dos Sinos (UNISINOS)
Eliseo Reategui | Universidade Federal do Rio Grande do Sul (UFRGS)
Sean Siqueira | Universidade Federal do Estado do Rio de Janeiro (UNIRIO)
- **Coordenação dos Workshops do CBIE 20178:**
Juscileide Braga de Castro | Universidade Federal do Ceará (UFC)
Thomaz Edson Veloso da Silva | Universidade Federal do Ceará (UFC)
- **Coordenação do IV Concurso Integrado de Desenvolvimento de Soluções de Tecnologia e Objetos de Aprendizagem para a Educação (Apps.Edu 2018):**
Gilvandenys Leite Sales | Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)
- **Coordenação da Mostra de Práticas de Informática na Educação (MPIE 2018):**
Dennys Leite Maia | Universidade Federal do Rio Grande do Norte (UFRN)
- **Coordenação do Painel de Políticas e Diretrizes para Informática na Educação (PPDIE 2018):**
Sérgio Crespo | Universidade Federal Fluminense (UFF)
- **Organização Geral:**
Francisco Herbert Lima Vasconcelos | Universidade Federal do Ceará (UFC)
José Aires de Castro Filho | Universidade Federal do Ceará (UFC)
Luiz Eduardo Rodrigues da Silva | Universidade Federal do Ceará (UFC)
Alexsandra Lúcia de Castro Andrade | Universidade Federal do Ceará (UFC)
Amanda Lima Girão | Universidade Federal do Ceará (UFC)
Bruno Queiroz da Silva | Universidade Federal do Ceará (UFC)
Evenila Mariana da Silva Holanda | Universidade Federal do Ceará (UFC)
Felipe Gomes Rufino Moura Paiva | Universidade Federal do Ceará (UFC)
Luzia Eveline Soares Serra | Universidade Federal do Ceará (UFC)
Maria Luiza Bispo | Universidade Federal do Ceará (UFC)
Marília Prado Pinto | Universidade Federal do Ceará (UFC)

Nathan da Silva Chausse | Universidade Federal do Ceará (UFC)

Patricia Maria Nunes de Souza Bispo | Universidade Federal do Ceará (UFC)

Samuel Lucas da Silva | Universidade Federal do Ceará (UFC)

Ygor de Oliveira Moreira | Universidade Federal do Ceará (UFC)

- Editoração dos anais:

Rafael Dias Araújo | Universidade Federal de Uberlândia (UFU)

Comitê Gestor da Comissão Especial de Informática na Educação (CEIE)

Coordenador: Sérgio Crespo | Universidade Federal Fluminense (UFF)

Vice-coordenadora: Thais Castro | Universidade Federal do Amazonas (UFAM)

Alex Sandro Gomes | Universidade Federal de Pernambuco (UFPE)

Carlos Roberto Lopes | Universidade Federal de Uberlândia (UFU)

Crediné de Menezes | Universidade Federal do Rio Grande do Sul (UFRGS)

Eleandro Maschio | Universidade Tecnológica Federal do Paraná (UTFPR)

Esdras Bispo | Universidade Federal de Goiás (UFG)

Fabiano Azevedo Dorça | Universidade Federal de Uberlândia (UFU)

José Aires de Castro Filho | Universidade Federal do Ceará (UFC)

Márcia Aparecida Fernandes | Universidade Federal de Uberlândia (UFU)

Patrícia Cabral Tedesco | Universidade Federal de Pernambuco (UFPE)

Patrícia Jaques | Universidade do Vale do Rio dos Sinos (UNISINOS)

Sean Siqueira | Universidade Federal do Estado do Rio de Janeiro (UNIRIO)

Thiago Schumacher Barcelos | Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP)

Tiago Primo | Universidade Federal de Pelotas (UFPel)

Sociedade Brasileira de Computação (SBC)

Presidência

Lisandro Zambenedetti Granville (UFRGS) – Presidente

Thais Vasconcelos Batista (UFRN) – Vice-Presidente

Diretorias

Renata de Matos Galante (UFRGS) – Diretora Administrativa

Carlos André Guimarães Ferraz (UFPE) – Diretor de Finanças

Antônio Jorge Gomes Abelém (UFPA) – Diretor de Eventos e Comissões Especiais

Renata Mendes de Araujo (UNIRIO) – Diretora de Educação

José Viterbo Filho (UFF) – Diretor de Publicações

Claudia Lage Rebello da Motta (UFRJ) – Diretora de Planejamento e Programas Especiais

Marcelo Duduchi Feitosa (CEETEPS) – Diretor de Secretarias Regionais

Eliana Silva de Almeida (UFAL) – Diretora de Divulgação e Marketing

Diretorias Extraordinárias

Ricardo de Oliveira Anido (UNICAMP) – Diretor de Relações Profissionais

Esther Colombini (UNICAMP) – Diretora de Competições Científicas

Raimundo José de Araújo Macêdo (UFBA) – Diretor de Cooperação com Sociedades Científicas

Cláudia Cappelli (UNIRIO) – Diretora de Articulação com Empresas

Leila Ribeiro (UFRGS) – Diretora de Ensino de Computação na Educação Básica

Sumário

Capítulo 1. Geração Procedural de Conteúdo Aplicada a Jogos Digitais Educacionais.....	01
<i>Wendell Oliveira de Araújo (UFRN), Eduardo Aranha (UFRN) e Charles Madeira (UFRN)</i>	
Capítulo 2. Planejamento e Execução de Estudos Secundários em Informática na Educação: Um Guia Prático Baseado em Experiências	21
<i>Luis Jorge Enrique Rivero Cabrejos (UFMA), Davi Viana (UFMA) e Rodrigo Pereira dos Santos (UNIRIO)</i>	
Capítulo 3. Desenvolvimento do Pensamento Computacional através do Uso de Tutoriais Interativos	53
<i>Gilson Pedroso dos Santos (UFOPA) e Ronilson dos Santos Bezerra (UFOPA)</i>	

Capítulo

1

Geração Procedural de Conteúdo Aplicada a Jogos Digitais Educacionais

Wendell Oliveira de Araújo, Eduardo Aranha e Charles Madeira

Abstract

Digital educational games have proved to be a great area of research that has been increasing over the years in the international scene. This fact has been due to the potential that the games have of fun, immersion and stimulation to the learning of natural and personalized way. However, one of the major challenges in this area is the costs involved and the difficulty in balancing entertainment with learning. In this sense, the procedural generation of content has been presented as an area that can aid the development of educational games. The procedural content generation (PCG) deals with the automatic creation of content through algorithms. Thus, this chapter will present techniques and a strategy for applying PCG in educational games.

Resumo

Jogos digitais educacionais têm se mostrado uma grande área de pesquisa que vem aumentando com o decorrer dos anos em todo o cenário internacional. Esse fato tem se dado pelo potencial que os jogos têm de diversão, imersão e estímulo à aprendizagem de maneira natural e personalizada. Entretanto, um dos grandes desafios encontrados nessa área são os custos envolvidos e a dificuldade em balancear o entretenimento com o aprendizado. Nesse sentido, a geração procedural de conteúdo tem se apresentado como uma área que pode auxiliar o desenvolvimento de jogos educacionais. A geração procedural de conteúdo (PCG) trata da criação automática de conteúdo por meio de algoritmos. Assim, este capítulo apresentará técnicas e uma estratégia para aplicação da PCG em jogos educacionais.

1.1. Indústria de jogos digitais e seu potencial para melhoria da educação

A indústria de jogos digitais vem crescendo consideravelmente nos últimos anos e, segundo a consultoria Newzoo (2018), continuará crescendo na ordem de 11% ao ano até atingir o faturamento de US\$ 180 bilhões em 2021 (ver Figura 1), o que consiste em um montante maior que a soma das indústrias do cinema e da música. Essa evolução acelerada da indústria de jogos também tem impulsionado os investimentos em pesquisas que buscam novas formas de fazer uso dos jogos nas mais diversas áreas tais como educação, saúde, simulação, entre outras. Devido ao grande número de pesquisas já efetuadas tendo como objetivo o uso de jogos na educação, é possível constatar o grande potencial que os jogos têm para contribuir com a melhoria da área. Savi e Ulbricht (2008) apresentam diversos benefícios que os jogos digitais educacionais podem promover, como socialização, engajamento, desenvolvimento de habilidades cognitivas, aprendizado por descoberta, facilitação do aprendizado, entre outros. Dessa forma, os jogos educacionais se mostram aptos a serem utilizados como uma ferramenta de ensino. Portanto, eles têm ganhado mais espaço nas escolas, tornando as aulas mais lúdicas e agradáveis para os alunos, sendo uma ótima forma de motivá-los e estimulá-los na superação dos desafios impostos nos ambientes virtuais simulados [Lara 2003].



Figura 1.1 Evolução do faturamento da indústria de jogos digitais durante no período de 2012 a 2021. De 2012 a 2017, os valores são reais. De 2018 a 2021, os valores correspondem a previsões baseadas em uma taxa de crescimento de 11% ao ano.

Sendo assim, os jogos se constituem como uma ferramenta importante no processo de ensino-aprendizagem dos estudantes. Contudo, para serem efetivos, os jogos precisam ter objetivos de ensino-aprendizagem bem definidos que proporcionem boas estratégias para a prática de conteúdos e o desenvolvimento de habilidades [Savi e Ulbricht 2008].

1.2. Desafios para o desenvolvimento de jogos digitais, especialmente os educacionais

Um dos grandes desafios para o desenvolvimento de jogos digitais são os custos envolvidos com a sua produção. A produção de um jogo é uma tarefa multidisciplinar que demanda o trabalho conjunto de várias áreas de conhecimento (programação, artes, design, música) para poder conseguir entregar um produto de boa qualidade de acordo com os padrões atuais de mercado. Os custos envolvidos têm uma relação direta com a complexidade do jogo que está em produção, influenciando principalmente, e de forma considerável, a etapa de produção de conteúdo (partes que envolvem um jogo tais como cenários, objetos, personagens, narrativas, texturas, músicas, etc.). Com isso, muitas vezes os desenvolvedores são levados a desistir do projeto durante o seu desenvolvimento, enquanto que alguns outros, que trabalham com equipes menores, tentam buscar alternativas para reduzir os custos de produção. Apesar da forte taxa de crescimento da indústria, ainda há uma carência de ferramentas que auxiliem os designers e artistas durante o processo de produção dos jogos, o que o torna bastante demorado e, conseqüentemente, custoso.

O “sucesso” de um jogo depende, na grande maioria das vezes, da experiência que ele proporciona aos jogadores. Muitas pesquisas apontam que a experiência proporcionada por um jogo depende muito do nível de desafio que é imposto ao jogador, adaptando-se às suas capacidades. Contudo, com o aumento do número de jogadores habilidosos que conseguem superar com muita facilidade os desafios propostos, os desenvolvedores passaram a se preocupar com a criação de desafios adequados para esses jogadores, visando que os mesmos não percam o interesse no jogo por não se sentirem desafiados [Bauckhage et al. 2012]. Assim, é necessário que sejam buscadas alternativas ao processo de produção padrão com o intuito de diminuir os custos, gerando rapidamente desafios adequados aos níveis dos jogadores.

Quando tratamos especificamente de jogos educacionais, além dos custos envolvidos com o processo de produção, a proposta do jogo precisa estar bem definida com os objetivos de ensino que deve ser passado, se ele deve ser usado para ensinar o aluno ou somente como uma ferramenta para prática de conteúdo, ajudando no desenvolvimento de certas habilidades [Savi e Ulbricht 2008]. Na realidade, o que geralmente acontece é que os projetos voltados ao desenvolvimento de jogos educacionais dispõem de financiamentos reduzidos, o que leva-os na maioria das vezes a dispor de equipes despreparadas que constroem produtos com foco quase que exclusivo no conteúdo a ser ensinado. Por esta razão, a experiência do jogador, que é um fator essencial do sucesso dos jogos comerciais, se torna negligenciada, fazendo com os jogos educacionais sejam reconhecidos pelos alunos como jogos simples e chatos, ou seja, que proporcionam uma experiência desinteressante.

Com o objetivo de melhor elaborar os jogos educacionais a fim de levá-los a um outro patamar, precisamos de alternativas que facilitem e reduzam os custos do processo de desenvolvimento, muito mais ainda do que no contexto dos jogos comerciais devido aos financiamentos reduzidos. Portanto, é de fundamental importância que tenhamos em mente a necessidade de gerar uma boa experiência para o jogador, proporcionando uma nova visão dentro da equipe de desenvolvimento focada no engajamento do jogador. Neste sentido, as técnicas de geração procedural de conteúdo (PCG) [Togelius *et al.* 2011]

se mostram como uma excelente alternativa pois permitem alavancar o processo de geração de conteúdo dos jogos, que é bastante custoso quando efetuado de forma manual.

As pesquisas realizadas na área de PCG têm sido, em grande parte, voltadas para o processo de geração de fases de jogos comerciais. Esse processo faz uso de técnicas tanto para criar fases completas de um jogo quanto para criar partes de fases que podem ser completadas pelos designers. Elas são, em sua maioria, voltadas para gêneros específicos de jogos, como os de plataforma, corrida, masmorra e aqueles envolvendo puzzles físicos. Assim, essas técnicas vêm sendo aplicadas em jogos já consagrados no mercado como *Super Mario Bros* (Nintendo), *Angry Birds* (Rovio), entre outros. No entanto, percebe-se que a aplicação de técnicas de PCG ainda se apresenta em estágio embrionário no contexto da área de jogos educacionais [Rodrigues 2017]. Assim, há uma grande carência de pesquisas envolvendo PCG e jogos educacionais, sistematizando sua aplicação e considerando um aspecto adicional que é o objetivo pedagógico.

1.3. Geração procedural de conteúdo

De acordo com Togelius *et al.* (2011), a definição de geração procedural de conteúdo (PCG) é a criação algorítmica de conteúdo do jogo com entrada de usuários limitada ou indireta. Em outras palavras, a PCG pode criar um conteúdo com o auxílio de um usuário ou por conta própria.

Os primórdios do uso da PCG se deu por volta do ano de 1980, uma vez que os jogos precisavam evoluir e a capacidade de armazenamento das máquinas era limitada. Assim, *Elite* [Brabel e Bell 1984] e *Rogue* [Toy et al 1980] foram alguns dos primeiros jogos a utilizarem PCG (ver Figura 1.2). *Elite* para gerar proceduralmente 8 galáxias, cada uma delas com 256 planetas com propriedades únicas, e *Rogue* com níveis sendo gerados aleatoriamente a cada início de partida.

Com o passar dos anos, os produtores de jogos comerciais também começaram a fazer o uso da PCG em seus jogos como *Diablo* [Blizzard 1996], gerando mapas e alocando itens e inimigos no mapa de forma procedural. Outros exemplos são *Spore* [Maxis 2008], no qual o jogador desenvolvia novos personagens e as animações dos mesmos eram criadas usando técnicas de animação procedural, *Minecraft* [Persson 2009], sendo um dos mais populares por gerar o mundo inteiro e o conteúdo deste mundo, *Spelunky* [Yu 2008], para gerar automaticamente variações dos níveis do jogo, também sendo um jogo bastante utilizado em pesquisas na área.



Figura 1.2. Exemplos de jogos que fizeram uso de PCG no processo de desenvolvimento: (1) Elite (2) Rogue (3) Diablo (4) Spore (5) Minecraft (6) Spelunky.

O uso de PCG oferece diversos benefícios como a redução do tempo de produção do jogo, e conseqüentemente dos custos, visto que o computador é capaz de gerar o conteúdo de maneira rápida e automática. Por outro lado, a facilidade da PCG gerar novos conteúdos aumenta o fator de *replay* do jogo (alonga o ciclo de vida pois o jogador é propenso a retornar a jogar). Além disso, no contexto dos computadores da atualidade não existe mais uma forte restrição quanto ao poder de processamento e a capacidade de memória conforme existia nos anos 80, o que era um dos principais entraves da PCG.

Quanto às limitações que ainda persistem da PCG, temos o fato de que o conteúdo gerado é, na maioria dos casos, específico, como por exemplo no caso de um gerador de fases. Uma dificuldade da sua aplicação consiste na falta de controle sobre o conteúdo gerado, não permitindo garantir que o mesmo sempre tenha solução ou utilidade dentro do jogo. De maneira similar, também é difícil avaliar a qualidade do conteúdo gerado pelo algoritmo, já que são necessários estudos mais elaborados com usuários e que os interesses variam de jogador para jogador.

1.3.1. Tipos de conteúdo gerado

A PCG pode ser destacada da seguinte forma: os termos geração e procedural servem para indicar que estão sendo utilizados algoritmos e procedimentos para criar algo; O termo conteúdo passa a se referir às partes de um jogo que podem ser geradas, como mapas, regras do jogo, texturas, objetos, personagens, fases, entre outros. A partir dessas definições podemos considerar como exemplos de PCG:

- Uma ferramenta de software que cria masmorras (*dungeons*) para um jogo como *The Legend of Zelda* sem qualquer interferência humana, sendo um novo nível criado a cada vez que a ferramenta é executada;
- Um sistema que cria novas armas em um jogo de tiro espacial em resposta ao que o coletivo de jogadores faz, de modo que as armas apresentadas são versões evoluídas de armas que outros jogadores acharam divertidas para usar;

- Uma ferramenta de design gráfico que permite que um usuário crie mapas para um jogo de estratégia, enquanto avalia continuamente o mapa projetado para suas propriedades de jogabilidade e sugere melhorias no mapa para torná-lo mais equilibrado e mais interessante.

Por outro lado, não pode ser considerado como exemplos de PCG:

- Um editor de mapas para um jogo de estratégia que simplesmente permita ao usuário colocar e remover itens;
- Um jogador artificial para um jogo de tabuleiro.

Na área acadêmica, vários pesquisadores apresentam suas perspectivas e métodos para enfrentar os problemas que a PCG possui, como garantir que o conteúdo gerado seja jogável, controlar a geração do algoritmo, entre outros. Assim, resultando em uma série de novos métodos, variações e combinações já desenvolvidas, muitos ainda precisando dar continuidade às pesquisas para poderem ser aplicados no contexto de jogos reais.

A PCG dispõe de diversas formas nas quais conteúdos podem ser gerados, apresentando bastante espaço para novas pesquisas. Normalmente, o desenvolvimento ocorre para um tipo de conteúdo específico, sendo alguns exemplos deles citados a seguir:

- **Níveis/fases:** PCGs focados em fases são aqueles que se preocupam em criar fases completas de jogos, desde a adição de plataformas, inimigos, bônus para o jogador e também um dos tipos de conteúdo mais utilizados nas pesquisas. Um exemplo deste tipo de conteúdo é o trabalho de [Moghadam e Rafsanjani 2017];
- **Mapas/terrenos:** São focados somente em criar as estruturas dos mapas, muito comum em jogos 3D para criação das montanhas, ambientes de navegação conforme pode ser visto no trabalho de [Raffe et al. 2015];
- **Quest/história:** Bastante comum em jogos de RPG para criação de missões secundárias para o jogo, que ajudam o personagem a aumentar o nível de experiência. Um exemplo pode ser visto no trabalho de [Kybartas e Verbrugge 2014];
- **Elementos de Cenário:** Partes de estruturas que compõem um cenário. No trabalho de [Stephenson e Renz 2016] são geradas estruturas complexas para o jogo *Angry Birds*;
- **Armas:** Mais comum em jogos de tiro para o balanceamento das armas e criação de armas com um visual que agrada ao jogador. Um exemplo deste tipo de conteúdo pode ser visto no trabalho de [Gravina e Loiacono 2015];
- **NPCs:** Normalmente são criados agentes inteligentes para interação com o jogador e até mesmo inimigos para o mesmo, um pouco sobre isso pode ser visto no trabalho de [Anand e Wei 2016];
- **Puzzles:** São focados na criação de quebra-cabeças e enigmas que precisam ser resolvidos para que as fases sejam concluídas. Um exemplo pode ser visto no trabalho de [Shaker et al 2013];
- **Jogos de tabuleiro:** Focados na criação de tabuleiros para jogos como Banco Imobiliário [Halim 2014], normalmente jogos que envolvem o uso de dados e movimentação de um tabuleiro;

1.3.2. Técnicas existentes

Entre as técnicas de PCG mais comuns temos a *search-based PCG* (SBPCG) e a *grammar-based PCG*. A SBPCG faz uso de técnicas de computação evolutiva, algoritmos

de busca e métodos similares para se buscar um conteúdo desejado. A *grammar-based PCG* pode ser utilizada para geração de plantas, geração de fases de ação e aventura entre outras aplicações. Também podemos destacar outros métodos como geração construtiva, que busca um tipo de conteúdo específico que é a geração de masmorras (*dungeons*) e fases. Entre esses podemos destacar, o *space partitioning*, *grammar-based dungeon generation*, *cellular automata*, entre outros [Shaker et al 2016]. *Answer set programming (ASP)* segue uma abordagem para programação lógica similar a *Prolog*, sendo utilizado também em gerações de labirintos e fases, como pode ser visto no trabalho de [Smith et al 2016], bem como *Noises*, *fractals*, agentes para geração de paisagens, terrenos e texturas, através de métodos clássicos como o *perlin-noise* [Korn et al 2017].

1.3.2.1 Gramática

De acordo com [Prusinkiewicz e Lindenmayer 1990], as gramáticas envolvem uma técnica de reescrita, na qual objetos complexos (fases de um jogo, por exemplo) são gerados substituindo sucessivamente partes de um objeto inicial simples (estrutural geral da fases) usando um conjunto de regras ou produções de reescrita. Portanto, uma gramática é criada como um alfabeto com diversas palavras ou símbolos e um conjunto de regras que regem que palavras ou símbolos podem ser substituídos. Como exemplo podemos considerar uma simples gramática, conforme citada abaixo:

$A \rightarrow AB$

$B \rightarrow b$

Dada essa gramática pode ser gerada uma string maior a cada vez que a gramática aplicar a regra. Levando em consideração que iniciamos uma string com a letra A teríamos: A, AB, ABb, ABbb, ABbbb, etc. Dessa forma, a medida que a regra da gramática vai sendo aplicada, a *string* final é modificada.

Na geração de conteúdo para jogos, a gramática tem sido aplicada de diversas formas diferentes. Em [Font et al 2016], é apresentado um gerador de fases através de algoritmos evolutivos baseados em gramática. Ele apresenta um método em dois estágios para a criação da fase, respeitando as restrições do projeto. A ideia geral dos autores é gerar uma gramática e evoluí-la através de um algoritmo genético e, em um segundo passo, essas fases são expandidas para produzir mapas completos.

Outra aplicação pode ser vista em [Smith e Treanor et al 2009]. Este trabalho, usa PCG em jogos de plataforma baseados em ritmos (considere ritmo algo que o jogador sente com as mãos), sendo um exemplo de combinação de gramáticas. A primeira gramática é usada para gerar ritmos e a outra para gerar a geometria do cenário com base nos ritmos gerados.

Métodos baseados em gramáticas também foram usados na geração de níveis na competição *Mario AI Competition*, na qual os autores de [Shaker e Nicolau et al. 2012] evoluíram os níveis jogáveis usando gramática evolutiva. Ela permite uma codificação simples das restrições da fase e descrições compactas para fases grandes. Um exemplo de uma versão simplificada do design de fases pode ser observado na Figura 1.3. No qual, ao iniciarmos um *level*, ele pode ser substituído por um *chunks* ou *enemy*, e consequentemente após as demais execuções substituir os *chunk* por uma *platform* ou uma *coin*, de acordo com o que o algoritmo for escolhendo.

```

<level> ::= <chunks> <enemy>
<chunks> ::= <chunk> |<chunk> <chunks>
<chunk> ::= gap(<x>, <y>, <wgbeforeafterbeforeafterbeforeaftercbeforeafterbeforeafter2 | ...
    | <box_type> (<x>, <y>)6

<box_type> ::= blockcoin | blockpowerup
    | rockcoin | rockempty

<enemy> ::= (koopas | goompas) (<x>)2 | ...
    | (koopas | goompas) (<x>)10
<x> ::= [5..95] <y> ::= [3..5]
    
```

Figura 1.3. Versão simplificada da gramática final para geração de nível para Super Mario Bros [Shaker et al 2012].

A gramática também pode ser utilizada para gerar diversos tipos de estruturas como grafos, *tilemaps*, formas em 2D ou em 3D. Assim, como abordado por [Shaker et al. 2016], pode ser utilizada uma combinação de gramática de grafos para gerar missões e de *tilemaps* para gerar um nível de jogo.

1.3.2.2 Abordagem baseada em busca

A computação evolutiva é muito utilizada em muitas soluções em PCG, sendo mais comum a *SBPCG* com a geração baseada em busca e também a gramática evolutiva quando fazem uma combinação geração de conteúdo com gramática e algoritmos evolutivos. Os algoritmos evolutivos podem ser definidos como um procedimento iterativo de busca (otimização) baseado nos mecanismos evolutivos biológicos. Assim, o algoritmo é utilizado para procurar conteúdo com as qualidades desejadas pelo desenvolvedor. Uma metáfora básica utilizada com os algoritmos evolutivos é que existe uma solução suficientemente boa para o problema dentro de algum espaço de soluções, e se continuarmos repetindo e aprimorando uma ou várias soluções possíveis, mantendo as alterações que compõem a(s) melhor(es) solução(ões) e descartando aqueles que são prejudiciais, eventualmente chegaremos a solução desejada [Shaker 2016]. O mesmo, apresenta os principais componentes da abordagem baseada em busca para resolver um problema de geração de conteúdo, sendo eles:

1. **Um algoritmo de busca:** É o principal mecanismo de um método baseado em busca. Algoritmos evolutivos relativamente simples funcionam bem, porém em casos específicos são necessários algoritmos mais sofisticados quando se quer levar em consideração alguma restrição ou algoritmos que são especializados em um conteúdo específico.

2. **Uma representação de conteúdo:** é o que será gerado, por exemplo, missões, fases, inimigos etc. A representação de conteúdo pode ser qualquer coisa, uma matriz de números reais, uma *string*. A representação de conteúdo define qual conteúdo pode ser gerado e determina se a busca é possível.
3. **Uma ou mais funções de avaliação:** Uma função de avaliação é uma função de uma parte individual de conteúdo para um número que indica a qualidade do que foi gerado. A saída de uma função de avaliação pode indicar, a jogabilidade de um nível, a complexidade de uma busca ou o apelo estético de um inimigo. Criar uma função de avaliação que indique com segurança a qualidade do jogo é uma das tarefas mais difíceis na criação de uma *SBPCG*.

A *SBPCG* foi aplicada em jogos de diferentes formas. [Loiacono et al. 2015] desenvolveu uma ferramenta de geração de espadas, o algoritmo evolucionário utilizado para gerar os parâmetros das espadas é uma versão com poucas modificações de um algoritmo genético simples, sendo o processo de seleção feito pelo usuário.

[Ferreira e Toledo 2014] fizeram uma abordagem baseada em busca para a geração de fases para o jogo *Angry Birds*. Eles usam algoritmo genético e com essa abordagem conseguiram gerar fases estáveis para o jogo.

[Dahlskog e Togelius 2014] também trabalharam com um gerador de fases, porém para jogos de plataforma. Eles fizeram o uso de micro-padrões que são “fatias” de fases de já desenvolvidas por humanos e fazendo busca por meso-padrões que são abstrações de padrões comuns vistos nas mesmas fases. Assim, esse método conseguiu gerar fases similares em estilo aos níveis com os padrões originais que foram extraídos.

1.4. Aplicação da PCG para jogos educacionais

A utilização de PCG em jogos educacionais ainda é bastante escassa, dificilmente se encontra trabalhos que faça a relação de PCG com jogos educacionais, pois grande parte desses trabalhos estão voltados para melhorias na área de inteligência artificial e no uso de jogos de entretenimento. Contudo, PCG tem muito a acrescentar para auxiliar no uso desse tipo de jogo, podendo ajudar com a adaptação do conteúdo visto pelo jogador, tornando mais fácil ou mais difícil de acordo com o desempenho do mesmo. Outra contribuição, são as possibilidades de tornar o jogo “infinito” permitindo assim a prática constante e como os conteúdos são criados proceduralmente dificilmente o aluno repetiria o mesmo conteúdo aumentando assim a variedade de conteúdos e sempre apresentando novas experiências ao jogador.

1.4.1. Desafios

Os jogos educacionais precisam ser bem definidos no momento que está sendo produzido. Uma das grandes dificuldades em produzir esses tipos de jogos está na dificuldade em balancear o entretenimento com o aprendizado que o jogo deve apresentar para o aluno. Apesar do aluno estar jogando, o jogo deve contribuir com o aprendizado do mesmo e realizar este procedimento de maneira automatizada se torna um pouco complicado. Na utilização da PCG quando voltada para os jogos educacionais é necessário ter bastante cuidado no controle do que está sendo gerado, pois caso a PCG não esteja bem planejada e definida, ela pode produzir coisas muito difíceis para o aluno ou muito fáceis e por vezes criar coisas impossíveis de serem resolvidas.

1.4.2. Estratégia

1.4.2.1. Geração em etapas

Uma das possibilidades para melhorar o uso da PCG em jogos educacionais é dividir sua criação em etapas, assim como em outros trabalhos voltados para o entretenimento, como o trabalho de [Dormans 2010]. Ele propõe uma abordagem de PCG baseada em gramática generativa para jogos de aventura com calabouços, no qual o mesmo divide a geração em duas etapas, desenvolvendo inicialmente as missões que são divididas por salas e em seguida com base nessas missões o mesmo cria uma sala para cada missão que é gerada disposta em formato de grafos. Assim, dividindo todo mapa gerado em várias missões simples.

Uma proposta que pode ser utilizada para jogos educacionais seria dividir a geração em três etapas em caso de cenários, sendo elas:

1. **Estrutura:** A geração da estrutura base da fase envolvendo chãos e paredes.
2. **Objetivos:** A inserção do objetivo pedagógico que a fase pretende abordar.
3. **Elementos Adicionais:** A adição de elementos adicionais da fase como inimigos e elementos decorativos.

O intuito dessa geração por etapas busca tornar mais livre a inserção do objetivo na fase do jogo, como somente a estrutura base do cenário é criada inicialmente todo o espaço disponível e trafegável pelo jogador pode ser utilizado para colocar o objetivo, e com um mesmo gerado a cena pode ser completada com o restante dos elementos visuais e objetos que possam atrapalhar o jogador de concluir o objetivo. Assim, priorizamos que a fase sempre tenha um objetivo para ser realizado. Uma visão geral dessa divisão em etapas pode ser vista na Figura 1.4.

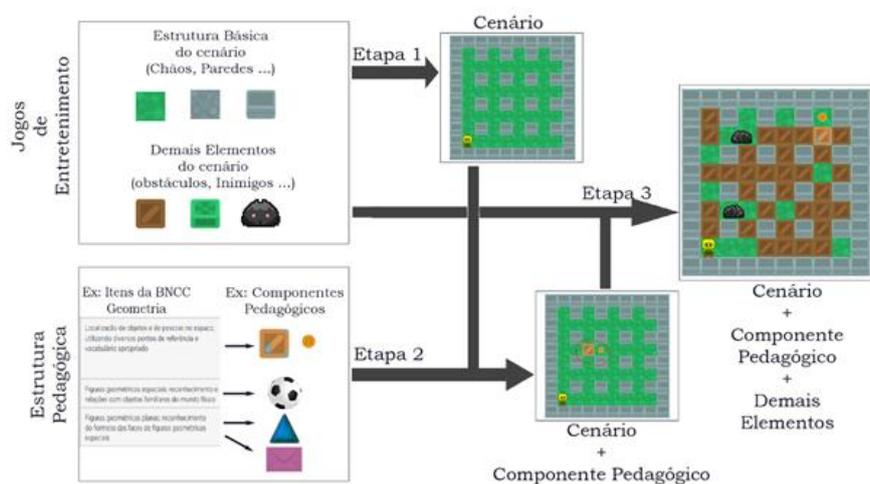


Figura 1.4. Visão geral da abordagem dividida em três etapas.

1.4.2.1.1. Modelo de jogos de entretenimento

Os jogos de entretenimento como o próprio nome sugere tem o objetivo de entreter os jogadores. A ideia por trás da abordagem visa trazer a união de mecânicas de jogos clássicos e consagrados no mercado com aspectos pedagógicos para melhorar o interesse dos estudantes pelos jogos educacionais, pois os mesmos muitas vezes são taxados como jogos desmotivantes.

Assim, estes jogos podem servir como inspiração para as gerações de fases tal como fazer o jogador passar mais tempo no mesmo. Dessa forma, propomos uma geração incremental das fases, logo para construção da mesma é feita uma divisão dos objetos que compõem o cenário por completo para assim formarmos a estrutura básica do cenário e a adição dos demais elementos do cenário.

A estrutura básica do cenário é composta pelos objetos que compõem o chão e paredes (se houver), sendo assim para gerar uma cena no qual o jogador possa se mover livremente e os objetos que compõem essa estrutura não podem fazer interação com o personagem. O objetivo dessa estrutura é criar a base para o nosso cenário e poder usar todo o espaço disponível para a adição do objetivo do jogo. Na Figura 1.4 temos o exemplo de uma estrutura base do cenário atual. Com isso temos disponível toda a área trafegável para a adição de objetivos para o jogador.

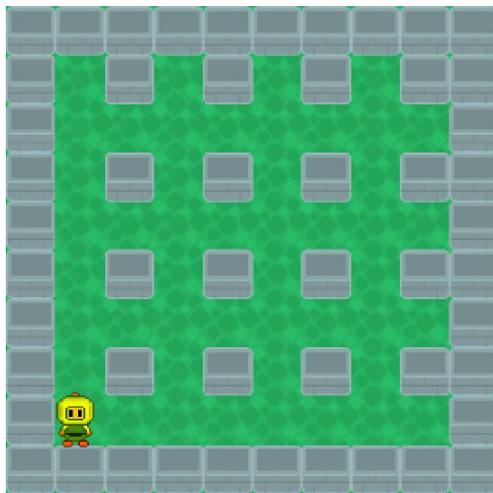


Figura 1.5. Estrutura básica do cenário do jogo.

Os demais elementos do cenário que são adicionados na última etapa da geração são elementos que podem servir para incrementar o cenário como objetos de decoração, inimigos (se houver) para atrapalhar o jogador e que estes inimigos não façam parte do objetivo do jogo. Assim, com a finalização de todas as etapas a representação da Figura 1.5 poderia ficar conforme a Figura 1.6 sendo um possível cenário após toda a geração.



Figura 1.6. Cenário completo do jogo.

1.4.2.1.2. Estrutura pedagógica

A estrutura pedagógica é composta com base nos objetivos pedagógicos, competências e habilidades curriculares, que o desenvolvedor pretende atender. Assim, o mesmo deve planejar um objetivo para o jogo que cumpra algum requisito pedagógico. Como referência de competências e habilidades o desenvolvedor pode utilizar a Base Nacional Comum Curricular (BNCC) [Ministério da educação 2017] que apresenta as competências que precisam ser aprendidas durante a educação básica. Outro documento que pode ser levado como base é o da Sociedade Brasileira de Computação (SBC) [SBC 2017] que traz referenciais para formação de computação na educação básica.

Com essas informações o desenvolvedor pode montar mecânicas e objetivos que cumpram esses requisitos e assim montar um componente pedagógico para ser adicionado ao jogo como um objetivo a ser realizado.

1.4.2.1.3. Geração da estrutura do jogo

Uma vez especificado o modelo de jogo, a primeira etapa da PCG é responsável por definir o mapeamento inicial da fase, definindo toda a área trafegável pelo personagem durante a fase, um exemplo da mesma pode ser observado na Figura 1.4. O objetivo principal desta etapa é montar a base inicial do cenário e passar a informação para as próximas etapas dos locais disponíveis para adição de componentes pedagógicos e demais elementos que formarão o cenário do jogo.

Portanto, essa etapa irá utilizar os componentes do modelo de jogo utilizados para montar a estrutura básica do cenário, chãos e paredes, o personagem principal e outros elementos obrigatórios no cenário e com alguma função específica e importante para o jogo. Uma gramática precisa ser especificada para gerar essa estrutura básica do jogo.

1.4.2.1.4. Geração dos componentes pedagógicos

Com as áreas trafegáveis definidas, próximo passo é o de inserir componentes pedagógicos, uma vez que a fase a ser gerada tem obrigatoriamente um caráter educacional. Para isso, verifica-se as habilidades selecionadas no modelo pedagógico e os chamados componentes pedagógicos que conseguem trabalhar essas habilidades no jogo.

Supondo que foi pedido para que fosse feito um objetivo pedagógico para contagem crescente dos números, então foi desenvolvido um componente pedagógico no qual o objetivo é coletar os números no cenário em ordem crescente. Nesse caso, ação que foi utilizada foi a de coletar objetos no cenário combinado com uma competência a ser praticada pelo aluno se tornando assim um objetivo para poder vencer a fase. Levando em consideração a base estrutural da Figura 1.5, nesta etapa a geração iria dispor os objetos a serem coletados no cenário nos locais disponíveis para serem colocados os objetivos, um exemplo pode ser observado na figura 1.7.

Por fim, mesmo que nesta etapa o ideal é que o objetivo seja colocado em algum lugar disponível no cenário, caso durante a geração desta etapa não encontre um local para pôr o objetivo, a mesma pode realizar modificações na estrutura básica do cenário, como ampliar os espaços disponíveis, para que o objetivo possa ser alocado. Também é possível balancear o nível de dificuldade educacional com base nas informações

disponíveis sobre o jogador. Em termos de gramática, é necessário definir as regras de inserção de cada tipo de componente pedagógico, considerando suas especificidades.

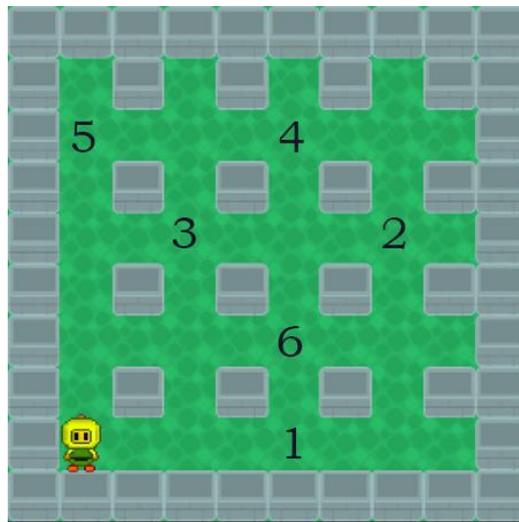


Figura 1.7. Exemplo de cenário com objetivo pedagógico.

1.4.2.1.5 Geração dos elementos adicionais

Com a alocação realizada dos componentes obrigatórios para o jogo educacional, outros componentes podem ser inseridos para se tentar garantir uma melhor atratividade para o jogo. São inseridos itens de decoração do cenário (vasos, quadros, estantes), de acordo com o modelo do jogo selecionado, além de inimigos, obstáculos, outros desafios e recompensas, características presentes em qualquer jogo de sucesso.

A PCG utilizada nesta etapa pode ainda levar em consideração as habilidades de jogo do estudante, regulando o nível de dificuldade das mecânicas de entretenimento. Esta etapa é a última, para que os elementos adicionados não interfiram na solução do objetivo pedagógico e de entretenimento proposto para que se vença a fase. Ela se torna responsável por deixar o cenário mais bonito visualmente para os jogadores, tornando assim o ambiente mais atrativo. Conforme a Figura 1.8, temos os objetos que seriam considerados como demais elementos do cenário circulos em vermelho, nesse caso as caixas e os inimigos. Com relação a gramática, é necessário definir as regras de inserção de cada tipo de componente, podendo variar de regras simples a regras mais elaboradas, envolvendo a inserção de mais de um componente em conjunto.



Figura 1.8. Exemplo de cenário com elementos adicionais.

1.4.3. Exemplos de aplicação

1.4.3.1. Bomberman

A saga Bomberman é um jogo desenvolvido pela *HudsonSoft* o qual teve seu primeiro título publicado em 1983. Ele é um jogo de estratégia no qual o personagem pode colocar bombas, explodindo caixas e inimigos que tentam impedi-lo de chegar ao ponto de saída. Dessa forma, para vencer é necessário que todos os inimigos sejam derrotados e que o mesmo deve chegar até o portal. A Figura 1.9 mostra uma fase do jogo.



Figura 1.9. Cenário do jogo Bomberman.

Neste exemplo, foi utilizada a habilidade matemática EF01MA14 (Identificar e nomear figuras planas (círculo, quadrado, retângulo e triângulo) em desenhos apresentados em diferentes disposições ou em contornos de faces de sólidos geométricos.) da BNCC [Ministério da Educação 2017]. Usando a estratégia da seção 1.4.2.1 foi proposto uma gramática para geração do cenário que atendesse aos objetivos do jogo *Bomberman* e a adição de um objetivo pedagógico no jogo, conforme a Figura 1.10.

<pre> // Para Geração do Bomberman: <ParedesX> ::= <Colisivel>^{nº colunas} ProximoY <ParedesAltX> ::= <Colisivel> (<Chao> <Colisivel>)^{nº colunas - 1} [<Colisivel>]^{nº colunas % 2} <BlocosX> ::= <Colisivel> <Espacos>^{nº colunas-1} <Colisivel> <VazioX> ::= <Colisivel> <Chao>^{nº colunas-1} <Espacos> ::= <Chao> [75%] <Bloco>[25%] <Personagem> ::= <Colisivel><Player> <Chao>^{nº colunas-2}<Colisivel> <Colisivel> ::= <Parede> <Bloco> ::= bloco ProximoX <Chao> ::= chao ProximoX <Player> ::= player <Parede> ::= parede ProximoX <Inimigos> ::= inimigos <Colunas> ::= [5...20] <Linhas> ::= [5...10] <DLevel> ::= <Espacos>^{nº espaços vazios} </pre>	<pre> //Pedagógico - Coleta <Coleta> ::= <Itens>^{Total_itens} <Itens> ::= <Quadrangulares>[25%] <Retangulares>[25%] <Triangulares> [25%] <Circulares>[25%] <Quadrangulares> ::= quadrado <Retangulares> ::= retangulo <Triangulares> ::= triangulo <Circulares> ::= circulo <Total_itens> ::= [2...5] //Preenchimento da fase <DLevel> ::= <Space>^{nº empty spaces} </pre>
---	--

Figura 1.10. Exemplo de gramática.

Com uma gramática produzida é necessário realizar uma chamada para geração da fase conforme a estratégia da seção 1.4.2.1 fazendo uso dos elementos da gramática descrita. Assim, a chamada da gramática foi realizada em três etapas, como mostra a Figura 1.11.

//Chamada da Geração da fase do Bomberman:

```

<Jogo> ::= //Estrutura
        <ParedesX><Personagem>
        (<ParedesAltX> ProximoY <VazioX> ProximoY)(Linhas-4)/2
        <ParedesAltX> ProximoY
        [<VazioX>ProximoY]Linhas%2 != 0
        <ParedesX>
        //Escolha do Objetivo Pedagógico
        <Coleta>
        //Elementos de cenário
        <DLevel>
        <Inimigos>2

```

Figura 1.11. Geração em etapas da fase do Bomberman com coleta.

Ao produzir uma chamada da gramática que siga as regras da geração é possível criar fases para o jogo. Assim, resultando na aplicação de jogo de entretenimento com a adição de um objetivo pedagógico, conforme pode ser observado na Figura 1.12. Neste caso, um objeto é escolhido aleatoriamente para ser coletado pelo jogador e o mesmo deve somente coletar o item informado, caso o contrário o mesmo perde a tentativa e inicia a fase novamente.

2. **Geração da Geometria:** A geração das plataformas e obstáculos que o jogador enfrenta durante o jogo. Nesta etapa foi utilizada a gramática para que os elementos fossem colocados no cenário. Conforme a gramática proposta na Figura 1.15.
3. **Decoração da fase:** Nesta etapa são adicionados os elementos que não causam tanto impacto no jogo além de trazer um apelo visual. No caso deste exemplo, o autor adicionou moedas aleatoriamente com base nos locais em que acontecia cada ritmo.

0	0	1	0	0	-1	-1	0	-1	1
---	---	---	---	---	----	----	---	----	---

Figura 1.14. Exemplo de ritmo gerado [Moghadam e Rafsanjani 2017].

Run	→	Simple platform bridge green platform
Jump	→	Rock angry enemy thistle
Slide	→	Cave tall tree hole

Figura 1.15. Exemplo de ritmo gerado [Moghadam e Rafsanjani 2017].

Assim, em cada etapa foi aplicada uma técnica diferente para geração dos elementos para finalizar uma fase. Após toda a execução é gerado uma fase, conforme a Figura 1.16 de uma fase de nível fácil.



Figura 1.16. Exemplo de fase gerada do jogo [Moghadam e Rafsanjani 2017].

1.5. Considerações finais

Este capítulo buscou trazer uma abordagem geral sobre o uso de geração procedural de conteúdo aplicada em jogos digitais educacionais. Foi visto que a indústria de jogos vem crescendo no decorrer dos anos, se apresentando como um mercado bastante promissor. Os jogos educacionais têm ganhado cada vez mais espaço no que consiste a sua aplicação em sala de aula por ser uma ótima ferramenta de motivação para os alunos, tornando as aulas mais agradáveis. No entanto, um dos grandes desafios encontrados no processo de concepção desse gênero de jogos se refere aos custos de produção uma vez que, em muitos casos, conta-se apenas com um orçamento muito restrito. Além disso, também resta um outro desafio relacionado a sua coerência com os objetivos de ensino-aprendizagem, de forma que estejam bem definidos para ensinar e/ou praticar um determinado conteúdo, não deixando de lado, no entanto, de engajar o jogador. Dessa

forma, são necessárias alternativas que facilitem e reduzam os custos do processo de desenvolvimento.

A geração procedural de conteúdo (PCG) é uma ótima alternativa para contribuir com o desenvolvimento desses jogos, pois os algoritmos desenvolvidos são responsáveis por gerar conteúdos que seriam custosos caso fossem desenvolvidos manualmente. Além disso, a PCG pode trabalhar com diversos tipos de conteúdo tais como fases, agentes inteligentes, histórias, entre outros.

No presente trabalho, duas técnicas foram apresentadas para auxiliar na geração de conteúdo: baseada em gramática; e baseada em busca. Ambas são bastantes utilizadas em pesquisas que envolvem a geração de conteúdo. Neste mesmo sentido, também foi apresentada uma estratégia de geração baseada em etapas, cujo cada etapa possuía uma PCG responsável pela execução.

A PCG abre margens para o uso de adaptação de conteúdo, controlando a geração e tornando os desafios adaptados ao grau de desempenho do jogador. Assim, auxilia no processo de aprendizagem do aluno, apresentando desafios adaptados tanto às suas habilidades e competências quanto ao objetivo pedagógico que se deseja alcançar. Por fim, também pode ser utilizada para a concepção de ferramentas que auxiliem o desenvolvedor a agilizar o processo de produção do jogo.

1.6. Referências

- Amato, F., Moscato, F. (2017) “Formal Procedural Content Generation in games driven by social analyses”, In: 31st International Conference on Advanced Information Networking and Applications Workshops - AINA, p. 674-679.
- Anand, B., Wei, W. H. (2016) “TITAN – Asynchronous multiplayer shooter with procedurally generated maps”, In: Entertainment Computing, Vol. 16, pp. 81-93.
- Dahlskog, S., Togelius, J. (2014) “Procedural Content Generation Using Patterns as Objectives”, In: Esparcia-Alcázar A., Mora A. (eds) Applications of Evolutionary Computation. EvoApplications 2014. Lecture Notes in Computer Science, vol 8602. Springer, Berlin, Heidelberg.
- Ferreira, L., Toledo, C. (2014) “A search-based approach for generating Angry Birds levels”, In: IEEE Conference on Computational Intelligence and Games, Dortmund, pp. 1-8.
- Font, J. M., Izquierdo, R., Manrique, D., Togelius, J. (2016) “Constrained Level Generation through grammar-based evolutionary algorithms”. In: Applications of Evolutionary Computation, pp. 558-573.
- Gravina, D., Loiacono, D. (2015) “Procedural Weapons Generation for Unreal Tournament III”, In: Games Entertainment Media Conference (GEM).
- Halim, Z. (2014) “Evolutionary Search in the Space of Rules for Creation of New Two-Player Board Games”, CoRR abs/1406.0175.
- Korn, O., Blatz, M., Rees, A., Schaal, J., Schwind, V., Gorlich, D. (2017) “Procedural Content Generation for Game Props? A Study on the Effects on User Experience”, In: Computers in Entertainment (CIE) – Theoretical and Practical Computer Applications in Entertainment, Nr. 15(2).

- Kybartas, B., Verbrugge, C. “Analysis of ReGEN as a Graph-Rewriting System for Quest Generation”, In: IEEE Transactions on Computational Intelligence and AI in Games, vol. 6, no. 2, pp. 228-242.
- Lara, I. C. M. (2003) “Jogando com a matemática de 5ª a 8ª série”. São Paulo: Rêspel.
- Loiacono, D., Mainett, R., Pirovano, M. (2015) “Volcano: An Interactive Sword Generator”, In: Games Entertainment Media Conference (GEM).
- Moghadam, A. B., Rafsanjani, M. K. (2017) “A Genetic Approach in Procedural Content Generation for Platformer Games Level Creation”, In: 2nd Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), Iran.
- Newzoo. (2018) “Newzoo Global Games Market Report 2018”, 2018. Disponível em: <https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2018-light-version/>.
- Prusinkiewicz, P., Lindenmayer, A. (1990) “The Algorithmic Beauty of Plants”, In: Springer.
- Raffe, W. L., Zambetta, F., LI, X., Stanley, K. O. (2015) “Integrated Approach to Personalized Procedural Map Generation Using Evolutionary Algorithms”, In: IEEE Transactions on Computational Intelligence and AI in Games, Vol. 7, NO. 2, pp. 139-155.
- Savi, R., Ulbricht, V. R. (2008) “Jogos Digitais Educacionais: Benefícios e Desafios”, In: Revista Novas Tecnologias na Educação - RENOTE, v.6, n.2.
- Shaker, N., Togelius, J., Nelson, M. J. (2016) “Procedural Content Generation in Games: A Textbook and an Overview of Current Research”, In: Springer. ISBN 978-3-319-42714-0.
- Shaker, M., Sarhan, M. H., Naameh, O. A., Shaker, N., Togelius, J. (2013) “Automatic generation and analysis of physics-based puzzle games”, In: IEEE Conference on Computational Intelligence in Games (CIG), Niagara Falls, ON, pp. 1-8.
- Smith, G., Treanor, M. et al. (2009) “Rhythm-Based Level Generation for 2D Platformers”, In: Proceedings of the 4th International Conference on Foundations of Digital Games. ACM, pp. 175-182.
- Stephenson, M., Renz, J. (2016) “Procedural generation of complex stable structures for angry birds levels”, In: IEEE Computational Intelligence and Games Conference (CIG).
- Togelius, J., Kastbjerg, E., Sschedl, D., Yannakakis, G. N. (2011) “What is Procedural Content Generation?: Mario on the Borderline”, In: Proceedings of the 2nd Workshop on Procedural Content Generation in Games.

Sobre os autores

Wendell Oliveira de Araújo

Doutorando em Sistemas e Computação pelo Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte (DIMAp/UFRN), mestrado em sistemas e computação pela UFRN (2018) e pós-graduado em Desenvolvimento de Jogos Digitais pela Faculdade Estácio tem experiência em Desenvolvimento de jogos (2017), atuou durante a graduação em diversos projetos de pesquisa na área de jogos pela GAMEDU. Atuou como tutor no curso técnico do Instituto MetrÓpole Digital no curso de Programação de jogos digitais no ano de 2016, bem como experiência no ensino de programação de jogos para estudantes do ensino fundamental através dos Clubes de Programação oferecidos pela GAMEDU em escolas privadas nos anos de 2016 e 2017. Atualmente realiza pesquisas principalmente no uso de games para melhoria da educação básica. Atua em parceria com outros centros de pesquisa, escolas do ensino básico e empresas de jogos digitais e de educação.

Eduardo Henrique da Silva Aranha

Doutor (2009) em Ciência da Computação pelo Centro de Informática da Universidade Federal de Pernambuco (CIn/UFPE), tem experiência como professor no ensino superior desde 2002, bem como experiência na gerência e desenvolvimento de software desde 1997. Atualmente é professor do Departamento de Informática e Matemática Aplicada da UFRN e coordena o Laboratório de Pesquisa em Games e Educação do Instituto MetrÓpole Digital da UFRN. Atualmente realiza pesquisas principalmente no uso de games para melhoria da educação básica e em engenharia de software experimental. Atua em parceria com outros centros de pesquisa, escolas do ensino básico e empresas de jogos digitais e de educação.

Charles Andrye Galvão Madeira

Possui graduação em Ciência da Computação pela Universidade Federal do Rio Grande do Norte (1998), mestrado em Ciência da Computação pela Universidade Federal de Pernambuco (2001), doutorado (2007) e pós-doutorado (2008) em Informática pela Université Paris 6 (Pierre et Marie Curie). Foi Engenheiro de Pesquisa & Desenvolvimento da empresa MASA Group (2008-2012), atuando no desenvolvimento de sistemas inteligentes para as áreas de defesa e segurança (simulação militar e gestão de crises). Atualmente é Professor Adjunto da Universidade Federal do Rio Grande do Norte, onde coordenou a criação de cursos nos níveis técnico e de graduação na área de jogos digitais, e de pós-graduação em tecnologias educacionais no Instituto MetrÓpole Digital (IMD). Tem atuado em diversos projetos de pesquisa e extensão nestas áreas, orientando dezenas de alunos de graduação e pós-graduação. Atualmente orienta mais de uma dezena de alunos de pós-graduação nos temas dos jogos digitais, tecnologias educacionais e inteligência artificial. Foi gerente adjunto da Inova MetrÓpole (2014-2015), a incubadora de empresas de base tecnológica do IMD. É coordenador do Programa de Pós-graduação em Inovação em Tecnologias Educacionais, assim como do Laboratório de Experimentação de Jogos Digitais e Realidade Virtual do IMD. Tem experiência na área de Ciência da Computação, com ênfase em Sistemas de Computação, atuando principalmente nos seguintes temas: Jogos Digitais, Jogos Sérios, Gamificação, Tecnologias Educacionais, Aprendizado de Máquina, Inovação e Empreendedorismo.

Capítulo

2

Planejamento e Execução de Estudos Secundários em Informática na Educação: Um Guia Prático Baseado em Experiências

Luis Jorge Enrique Rivero Cabrejos, Davi Viana e Rodrigo Pereira dos Santos

Abstract

Performing secondary studies is an important stage during the execution of any research. If performed systematically, secondary studies can be replicated, including new publications. This chapter presents an analysis of two types of secondary studies: Systematic Literature Reviews (SLRs), that aim to comparatively identify, evaluate and interpret all empirical studies about a research topic; and Systematic Mappings Studies (SMSs), which aim to characterize a research area. We will present the concepts and basics of these types of secondary studies, including necessary information and a description of their application processes. Furthermore, we will provide examples of these types of studies and show how to employ tools to support the execution of secondary studies.

Resumo

A realização de estudos secundários é uma das etapas mais importantes de uma pesquisa. Se conduzidos de forma sistemática, estes estudos podem ser replicados, permitindo a inclusão de novas publicações. Este capítulo apresenta uma análise de dois tipos de estudos secundários: Revisões Sistemáticas da Literatura (RSLs), que identificam, avaliam e interpretam todos os estudos experimentais sobre um tópico de pesquisa de maneira comparativa; e Mapeamentos Sistemáticos da Literatura (MSLs), que permitem caracterizar uma área de pesquisa. Considerando este tipo de estudos, serão apresentados os seus objetivos e fundamentos, incluindo os conceitos e etapas necessários para sua execução. Finalmente, este processo será exemplificado, apresentando também ferramentas de apoio.

1. Introdução

Segundo Mafra *et al.* (2006), em uma metodologia de pesquisa, a revisão da literatura permite analisar as publicações prévias dentro de um tópico de pesquisa, identificando o conhecimento disponível sobre a área investigada. Este levantamento do estado da arte é uma das etapas mais importantes na execução de uma pesquisa, servindo como base para a identificação de oportunidades de pesquisa e a geração de novas tecnologias. No entanto, devido ao aumento rápido do corpo de conhecimento em Informática na Educação, pesquisadores que têm interesse em identificar os trabalhos relevantes em um determinado domínio do conhecimento se deparam com um número extensivo de publicações, que deve ser filtrado, classificado e analisado [Magalhães *et al.* 2013]. Consequentemente, é necessário aumentar a confiabilidade dos resultados obtidos. Para atingir níveis adequados de confiabilidade, pode-se fazer uso de estudos secundários, pois evitam retrabalho e aumentam o valor dos resultados para a comunidade científica interessada no tema investigado.

Estudos secundários têm por objetivo agregar e sintetizar os resultados de estudos primários. Estudos primários, por sua vez, apresentam resultados experimentais para responder a uma questão de pesquisa [Shull *et al.* 2001]. Desta forma, ao analisar os resultados de estudos primários de forma conjunta, os estudos secundários visam identificar oportunidades de investigação dentro de um tópico de interesse, ou identificar tecnologias/intervenções que possam ser aplicadas em um determinado contexto. Kuhrmann *et al.* (2017) afirmam que os estudos secundários podem ser classificados em: (a) Revisões Sistemáticas da Literatura (RSLs), que têm por objetivo identificar, avaliar e interpretar todos os estudos experimentais relevantes a um determinado tópico de pesquisa, de maneira comparativa; e (b) Mapeamentos Sistemáticos da Literatura (MSLs), que permitem classificar os tópicos estudados dentro de uma área de pesquisa. A palavra “sistemático” indica que procedimentos metodológicos são aplicados para diminuir as ameaças à validade dos resultados decorrentes da realização destas revisões [Wohlin *et al.* 2012], permitindo: (i) identificar publicações relevantes em relação a um tema ou assunto específico; (ii) definir critérios claros para a seleção de estudos; (iii) formalizar a extração de dados advindos de cada publicação selecionada; (iv) sumarizar as evidências existentes sobre um determinado tópico; e (v) reproduzir a revisão da literatura, incluindo novas publicações.

Devido aos benefícios decorrentes da realização de estudos secundários, os pesquisadores em Informática na Educação no Brasil têm aderido à utilização de RSLs e MSLs para analisar o estado da pesquisa na área [Detroz *et al.* 2015]. Além disso, uma nova trilha de “Levantamento do estado da arte em Informática Aplicada à Educação” foi proposta no Simpósio Brasileiro de Informática na Educação (SBIE) para permitir a disseminação dos resultados de estudos secundários. No entanto, pesquisadores novatos ou com pouca experiência na realização de estudos secundários se deparam com dificuldades cujas soluções específicas não são completamente descritas nos guias disponíveis para execução de estudos secundários [Kuhrmann *et al.* 2017]. Desta forma, a condução de um estudo secundário depende altamente da experiência dos pesquisadores envolvidos. Além disso, existe a necessidade de treinamento nas

ferramentas de apoio disponíveis para o planejamento, execução e análise dos resultados dos estudos secundários [Tell *et al.* 2016].

O conhecimento sobre o processo de aplicação de estudos secundários tem grande relevância para a comunidade de Informática na Educação por propiciar o contato com um tópico atual e amplamente explorado pela comunidade acadêmica. Contudo, a condução de um estudo secundário depende altamente da experiência dos pesquisadores envolvidos e existe a necessidade de treinamento nas ferramentas de apoio disponíveis para o planejamento, execução e análise dos resultados deste tipo de estudos [Tell *et al.* 2016]. Logo, sedimentar um ferramental conceitual e prático para o planejamento e execução de estudos secundários é de grande importância para a comunidade de Informática na Educação.

Diante do exposto, este capítulo tem por objetivo prover conhecimentos, discussões e reflexões, de forma didática, sobre a teoria e prática necessárias para a execução de estudos secundários. São descritas práticas e expostos exemplos aplicados que tenham potencial de despertar o interesse de acadêmicos e profissionais com relação à realização de estudos secundários, de modo a evitar ameaças à validade dos seus resultados. Este material serve de referência para alunos de graduação e pós-graduação em Informática na Educação, cobrindo um tópico relevante (condução de estudos secundários), facilitando a disseminação do conhecimento e a consolidação das pesquisas em Informática na Educação. Espera-se com isso: (a) apresentar uma visão geral e formativa do processo de execução de Estudos Secundários em Informática na Educação, fornecendo exemplos de aplicação de RSLs e MSLs para responder questões de pesquisas em desenvolvimento da área; (b) contribuir para entender, analisar e resolver problemas enfrentados por pesquisadores pelo fato de que este conceito vem sendo extensamente utilizado na academia; e (c) atrair estudantes de graduação e pós-graduação da área de Computação, além de professores e pesquisadores na condução de estudos secundários, através de exemplos de ferramentas e protocolos de pesquisa.

O restante deste capítulo está organizado como segue. A Seção 2 apresenta uma visão geral sobre estudos experimentais que podem ser aplicados em Informática na Educação, diferenciando RSLs de MSLs e indicando estudos secundários realizados na área de Informática na Educação. A Seção 3 apresenta a metodologia para a realização de estudos secundários e exemplos destas etapas sendo realizadas em uma pesquisa na área de Informática na Educação. Por sua vez, a Seção 4 apresenta exemplos de ferramentas de apoio ao processo de condução de estudos secundários, mostrando o passo a passo de sua aplicação. A Seção 5 apresenta um exemplo de revisão sistemática desenvolvida recentemente na área. Finalmente, a Seção 6 apresenta as conclusões deste capítulo de livro e os desafios futuros para a realização de MSLs e RSLs em Informática na Educação.

2. Fundamentação Teórica Sobre Estudos Secundários em Informática na Educação

2.1. Caracterizando Estudos Secundários

Segundo Wohlin *et al.* (2012), existem vários tipos de estudos experimentais que podem ser realizados em uma pesquisa científica. Estes estudos podem ser classificados em: primários, secundários e terciários. Estudos primários são estudos experimentais cujo objetivo é caracterizar, avaliar, prever, controlar ou melhorar produtos, processos, recursos, modelos, teorias, entre outros [Travassos *et al.* 2002]. Nesse contexto, estudos primários são a fonte base para se obter resultados sobre uma determinada tecnologia e seu potencial uso para resolver um problema. Shull *et al.* (2001) afirmam que existem vários tipos de estudos primários que podem ser realizados dependendo do objetivo a ser alcançado durante uma metodologia de pesquisa. A Tabela 1.1 apresenta uma descrição de alguns dos tipos de estudos que podem ser realizados e seu propósito.

Tabela 1.1 – Tipos de estudos experimentais (por objetivo) [Shull *et al.* 2001]

Estudo	Pergunta	Propósito
Estudo de Viabilidade	Os resultados da tecnologia são viáveis e o tempo bem empregado?	Determinar se é possível viabilizar o uso da tecnologia.
Estudo de Observação	Os passos para aplicar a tecnologia fazem sentido?	Observar a tecnologia a fim de aprimorar o entendimento dos pesquisadores em relação à aplicação da mesma e possibilitar seu refinamento.
Estudo de Caso (ciclo de vida)	A tecnologia é adequada ao ciclo de vida real?	Analisar a tecnologia em um determinado contexto de ciclo de vida e para caracterizar a aplicação da mesma.
Estudo de Caso (indústria)	A tecnologia é adequada ao ambiente industrial?	Identificar se existem problemas de integração na aplicação da tecnologia proposta em um ambiente industrial.

Estudos primários apresentam as evidências que são usadas como base para estudos secundários e terciários. Estudos secundários têm por objetivo coletar os resultados de estudos primários de modo a analisar os dados de forma conjunta [Wohlin *et al.* 2012]. De modo geral, estudos secundários permitem analisar e discutir os resultados de estudos primários e agregá-los. Com isso, estudos secundários se tornam a base para qualquer pesquisa, permitindo ao identificar oportunidades de pesquisa dentro de uma determinada área e evitar o retrabalho [Mafra *et al.* 2006].

As questões de pesquisa que um estudo secundário permite responder são [Kuhmann *et al.* 2017]: (a) Quais/quantas publicações em um determinado contexto são publicadas? (b) Quais/quantas publicações em um determinado contexto são publicadas por ano? (c) Qual é o nível de maturidade científica das publicações? (d) Qual é a contribuição das publicações? (e) Qual é o comportamento da área

considerando o fluxo de publicações? e (f) Quais novas abordagens existem com relação ao tópico de interesse?. Para responder a estas questões, dois tipos de estudos secundários podem ser realizados: Revisões Sistemáticas da Literatura (RSLs) e (b) Mapeamentos Sistemáticos da Literatura (MSLs). Como descrito anteriormente, a principal diferença entre estes estudos é que RSLs realizam comparações dos resultados de estudos primários, enquanto MSLs caracterizam uma determinada área do conhecimento, identificando todos os resultados retornados para esta área sem a necessidade de fazer um comparativo entre os mesmos.

Em determinadas situações, mais do que um estudo secundário (RSL ou MSL) pode ser publicado dentro de uma mesma área, pois existe a necessidade de incluir novos resultados não identificados por um estudo secundário realizado até um determinado período de tempo ou responder novas questões de pesquisa. Nesse contexto, estudos terciários surgem para agregar o conhecimento de um ou mais estudos secundários [Cruzes e Dybå 2011]. Com isso, estudos terciários permitem analisar e interpretar os resultados relevantes dentro de uma área de pesquisa, tendo os mesmos benefícios de estudos secundários.

Apesar da importância de estudos primários para a identificação de resultados relevantes para a área de Informática na Educação, o risco de replicar soluções ou cometer os enganos com relação às tecnologias propostas pode ser evitado ao se realizar estudos secundários. Ao aplicar estudos secundários, um pesquisador pode utilizar os resultados já publicados dentro da área investigada para minimizar eventuais dificuldades e incertezas que povoam o processo de definição de uma nova tecnologia em Informática na Educação. Com o intuito de explicar e exemplificar o processo de realização de estudos secundários, a seção a seguir apresenta, resumidamente, o processo de planejamento, execução e publicação dos resultados de RSLs e MSLs, que pode ser aplicado para a realização de pesquisas em Informática na Educação.

2.2. Estudos Secundários em Informática na Educação

Na área de Informática na Educação há diversas publicações relatando a execução e resultados de estudos secundários. Na edição de 2017, o Simpósio Brasileiro de Informática na Educação (SBIE) apresentou uma trilha somente de trabalhos de estudos secundários. Um dos trabalhos desta trilha é relatado por Schiehl e Gasparini (2017). Neste trabalho, os autores apresentam a condução de um mapeamento sistemático da literatura onde, após o processo de seleção, identificaram 32 trabalhos que tratam sobre o ensino híbrido utilizado em escolas públicas no Brasil. Os autores utilizam sete bibliotecas digitais e identificaram que dentre os modelos existentes, o modelo suplementar foi o mais utilizado no ensino híbrido, pois complementa o ensino tradicional de sala como ferramentas on-line.

Um exemplo de revisão sistemática da literatura é apresentado por Connolly *et al.* (2012). Neste estudo secundário, os autores buscam analisar as evidências empíricas sobre jogos de computador e jogos sérios. Os autores justificam a escolha das bibliotecas digitais levando em consideração a relevância das bases para educação, tecnologia da informação e ciências sociais. Assim como no exemplo anterior, os autores definiram a *string* de busca e os critérios de inclusão de publicações na revisão.

Na etapa de análise de dados dos estudos primários levantados, os autores aplicam uma estratégia de codificação dos artigos, isto é, uma categorização dos jogos em termos de: tipo de jogo (digital ou não digital); propósito primário do jogo; classificação de gênero do jogo; e plataforma do jogo. Além disso, os autores também categorizam os efeitos dos jogos. Por ser bem mais rígida que um mapeamento sistemático, os autores também realizam uma classificação de qualidade dos estudos selecionados na revisão sistemática. Esta revisão sistemática foi publicada no periódico *Computers & Education*, um importante veículo da área de Informática na Educação.

Com relação a estudos terciários, Detroz *et al.* (2015) analisaram artigos reportando estudos secundários publicados em veículos de publicação brasileiros na área de Informática na Educação. Os autores fizeram uma busca nos seguintes veículos: Simpósio Brasileiro de Informática na Educação, Workshop sobre Educação em Computação, Workshop de Informática na Escola, Workshop de Desafios da Computação Aplicada à Educação, Revista Brasileira de Informática na Educação, Revista de Informática Teórica e Aplicada, Revista Novas Tecnologias na Educação, e Revista Informática na Educação: Teoria & Prática. Neste trabalho, os autores conseguiram mapear revisões da literatura até o ano de 2012. Ao todo, foram identificados 42 artigos reportando estudos secundários em informática na educação. Estes estudos não eram necessariamente estudos seguindo uma metodologia sistemática, ou seja, as publicações selecionadas podiam conter relatos de revisões da literatura executadas sem rigor e irreproduzíveis. Como resultados, observou-se que o evento DesafIE apresenta o maior percentual de revisões da literatura, com 4% das duas publicações sendo para este fim. Os temas abordados pelas revisões sistemáticas identificadas na pesquisa de Detroz *et al.* (2015) foram classificados em: aspectos relativos ao desenvolvimento de software educacional; uso de jogos na educação; ensino na área de computação; e Ambientes Virtuais de Aprendizagem.

Com relação ao rigor das revisões sistemáticas da literatura realizadas na área de Informática na Educação, Detroz *et al.* (2015) identificaram que não existe um consenso entre os artigos apresentando revisões sistemáticas quanto ao seguimento de um protocolo de revisão. Nesse sentido, o presente capítulo visa apresentar as etapas de execução de estudos secundários para ser usado como base na condução de futuros estudos. A ideia principal é motivar pesquisadores a analisar e mitigar as ameaças à validade decorrentes da execução deste tipo de estudos. Com isso, espera-se contribuir com a qualidade de futuras publicações relatando RSLs e MSLs na área de Informática na Educação e contribuir com um material atualizado para que sirva de referência para estudantes, pesquisadores e profissionais que necessitem executar estes estudos secundários em suas atividades profissionais. A seguir, o processo é explanado, apresentando um exemplo prático.

3. Processo de Aplicação de Estudos Secundários

Segundo Mafra *et al.* (2006), o processo de caracterização de evidências por meio de RSLs e MSLs deve ser formalizado com o objetivo de se identificar e caracterizar eventuais resultados experimentais existentes relacionados à área de pesquisa a ser investigada. A formalização desse processo permite que essa etapa seja repetível e passível de ser melhorada. Nesse contexto, sugere-se que um estudo secundário seja

dividido nas seguintes etapas: planejamento, execução e reporte dos resultados, descritos a seguir [Wohlin *et al.* 2012].

Para exemplificar o processo de condução de um estudo secundário, considere o seguinte contexto: *“Uma equipe de desenvolvimento de software tem interesse em identificar técnicas para avaliar a experiência do usuário (User eXperience - UX) de jogos educacionais. A UX considera os aspectos de facilidade de uso e respostas emocionais relacionadas ao uso de um sistema ou serviço. Nesse sentido, uma UX positiva é vital para aumentar o engajamento dos usuários de jogos educacionais. Para identificar de forma exaustiva todas as possíveis técnicas propostas para este contexto e assim escolher uma técnica adequada ao ciclo de vida adotado pela equipe, a mesma decidiu realizar um estudo secundário.”* Este contexto é utilizado para exemplificar cada uma das etapas do processo de condução de estudos secundários, instanciando as atividades relacionadas a cada etapa. Na Seção 4, este mesmo exemplo é utilizado como base para instanciar o uso de ferramentas computacionais para o apoio ao processo de condução de estudos secundários.

3.1. Planejamento

O planejamento é a etapa inicial de um estudo secundário [Nakagawa *et al.* 2017]. Essa etapa permite o estabelecimento de um protocolo de revisão, que define os passos para seleção dos estudos primários relevantes, além de definir como e quais dados serão extraídos no decorrer da análise dos estudos primários, assim como as questões de pesquisa que serão respondidas com estes dados. Para planejar um estudo secundário, as atividades a seguir devem ser realizadas.

3.1.1. Identificação da Necessidade da Revisão

A necessidade de um estudo secundário é originada pelo objetivo do pesquisador de entender o estado da arte de uma área, ou de profissionais interessados em identificar evidências que possam ser utilizadas na tomada de decisões estratégicas. Se outros estudos secundários já existirem, deve-se avaliar se os mesmos respondem às questões de pesquisa estipuladas, ou se é necessário estendê-los ou, ainda, realizar novos estudos secundários.

Considerando o exemplo da revisão quanto a técnicas de avaliação da experiência do usuário de jogos educacionais, os proponentes da revisão podem consultar bibliotecas digitais para identificar se não existe uma revisão da literatura com este fim que já foi publicada. Para identificar trabalhos publicados nesta linha, os pesquisadores/profissionais podem pesquisar por termos que denotem revisões na área de interesse. Por exemplo, podem ser usadas palavras chave como: *“User Experience”*, *“Evaluation Techniques”*, *“Educational Games”* e *“Literature Review”*. Vale ressaltar que uma busca por termos em inglês pode trazer mais resultados, devido a que existem mais publicações escritas em Inglês e veículos de publicação internacionais utilizam este idioma como padrão para disseminação de resultados de pesquisa. Mais detalhes sobre a construção de um conjunto de termos (*string*) de busca apropriada para o uso em bibliotecas digitais são apresentados na Seção 3.1.4.

Uma busca inicial pelos termos citados acima em bibliotecas digitais como o Google Acadêmico, Scopus, IEEE Xplore, entre outras, permite identificar trabalhos na área [Nacke *et al.* 2010] [Bellotti *et al.* 2013] [Rivero e Conte 2017]. Estes trabalhos devem ser lidos e analisados com cuidado, pois podem servir para responder a questão de pesquisa proposta pelos pesquisadores/profissionais e evitar retrabalho na realização de um estudo secundário já disponível. Por exemplo, o trabalho de Nacke *et al.* (2010) apresenta uma análise de vários métodos que podem ser utilizados para avaliar a experiência do *gameplay* em jogos educacionais, listando técnicas como: *eye tracking*, modelagem de personas, avaliação de métricas de comportamento em jogos, entrevistas e questionários, etnografia, entre outras. Apesar do tema estar relacionado, existem alguns pontos que devem ser considerados antes de adotar esta revisão como base para a pesquisa dos proponentes da revisão. Por exemplo: (a) a listagem dos métodos indicada pelos autores não deriva de um estudo secundário sistemático e, portanto, pode estar incompleta; (b) a descrição dos métodos é vaga e não existe um relatório técnico para ser usado como base para consultar o processo de aplicação, caso o objetivo da realização da revisão seja adotar uma das técnicas identificadas; e (c) a revisão foi publicada em 2010 e pode estar defasada. Considerando ainda, que a revisão publicada por Nacke *et al.* (2010) não segue uma abordagem sistemática, a mesma não pode ser replicada e/ou atualizada, pois não se tem informações sobre que fonte de informações foi utilizada para identificar os estudos primários, qual o período de consulta, qual a forma de identificação dos estudos primários e seu processo de seleção etc. Desta forma, os pesquisadores/profissionais interessados em identificar técnicas de avaliação da experiência do usuário em jogos educacionais podem se questionar quanto à adequação da utilização deste trabalho como fonte única para a identificação de técnicas para sua pesquisa.

O trabalho de Bellotti *et al.* (2013) apresenta similaridades quanto ao trabalho de Nacke *et al.* (2010). Embora tenha sido publicado três anos depois, este trabalho também não seguiu uma abordagem sistemática para a identificação de estudos primários. Além disso, a quantidade de técnicas descritas pelos autores é menor que o número de técnicas descritas por Nacke *et al.* (2010). No entanto, a descrição destas técnicas é mais detalhada que a descrição proposta por Nacke *et al.* (2010) e os autores apresentam uma listagem de ferramentas de apoio ferramental para a aplicação das mesmas. Nesse contexto, os pesquisadores/profissionais interessados em identificar técnicas de avaliação da experiência do usuário devem analisar se os resultados propostos nesta revisão são suficientes para servir como base para sua pesquisa. Além disso, apesar da listagem de ferramentas de apoio servir como base para aplicar as técnicas de avaliação de UX, deve-se considerar que os proponentes do estudo secundário não definiram que possuir apoio ferramental é um critério de escolha das técnicas. Em ocasiões, um pesquisador/profissional pode se deparar com novas questões que não foram definidas durante o planejamento da revisão e decidir mudar o rumo da pesquisa. Contudo, o risco de a revisão não responder à questão de pesquisa inicial proposta pode aumentar, visto que o pesquisador/profissional não definiu um protocolo de pesquisa. As próximas seções do capítulo discorrem quanto a como formalizar o processo de identificação, seleção e extração de estudos primários de forma a responder

às questões de pesquisa propostas pelos pesquisadores/profissionais ao conduzir um estudo secundário.

Finalmente, o trabalho de Rivero e Conte (2017) apresenta uma revisão sistemática da literatura que é focada em métodos de avaliação da experiência do usuário em qualquer tipo de aplicação. Ou seja, esta revisão é mais abrangente e contém técnicas não aplicáveis à avaliação de jogos educacionais. Nesse contexto, os pesquisadores/profissionais interessados na identificação de técnicas de avaliação de UX para jogos educacionais devem decidir se vale a pena ou não filtrar os estudos secundários relacionados apenas com a avaliação de jogos educacionais. Embora seja necessário fazer este trabalho, os autores poderiam economizar tempo em refazer uma busca por técnicas de avaliação de UX apenas para o nicho de jogos educacionais. Além disso, embora a revisão de Rivero e Conte (2017) considere apenas trabalhos publicados até o ano de 2015, devido a que esta revisão foi realizada seguindo uma metodologia sistemática, os pesquisadores/profissionais podem atualizar este trabalho, identificando novas técnicas que tenham sido propostas desde então.

Considerando que não foi possível identificar um estudo secundário na área de técnicas de avaliação de UX para jogos educacionais (assumindo como exemplo os trabalhos listados neste capítulo de livro), os pesquisadores/profissionais interessados na identificação de técnicas de avaliação de UX de jogos educacionais podem considerar a execução de um novo estudo secundário. As subseções a seguir fornecem subsídios para a realização das etapas subsequentes do processo de condução de um estudo secundário.

3.1.2. Identificação do Tipo de Estudo Secundário: RSL ou MSL

Uma vez identificada a necessidade de realização de um novo estudo secundário, deve-se decidir qual é o tipo de estudo a ser realizado. Se o objetivo é caracterizar o estado da arte, independentemente dos resultados obtidos nos estudos primários analisados, pode ser conduzido um MSL. Por outro lado, se o objetivo é realizar uma análise dos resultados obtidos nos estudos primários, agregá-los e compará-los, uma RSL será mais adequada. O tipo de estudo deve ser definido com base nos objetivos dos pesquisadores e profissionais envolvidos na execução do estudo secundário.

O processo de execução de um MSL é similar ao de uma RSL. No entanto, uma RSL pode incluir critérios de qualidade mais rigorosos que os propostos em MSLs para a avaliação dos artigos retornados, visto que os mesmos são mais abrangentes que RSLs. A qualidade de um estudo secundário depende da qualidade dos estudos primários incluídos [Nakagawa *et al.* 2017]. Os critérios de qualidade são perguntas sobre aspectos metodológicos dos estudos primários selecionados que permitem analisar mais detalhadamente os estudos. Alguns critérios de qualidade em RSLs podem ser encontrados nos trabalhos de Imtiaz *et al.* (2013) e Usman *et al.* (2014).

No exemplo proposto neste capítulo, considerando que o objetivo do estudo secundário é a identificação dos possíveis estudos primários relatando técnicas de avaliação de UX de jogos educacionais e que não se pretende fazer um comparativo dos resultados obtidos por estas técnicas, pode-se considerar que os profissionais almejam realizar um MSL. Nesse contexto, não será feito um comparativo das técnicas

identificadas e sim um levantamento de quais técnicas existem para o contexto de avaliação de UX em jogos educacionais.

3.1.3. Definição da Questão de Pesquisa

Um estudo secundário foca na identificação de estudos primários, extração de dados destes estudos e sua análise. Nesse contexto, a questão de pesquisa deve ser formulada de forma clara o suficiente para guiar o processo de busca e extração. Alguns aspectos que podem ser considerados durante a definição de uma questão de pesquisa são [Wohlin *et al.* 2012]: (a) qual é a população que será investigada? (b) qual é a intervenção que será aplicada nos estudos primários identificados? (c) que tipo de comparação será considerada? (d) como os resultados se relacionam com a questão de pesquisa? e (e) qual é o contexto dos estudos primários que serão considerados? Estas informações serão utilizadas como base na definição do protocolo de revisão detalhado a seguir.

3.1.4. Definição do Protocolo

O protocolo de revisão é o documento que descreve os procedimentos para realização do estudo secundário. Este documento deve conter algumas informações como [Kitchenham e Charters 2007]: (a) fundamentação e motivação; (b) questões de pesquisa; (c) estratégia de busca, ou seja, onde (bibliotecas digitais, anais etc.) e em que período de tempo será realizada a busca das publicações a serem incluídas no estudo secundário; (d) critérios para incluir ou excluir uma publicação, segundo os resultados que o mesmo pode fornecer para responder as questões de pesquisa do estudo secundário; (e) avaliação da qualidade da publicação considerada; (f) estratégia para extração de dados, (g) processo de síntese dos dados coletados; e (h) como os resultados identificados serão reportados. Estas informações são descritas a seguir, com exceção dos itens (g) e (h), que são descritos na subseção 3.2.

A descrição da **fundamentação e motivação** tem por objetivo fornecer uma introdução para pesquisadores/profissionais que forem utilizar o estudo secundário como base em alguma pesquisa futura. Além do título, informações sobre os pesquisadores/profissionais envolvidos no processo de condução do estudo secundário e outras informações relevantes, os autores devem fornecer uma explicação sobre o contexto e justificativa para a realização do estudo secundário (ver Seção 3.1.1.). Além disso, os autores do estudo secundário devem indicar os objetivos a serem alcançados com a revisão da literatura. Estas informações servirão para ajudar a entender melhor o contexto associado ao estudo secundário, facilitando também o entendimento da análise dos dados obtidos [Nakagawa *et al.* 2017].

A **definição das questões de pesquisa** permite guiar a condução de estudos secundários, pois as mesmas definem o que deve ser respondido quanto ao tópico de pesquisa abordado. Dentro de um estudo secundário, questões de pesquisa podem ser primárias ou secundárias [Kitchenham e Charters 2007]. A questão de pesquisa primária ou principal define o tópico de pesquisa que está sendo investigado. Já as questões secundárias podem ser uma ou mais questões que definem características específicas do tópico investigado. Ao definir uma questão de pesquisa primária, a mesma pode ser

estruturada seguindo os critérios PICO [Kitchenham e Charters 2007]: (P) População, um papel/tipo de sistema/tipo de paradigma que será observado; (I) Intervenção, método ou procedimento que será investigado; (C) Comparação, método ou procedimento com que será feito o comparativo; e (O) Resultados/Outcome, é o efeito que será investigado. No exemplo deste capítulo, a questão de pesquisa principal (QP) pode ser organizada da seguinte forma: “Quais técnicas (I) de avaliação de UX (O) existem para avaliar jogos educacionais (P)?” Note que, neste exemplo, como o estudo secundário é um MSL, não existe um componente C (Comparação) do PICO, visto que os pesquisadores/profissionais não tem interesse em comparar o resultado obtido nos estudos primários reportando técnicas de avaliação de UX e sim, identifica-las.

Ao definir a questão principal de pesquisa, as questões secundárias visam fornecer subsídios para entender as especificidades do tópico de pesquisa estudado. Ou seja, uma questão de pesquisa secundária permite guiar a coleta de dados para identificar detalhes relacionados à questão de pesquisa principal. No contexto da QP, os autores podem formular questões relacionadas às características das técnicas identificadas, como por exemplo: (a) Qual é a base teórica da técnica proposta? (b) Que aspectos a técnica permite avaliar? (c) Qual é o processo de aplicação da técnica? (d) Em que etapa do processo de desenvolvimento a técnica pode ser aplicada? (e) Que tipos de jogos educacionais a técnica permite avaliar? (h) A técnica possui apoio ferramental para automatizar parte do seu processo? Se sim, qual e como funciona? (i) Quais são as vantagens de uso da técnica proposta? (j) Quais são as desvantagens do uso da técnica proposta? Estas questões são utilizadas no momento de definir a estratégia de extração de dados.

Uma vez definidas as questões de pesquisa principal e secundárias, os autores do estudo secundário devem definir **de que forma será feita a busca pelos estudos primários**. Para este fim, deve se determinar: (a) que termos serão usados para encontrar as publicações relacionadas; (b) qual é o período de publicação dos artigos a serem analisados durante a revisão; e (c) em que veículos de publicação serão aplicados os critérios de busca definidos. Para definir os termos que serão aplicados nas bases digitais, os autores do estudo secundário podem se basear na organização do PICO descrita acima. Desta forma, os pesquisadores/profissionais podem utilizar publicações relacionadas com a questão de pesquisa como base para definir os sinônimos dos termos identificados no PICO. Estes artigos base são publicações que tratam do tema a ser abordado pelo estudo secundário e são considerados de alto impacto na área de interesse dos pesquisadores/profissionais. Os autores podem explorar o título, resumo e palavras chaves destas publicações e procurar por sinônimos dos termos definidos inicialmente no PICO. A Tabela 1.2 apresenta o processo de identificação de termos relacionados para o exemplo utilizado neste capítulo de livro. Estes termos foram procurados em inglês para que artigos de diversos veículos de publicação (nacionais e internacionais) possam ser retornados. Adicionalmente, é possível que os pesquisadores/profissionais conduzindo o estudo secundário utilizem termos em outras línguas (e.g. Português, Espanhol etc.) para aumentar a quantidade de artigos retornados que foram publicados em outros veículos de publicação.

No momento da construção da *string* de busca formada pelos termos selecionados, os pesquisadores/profissionais devem relacionar os termos a partir de expressões booleanas (AND, OR, NOT, entre outras). Para o exemplo deste capítulo, os autores podem usar a seguinte *string* de busca:

((“Game” OR “Simulation”) AND (“Education” OR “Instructional” OR “Learning” OR “Serious” OR “Teaching” OR “Training”) AND (“Approach” OR “Framework” OR “Instrument” OR “Method” OR “Model” OR “Technique”) AND (“Assessment” OR “Measurement” OR “Evaluation” OR “Testing” OR “Recognition” OR “Tracking”) AND (“User experience” OR “UX”))

Adicionalmente, é necessário avaliar se utilizar o radical de um termo de pesquisa pode melhorar a quantidade de artigos relacionados de uma *string* de busca ou permitir a inclusão de publicações não desejadas. Por exemplo, uma publicação relatando um método de avaliação de UX de jogos educacionais pode não usar o termo “Evaluation” no seu texto, e sim o termo “Evaluating”, portanto, existe a possibilidade deste artigo não ser retornado ao aplicar a *string* de busca definida acima. O termo “Evaluat” poderia ser usado no lugar, o que permitiria identificar publicações contendo o termo “Evaluation” e “Evaluating”. Além disso, em algumas ocasiões, o uso de termos de forma separada pode fazer com que artigos não relacionados retornem aumento de forma considerável o trabalho de seleção de artigos.

Tabela 1.2 – Identificação de termos aplicando os critérios PICO

Critério	Termo Base	Sinônimos	Artigos Base
	Jogo	Game Simulation	Cascini e Campos (2015) Battistela <i>et al.</i> (2014)
População (P)	Educacional	Education Instructional Learning Serious Teaching Training	Cascini e Campos (2015) Battistela <i>et al.</i> (2014)
Intervenção (I)	Técnica	Approach Framework Instrument Method Model Technique	Bargas-Avila and Hornbæk (2011) Rivero e Conte (2017) Vermeeren <i>et al.</i> (2010)

Comparação (C)	Não aplicável	Não aplicável	Não aplicável
Resultado (O)	Avaliação	Assessment	
		Measurement	Bargas-Avila and Hornbæk (2011)
		Evaluation	
		Testing	Rivero e Conte (2017)
		Recognition	Vermeeren <i>et al.</i> (2010)
		Tracking	
	User Experience	User experience	Bargas-Avila and Hornbæk (2011)
		UX	Vermeeren <i>et al.</i> (2010)

No exemplo do estudo secundário sobre técnicas de avaliação de UX para jogos educacionais, ao invés de usar o termo “User Experience”, os pesquisadores/profissionais podem usar o termo “User” AND “Experience”. No entanto, esta mudança, pode fazer com que pesquisas como a de Iosup e Epema (2014) sejam retornadas. O trabalho de Iosup e Epema (2014) trata de um relato de experiência sobre o uso de gamificação na educação técnica. Apesar de conter os termos: “User”, “Experience”, “Evaluation”, “Game” e “Education”; o trabalho não reporta uma técnica que possa ser utilizada pelos pesquisadores/profissionais em sua pesquisa. Portanto, uma análise deve ser realizada pelos autores, verificando o impacto das decisões no momento de (des)agrupar os termos da pesquisa e utilizar os radicais dos mesmos.

Durante a definição da *string* de busca, os autores devem fazer testes nas bibliotecas digitais de modo a avaliar se ela permite encontrar artigos de controle. Um artigo de controle é um artigo que, antes de iniciar a condução do estudo secundário, foi definido como relacionado para responder à questão de pesquisa principal. Por exemplo, no contexto da identificação de métodos de avaliação de UX para jogos educacionais, o artigo de Law *et al.* (2012) pode ser considerado um artigo de controle. Ou seja, no momento de aplicar a *string* de busca, o artigo deve ser retornado, pois trata do foco de pesquisa do estudo secundário. A definição de um conjunto de artigos de controle pode ajudar os pesquisadores/profissionais a avaliar a eficácia da *string* de busca para identificar publicações relatando estudos primários relacionados com o objetivo de pesquisa.

Com relação à definição do período de inclusão de estudos primários, os pesquisadores devem definir também se existe um critério para incluir artigos dependendo da data de publicação dos mesmos. Por exemplo, dependendo do tema de pesquisa, publicações podem ser consideradas desatualizadas, ou não relevantes para o tópico de pesquisa; ou pode existir um marco que impacte a pesquisa realizada no tema. Por exemplo, em caso de os pesquisadores/profissionais estarem interessados em identificar tecnologias propostas para o contexto de desenvolvimento ágil, pode ser

limitado o período de pesquisa a partir de 2001, que é quando o manifesto ágil foi publicado [Fowler e Highsmith 2001]. Esta decisão deve ser tomada considerando o conhecimento sobre a área de pesquisa, ou procurar por fontes que justifiquem a tomada de decisões.

Finalmente, com relação à definição das bases onde serão pesquisados os trabalhos relatando estudos primários, os pesquisadores/profissionais devem considerar veículos de publicação segundo a especificidade do estudo secundário sendo realizado. Por exemplo, no Brasil, existem vários veículos de publicação na área de Informática na Educação, conforme apresentado na Tabela 1.3. No entanto, podem existir dificuldades associadas ao tema foco destes veículos além de dificuldade de aplicar *strings* de busca, sendo necessário realizar buscas manuais. Com relação a bibliotecas digitais internacionais, é possível acessar [Kitchenham *et al.* 2016]: Springer, ACM, IEEE eXplore e Scopus. Estas bibliotecas: (a) permitem o uso de expressões lógicas para as buscas ou mecanismo similar para localizar as publicações através do título e resumo das publicações; (b) possuem um bom funcionamento e abrangência de suas máquinas de busca; e (c) contêm em suas bases, publicações de variadas áreas de conhecimento.

Após a aplicação da *string* de busca, um conjunto de publicação deve ser retornado das bases digitais e/ou veículos de publicação selecionados onde será feita a busca de forma manual. Portanto, existe a possibilidade de que artigos não relacionados com o objetivo do estudo secundário sejam retornados. Para filtrar estes artigos e analisar os estudos primários que respondam à questão principal de pesquisa da revisão da literatura, é necessário **definir critérios para incluir ou excluir uma publicação**. Estes critérios permitem que o processo de seleção de artigos seja objetivo e documentado, uma vez que para justificar a inclusão ou exclusão de uma publicação, um (ou um conjunto de) critério(s) deve(m) ser associado(s) à publicação.

Tabela 1.3 – Lista de veículos de publicação relacionados à área de Informática na Educação [Detroz *et al.* 2015]

Nome do Veículo	Endereço Eletrônico
Revista Brasileira de Informática na Educação	http://www.br-ie.org/pub/index.php/rbie
Revista de Informática Teórica e Aplicada	http://seer.ufrgs.br/rita
Revista Informática na Educação: Teoria & Prática	http://seer.ufrgs.br/InfEducTeoriaPratica
Revista Novas Tecnologias na Educação	http://seer.ufrgs.br/renote/
Simpósio Brasileiro de Informática na Educação	http://www.br-ie.org/pub/index.php/sbie/index
Workshop de Desafios da Computação Aplicada à Educação	http://www.br-ie.org/pub/index.php/desafie
Workshop de Informática na	http://www.br-ie.org/pub/index.php/wie

Escola

Workshop sobre Educação
em Computação

<http://portaldeconteudo.sbc.org.br/index.php/wei/index>

No contexto do exemplo deste capítulo, a Tabela 1.4 apresenta um conjunto de critérios que poderiam ser adotados para selecionar os artigos retornados após a aplicação da *string* de busca. Critérios de inclusão são definidos pela sigla CIXX, onde XX indica o número do critério. Analogamente, critérios de exclusão são definidos pela sigla CEXX. Segundo Kitchenham *et al.* (2016), para selecionar as publicações que serão extraídas para responder a questão de pesquisa do estudo secundário, os pesquisadores/profissionais devem fazer a aplicação dos critérios de inclusão e exclusão em duas etapas, analisando dados distintos das publicações. Na primeira etapa é realizado o primeiro filtro, em que o título e resumo dos artigos são lidos visando identificar se o artigo atende algum dos critérios de inclusão. É sugerido que o processo de filtragem seja realizado por dois ou mais pesquisadores/profissionais de forma individual e que eventuais dúvidas sejam discutidas em conjunto para evitar o viés de classificação [Kitchenham *et al.* 2016]. Caso após uma discussão sobre rejeitar ou aceitar uma publicação para a próxima fase, é melhor que o artigo seja aceito para verificar se o mesmo pode ou não servir para responder à questão de pesquisa principal do estudo secundário.

Por sua vez, a segunda fase do processo de seleção de artigos é o segundo filtro, que consiste na leitura completa da publicação, visando avaliar se a mesma realmente atende os critérios de inclusão definidos. Após o segundo filtro, a equipe de pesquisadores/profissionais responsáveis pela condução do estudo secundário terá uma lista de publicações das quais serão extraídos os dados para responder a questão de pesquisa principal e as questões de pesquisa secundárias.

Tabela 1.4 – Lista de possíveis critérios de inclusão e exclusão para o exemplo de um estudo secundário sobre técnicas de avaliação de UX de jogos educacionais

ID	Descrição
CI1	A publicação descreve uma técnica que é utilizada para avaliar jogos educacionais em termos de UX.
CI2	A publicação descreve um processo ou conjunto de passos para avaliar jogos educacionais em termos de UX.
CE1	A publicação não está disponível para leitura e coleta dos dados (artigos pagos ou não disponibilizados pela máquina de busca).
CE2	A publicação está duplicada, isto é, foi selecionada em outra biblioteca digital.
CE3	A publicação não descreve uma técnica ou processo para avaliar a UX de jogos educacionais.
CE4	A publicação não é um artigo científico, e sim um resumo de minicurso, introdução a uma conferência etc.

Durante a **avaliação da qualidade da publicação considerada**, é mensurada a importância dos resultados obtidos nos estudos primários. Esta atividade pode ser necessária em casos onde um ou mais estudos apresentem resultados contraditórios com os resultados apresentados por outros estudos. Para revisões sistemáticas da literatura, esta é uma atividade obrigatória, uma vez que se precisa ter uma análise sobre as evidências experimentais dos estudos primários considerados. Para mapeamentos sistemáticos da literatura, esta é uma atividade opcional. Nesse caso, *checklists* podem ser aplicados para identificar alguns aspectos nos estudos primários selecionados. Exemplos de *checklists* para avaliar a qualidade das publicações podem ser encontrados nos trabalhos de Imtiaz *et al.* (2013) e Usman *et al.* (2014). No exemplo deste capítulo, optou-se por não desenvolver uma avaliação para as publicações, pois está sendo realizado um mapeamento sistemático para identificar as técnicas existentes de avaliação de UX para jogos educacionais. Portanto, a qualidade das publicações em termos dos resultados obtidos nos estudos primários não foi considerada, de modo que qualquer publicação relatando uma técnica de avaliação de UX para jogos educacionais foi considerada.

Com relação à **estratégia para extração de dados**, pode-se preparar um formulário de extração contendo um guia na resposta às questões secundárias definidas durante o planejamento do estudo secundário. Os dados coletados com o formulário de extração podem ser utilizados para realizar meta-análises, ou seja, a agregação dos resultados de vários estudos, aumentando o tamanho da amostra e confiabilidade para estudos que possuem contextos, hipóteses, métricas e fatores similares. Em outros casos, fatores, tecnologias ou conclusões podem ser extraídas ao agregar dados de vários estudos primários. É importante realizar adequadas sínteses dos resultados para que sejam de fácil compreensão e utilização por outros pesquisadores e profissionais interessados nos achados [Cruzes e Dybã 2011]. A Tabela 1.5 apresenta um exemplo de formulário de extração que permite responder as questões secundárias (Subquestão - SQ) de pesquisa propostas no cenário abordado neste capítulo de livro. Nesse contexto, vale ressaltar que o pesquisador/profissional deve ser objetivo no momento de preencher o formulário de extração, de modo a facilitar a replicação/atualização do estudo secundário e a análise dos resultados com base nos dados disponibilizados nos estudos primários. Adicionalmente, no momento de preparar o questionário, o mesmo deve conter perguntas relacionadas e possíveis respostas (ou uma descrição do tipo de resposta esperada) para facilitar seu preenchimento, caso mais de um pesquisador/profissional esteja participando do processo de extração de dados.

Após o processo de extração de dados, preenchendo o formulário de extração para cada um dos artigos selecionados, os pesquisadores/profissionais realizam as etapas de análise e publicação de dos resultados obtidos. Estas etapas são descritas com detalhes na próxima subseção.

Tabela 1.5 – Exemplo de formulário de extração

Formulário de Extração		
ID da Publicação	Para cada publicação selecionada e extraída criar um identificador SXXX, onde XXX é o número da publicação de forma crescente.	
Referência Completa da Publicação	Escrever os dados da publicação, incluindo sobrenome dos autores, ano de publicação, título da publicação, nome do veículo de publicação, edição, número de páginas etc. Seguir o formato de referências ACM.	
Nome da Técnica	Escrever o nome da técnica como descrito na publicação.	
ID	Pergunta	Possível Resposta
SQ1	Qual é a base teórica da técnica proposta?	Citar e descrever todas as teorias que foram usadas pelos autores para propor a técnica em questão.
SQ2	Que aspectos a técnica permite avaliar?	Citar e descrever todos os aspectos de UX que a técnica permite avaliar.
SQ3	Qual é o processo de aplicação da técnica?	Citar e descrever os passos para aplicar a técnica.
SQ4	Em que etapa do processo de desenvolvimento a técnica pode ser aplicada?	Escolher uma ou mais opções: <input type="checkbox"/> Análise e Projeto – Durante a definição do jogo. <input type="checkbox"/> Desenvolvimento – Durante a codificação do jogo. <input type="checkbox"/> Teste – Durante a validação do jogo. <input type="checkbox"/> Implantação – Cenário real de uso.
ID	Pergunta	Possível Resposta
SQ5	Que tipos de jogos educacionais a técnica permite avaliar?	Indicar a plataforma do jogo: <input type="checkbox"/> Web <input type="checkbox"/> Mobile <input type="checkbox"/> Desktop <input type="checkbox"/> Outros - Descrever Indicar o gênero do jogo: <input type="checkbox"/> Aventura <input type="checkbox"/> RPG <input type="checkbox"/> Quis

	<input type="checkbox"/> Outros – Descrever
	<input type="checkbox"/> Possui apoio ferramental
	<input type="checkbox"/> Não possui apoio ferramental
SQ6 A técnica possui apoio ferramental para automatizar parte do seu processo? Se sim, qual e como funciona?	Se sim: Citar e descrever cada uma das ferramentas utilizadas no processo de avaliação.
SQ7 Quais são as vantagens de uso da técnica proposta?	Citar e descrever cada uma das vantagens citadas pelos autores ao avaliar a técnica durante o uso.
SQ8 Quais são as desvantagens do uso da técnica proposta?	Citar e descrever cada uma das desvantagens citadas pelos autores ao avaliar a técnica durante o uso.

3.2. Execução e Publicação dos Resultados

O processo de execução de um estudo secundário consiste na aplicação prática do protocolo de revisão. Nesse contexto, as publicações são identificadas, filtradas e os dados extraídos são utilizados para responder as questões de pesquisa estabelecidas.

A identificação de publicações acontece criando as *strings* de busca com termos relevantes e aplicando-as em bibliotecas digitais; ou através da busca manual em anais de periódicos e conferências ou mesmo no currículo de pesquisadores de renome. O principal objetivo desta atividade é recuperar a maior quantidade de publicações relevantes para responder à questão de pesquisa do estudo secundário, diminuindo o total de falsos positivos (estudos primários retornados durante a identificação que não estão relacionados com a pesquisa). Logo, são aplicados os critérios de inclusão e exclusão para filtrar as publicações retornadas. O pesquisador/profissional pode ler partes da publicação (título, resumo, conclusões etc.) ou ler a publicação completa para decidir se a mesma apresenta resultados relevantes para responder à questão de pesquisa principal. Esta atividade pode contar com a participação de mais de um pesquisador. Ter mais de um pesquisador pode auxiliar na confiabilidade dos resultados obtidos. Além disso, ao envolver mais de um pesquisador, é interessante que se calcule o nível de concordância com inclusão ou exclusão de uma publicação e aumentar a confiabilidade nos resultados do estudo secundário.

Ao analisar os resultados obtidos, o pesquisador/profissional pode examinar os resultados de maneira quantitativa, isto é, apresentando alguns números e/ou percentuais sobre determinados aspectos do tópico estudado. Essas análises devem complementar as respostas da questão de pesquisa formulada no estudo secundário. Adicionalmente, os autores podem utilizar diferentes estratégias para representação dos resultados. O pesquisador/profissional pode aplicar procedimentos de codificação derivados da análise qualitativa de dados para representar os resultados obtidos no trabalho. Esses resultados podem ser agrupados em diferentes categorias ou padrões relevantes para o tópico de

pesquisa estudado. Cruzes e Dyba (2011) apresentam um conjunto de passos para realizar uma síntese temática de resultados na Engenharia de Software. Outra forma de apresentar os resultados obtidos é através de mapas conceituais onde os resultados mais relevantes do estudo secundário podem ser representados de maneira mais visual.

Os resultados de um estudo secundário podem ser publicados em diferentes veículos dependendo do seu objetivo. Por exemplo, se o objetivo do estudo for influenciar profissionais da indústria, os resultados podem ser publicados em periódicos ou revistas orientados para a indústria, pôsteres, panfletos, páginas Web, entre outros. Por outro lado, para a academia, a descrição dos procedimentos de execução do estudo secundário é vital para avaliar a sua qualidade. Ao publicar os resultados de um estudo secundário na academia, o pesquisador deve atualizar o protocolo considerando se houve alterações no mesmo, identificar os critérios de inclusão e exclusão, informações sobre o processo de classificação, listagem de publicações consideradas, entre outros. Uma estrutura dos pontos que podem ser incluídos ao publicar os resultados de um estudo secundário na academia pode ser encontrada no trabalho de Kuhrmann *et al.* (2017).

Todas as etapas de planejamento e execução de estudos secundários são apoiadas por ferramentas disponíveis na literatura. Essas ferramentas visam apoiar os pesquisadores a melhorar a qualidade, replicabilidade e verificação dos resultados obtidos na execução dos estudos secundários. Na seção a seguir, são descritas algumas ferramentas, assim como apresentadas algumas das suas principais funcionalidades.

4. Ferramentas de Apoio ao Processo de Execução de Estudos Secundários

Ferramentas podem prover um apoio relevante para as diferentes etapas de um estudo secundário. Cada uma dessas etapas necessita de esforço intensivo dos pesquisadores/profissionais e está sujeita a diversos erros de execução, fazendo com seja necessário automatizar o máximo de atividades possíveis [Marshall *et al.* 2014]. Diversas ferramentas já foram propostas para auxiliar as etapas de um estudo secundário e estão disponíveis¹ para pesquisadores/profissionais. Algumas ferramentas buscam apoiar atividades específicas, como desenvolvimento do protocolo, seleção de estudos e entre outros. Adicionalmente, há ferramentas que buscam gerenciar todo o processo de execução de um estudo secundário. Nesta seção, serão apresentadas duas ferramentas de apoio: Parsifal e StArt.

4.1. Parsifal

A ferramenta online Parsifal² foi desenvolvida para apoiar o processo de condução de estudos secundários. O objetivo da ferramenta é fornecer apoio a equipes distribuídas no processo de condução por meio da disponibilidade de um espaço de trabalho compartilhado para o planejamento e condução da pesquisa.

A ferramenta permite documentar o processo completo de execução e serve de apoio para lembrar aspectos que devem ser informados no decorrer da execução de um

¹ <http://systematicreviewtools.com/>

² <https://parsifal.org/>

estudo secundário como: objetivos, questões de pesquisa, *string* de busca, seleção de bases etc. A seguir, são apresentadas algumas telas da ferramenta Parsifal, exemplificando como a mesma pode ser aplicada no processo de condução de estudos secundários descrito na Seção 3. Nesse contexto, as tarefas realizadas na ferramenta foram instanciadas no exemplo do mapeamento sistemático sobre técnicas de avaliação de UX para jogos educacionais descrito neste capítulo de livro.

A Figura 1.1 apresenta a primeira tela, relacionada com a atividade de descrever o estudo secundário. Nesta tela, o usuário pode colocar o título da revisão e uma descrição sucinta dos pontos a serem tratados na mesma e sua justificativa.

The image shows a web browser window with the Parsifal logo in the top left corner. The main heading is "Create a new review". Below this, there are two input fields. The first is labeled "Title" and contains the text "Mapeamento Sistemático Sobre Técnicas de Avaliação de UX para Jogos Educacionais". The second is labeled "Description" and contains the placeholder text "Give a brief description of your systematic literature review". There is a hamburger menu icon in the top right corner.

Figura 1.1. Tela inicial da ferramenta Parsifal para o processo de condução de estudos secundários

A Figura 1.2, por sua vez, apresenta o início de preenchimento do protocolo de revisão. A ferramenta permite documentar os objetivos, PICO, questões de pesquisa, palavras chave e sinónimos, *string* de busca, bases que serão consideradas e os critérios de seleção dos artigos. Nesta figura, foi preenchido o objetivo de identificar as técnicas para avaliação de UX de jogos educacionais, assim como os termos do PICO definido na Seção 3.2.

Com relação a *string* de busca, às bases e aos critérios de inclusão e exclusão, a Figura 1.3 apresenta o formulário da ferramenta Parsifal para estes itens. Todos estes itens devem ser preenchidos, pois serão usados na etapa de condução do estudo.

Figura 1.2. Tela da ferramenta Parsifal para o processo de planejamento do estudo secundário, solicitando objetivos e especificação do PICO

Figura 1.3. Tela da ferramenta Parsifal para o processo de planejamento do estudo secundário, solicitando *string* de busca, bases contendo os artigos a serem consultadas e os critérios de inclusão e exclusão

Durante o planejamento do estudo secundário, os pesquisadores/profissionais também devem indicar os dados que serão extraídos dos artigos aceitos a partir da aplicação dos critérios de inclusão/exclusão. A Figura 1.4 apresenta o processo de criação do formulário. Nesse contexto, a ferramenta Parsifal permite criar diferentes tipos de questões: discursivas, múltipla escolha, booleana etc.

Description	Type	Values
Nome da Técnica	String Field	n/a
SQ1 Qual é a base teórica da técnica proposta?	String Field	n/a
SQ2 Que aspectos a técnica permite avaliar?	String Field	n/a
SQ3 Qual é o processo de aplicação da técnica?	String Field	n/a
SQ4 Em que etapa do processo de desenvolvimento a técnica pode ser aplicada?	Select Many Field	<ul style="list-style-type: none">Análise e Projeto – Durante a definição do jogo.Desenvolvimento – Durante a codificação do jogo.Implantação – Ao aplicar o jogo em um cenário real de uso.Teste – Durante a validação do jogo.

+ Add Field

Figura 1.4. Tela da ferramenta Parsifal para a definição do formulário de extração

Para executar o estudo secundário, os pesquisadores/profissionais devem importar as publicações das respectivas bases. A Figura 1.5 e a Figura 1.6 apresentam duas formas em que isto é feito na ferramenta Parsifal. Na primeira, a ferramenta Parsifal possui integração com bibliotecas digitais como Scopus e Science Direct, portanto basta aplicar a *string* de busca seguindo as normas de cada biblioteca. Por sua vez, no caso de bases não indexadas ou não integradas com a ferramenta, é possível importar arquivos com extensão BibTeX, que contem meta dados dos artigos. Estes arquivos podem ser baixados nas bibliotecas digitais, ou elaborados manualmente pelos pesquisadores/profissionais envolvidos no estudo secundário.

Durante o processo de condução da revisão, os pesquisadores devem classificar os artigos como: aceitos, rejeitados ou duplicados, considerando os critérios de inclusão e exclusão. Nesse contexto, a Figura 1.7 mostra a listagem de artigos importados na ferramenta Parsifal que podem ser filtrados segundo a base da qual foram importados; e a Figura 1.8 apresenta a tela com os dados da publicação. Nesta tela, os pesquisadores/profissionais devem classificar os artigos. Ao salvar a classificação do artigo, se uma publicação for classificada como aceita, o sistema apresenta o formulário de extração para a mesma e a equipe pode informar os dados relevantes para responder as questões de pesquisa do estudo secundário. A Figura 1.9 apresenta o exemplo de um formulário de extração para um dos artigos selecionados considerando as questões definidas na Seção 3.1.4.

VII Congresso Brasileiro de Informática na Educação (CBIE 2018)
VII Jornada de Atualização em Informática na Educação (JAIE 2018)

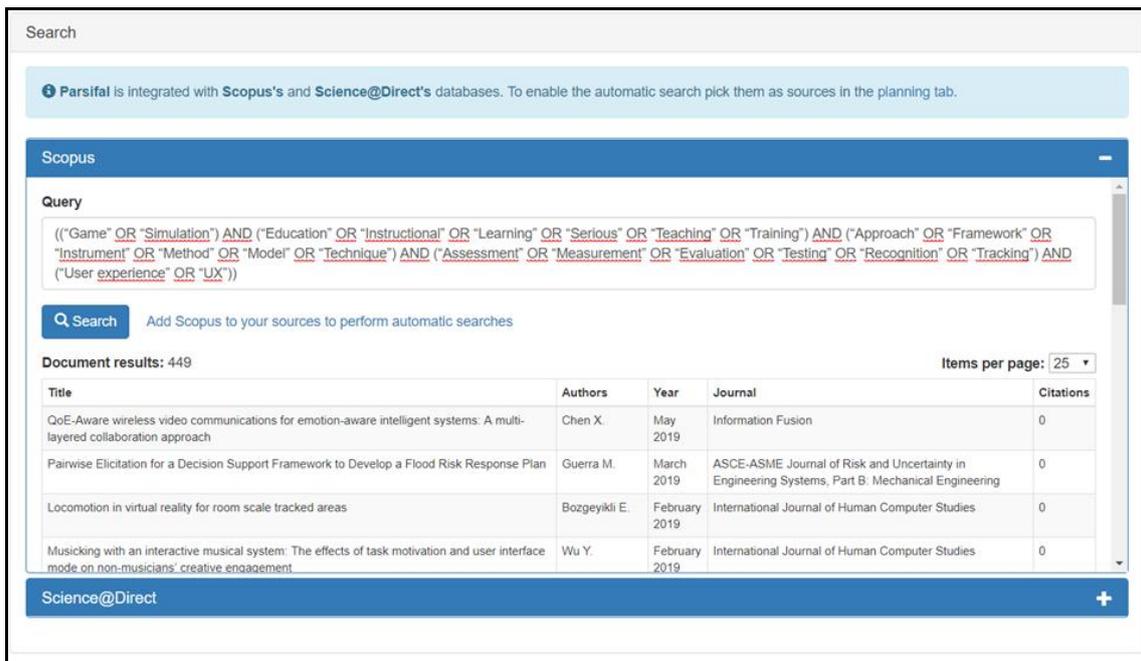


Figura 1.5. Tela da ferramenta Parsifal permitindo importar dados sobre publicações diretamente de bibliotecas digitais como Scopus

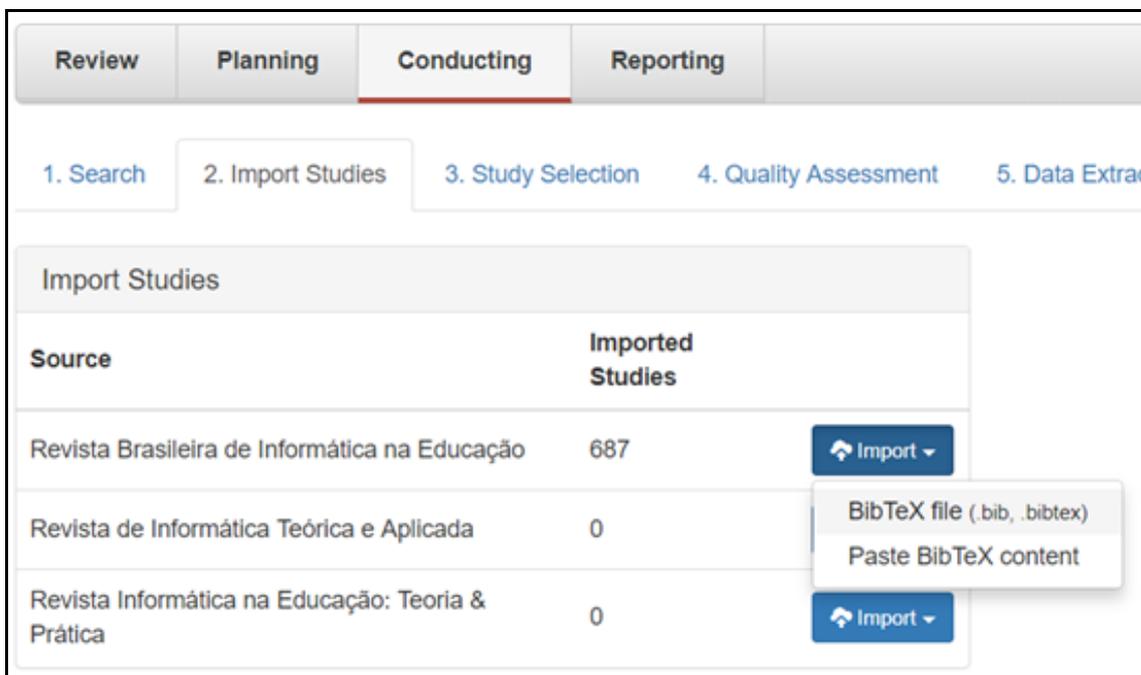


Figura 1.6. Tela da ferramenta Parsifal para importar os estudos secundários

VII Congresso Brasileiro de Informática na Educação (CBIE 2018)
VII Jornada de Atualização em Informática na Educação (JAIE 2018)

Study Selection

All Sources [Revista Brasileira de Informática na Educação](#) [Revista de Informática Teórica e Aplicada](#) [Revista Informática na Educação: Teoria & Prática](#)

[Find Duplicates](#) [Export Articles](#)

Action: 0 of 687 selected

Show: All Accepted Rejected Unclassified Duplicated

<input type="checkbox"/>	Bibtex Key	Title	Author	Journal	Year	Added by	Added at	Status
<input type="checkbox"/>	Madhavan20141823	DIA2: Web-based cyberinfrastructure for visual analysis of funding portfolios	Madhavan, K.a and Elmqvist, N.a and Vorvoreanu, M.a and Chen, X.a and Wong, Y.a and Xian, H.b and Dong, Z.b and Johri, A.c	IEEE Transactions on Visualization and Computer Graphics	2014	luchoufma	15 Sep 2018 18:53:45	Unclassified
<input type="checkbox"/>	NunesMarciano2014	Evaluating multiple aspects of educational computer games: Literature review and case study	Nunes Marciano, J. and Cunha De Miranda, L. and Esteves Cunha De Miranda, E.	International Journal of Computer Games Technology	2014	luchoufma	15 Sep 2018 18:53:45	Unclassified
<input type="checkbox"/>	GonzálezSánchez2014259	Assessing the player interaction experiences based on playability	González Sánchez, J.L.a and Gutiérrez Vela, F.L.b	Entertainment Computing	2014	luchoufma	15 Sep 2018 18:53:45	Unclassified

Figura 1.7. Tela da ferramenta Parsifal listando os artigos importados das bibliotecas digitais

Article Details (1/687)

Details [Comments](#) [External Link](#)

Status: Selection Criteria:

Title:

Abstract:

Year:

Author:

Automatically save the status on change and move next

Figura 1.8. Tela da ferramenta Parsifal para o processo de filtragem de artigos considerando os critérios de inclusão e exclusão

Data Extraction

To-do Done All

Export Data

Model for analysis of serious games for literacy in deaf children from a user experience approach

Nome da Técnica: Model for assessing the quality of serious games and user experience

SQ1 Qual é a base teórica da técnica proposta? Modelo Conceitual de UX
O modelo considera que a UX é decomposta por três aspectos:
- Usuário
- Estratégia de Aprendizado

SQ2 Que aspectos a técnica permite avaliar? Atributos hedônicos - relacionados com as emoções do usuário no contato com o jogo.
Atributos pragmáticos - relacionados com os objetivos a serem realizados no jogo

SQ3 Qual é o processo de aplicação da técnica?

SQ4 Em que etapa do processo de desenvolvimento a Análise e Projeto – Durante a definição do jogo. Desenvolvimento – Durante a codificação do jogo.
 Implantação – Ao aplicar o jogo em um cenário real de uso. Teste – Durante a validação do jogo.

Figura 1.9. Tela da ferramenta Parsifal para o processo de extração de dados

Finalmente, a ferramenta Parsifal fornece apoio ao processo de publicação dos resultados obtidos no estudo secundário. Por um lado, a ferramenta gera gráficos automáticos relacionados ao quantitativo de artigos aceitos, publicações identificadas por ano, entre outros (ver exemplo na Figura 1.10). Além disso, é possível exportar os dados dos formulários de extração para fazer análises mais aprofundadas e responder as questões de pesquisa do estudo secundário.

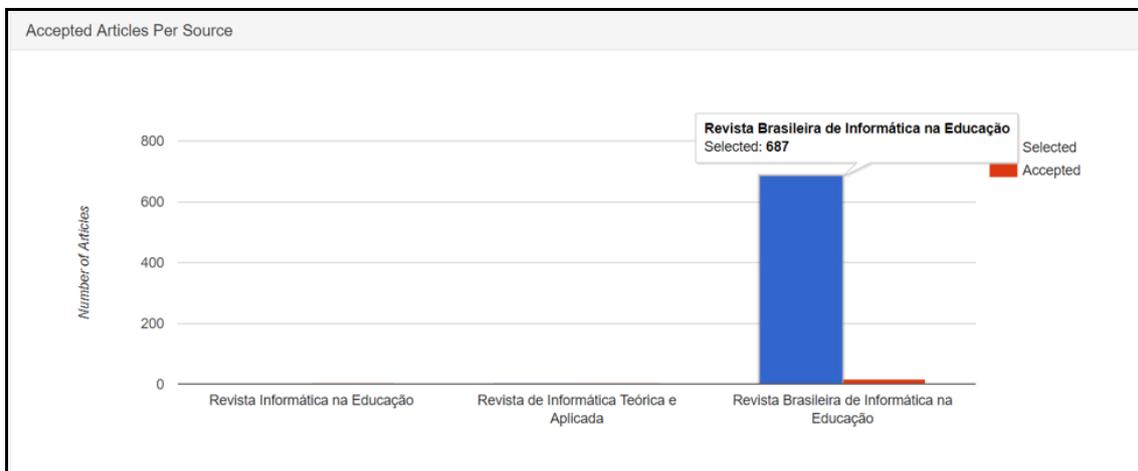


Figura 1.10. Exemplo de suporte da ferramenta Parsifal para a geração de gráficos

4.2. Start

A StArt3 (do inglês, *State of the Art through Systematic Review*) é uma ferramenta de apoio à execução de estudos secundários desenvolvida por pesquisadores da Universidade Federal de São Carlos (UFSCAR). Diversos estudos experimentais já

³ http://lapes.dc.ufscar.br/tools/start_tool

foram apresentados mostrando a aplicabilidade desta ferramenta em estudos secundários [Marshall *et al.* 2014] [Fabbri *et al.* 2016]. A ferramenta apoia a execução de todas as etapas do estudo secundário.

Na etapa de planejamento, a ferramenta permite a inclusão de todas as informações de um protocolo de revisão, incluindo itens do PICO. Em seguida, para cada fonte de publicações definida pelo pesquisador/profissional, pode-se criar “search sessions”. Essas “search sessions” permitem que diferentes buscas em uma base de publicações seja registrada. Desta forma, é possível manter um histórico das buscas feitas durante a execução do estudo secundário. A ferramenta permite a utilização de arquivos bibtex, facilitando a inclusão dos trabalhos retornados nas bases digitais de publicações.

A atividade de seleção de artigos é realizada de maneira que seja possível registrar o critério de inclusão ou o critério de exclusão utilizado. Além disso, a ferramenta permite que observações sejam feitas durante toda a atividade de seleção. Ao final do processo de seleção de artigos, a ferramenta apresenta, para cada publicação selecionada, o formulário de extração de dados. Nesta atividade, o pesquisador/profissional busca preencher todos os campos disponíveis no formulário de extração. Por exemplo, a Figura 1.11 apresenta o processo de seleção de publicações considerando os critérios de inclusão e exclusão definidos durante o planejamento do estudo secundário.

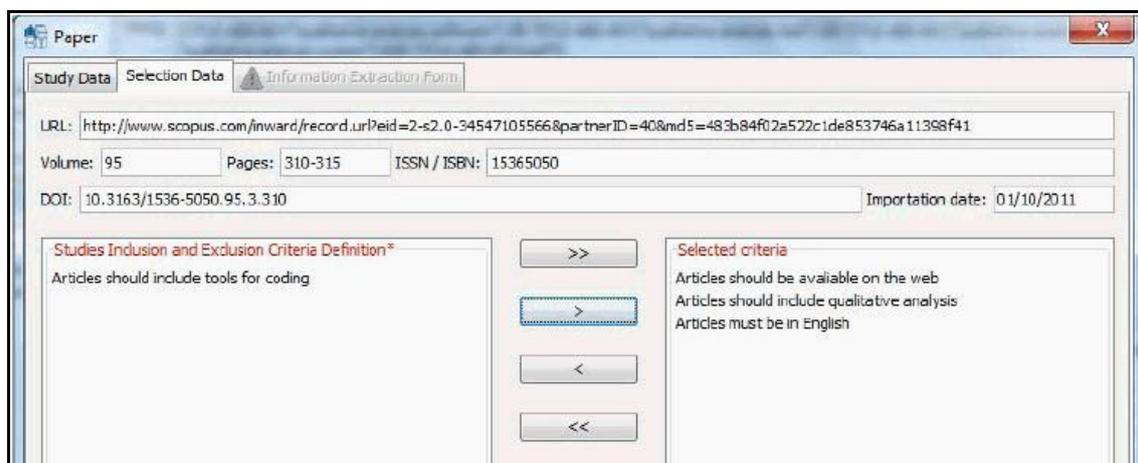


Figura 1.11. Exemplo de suporte da ferramenta StArt para o processo de seleção de publicações [Hernandes *et al.* 2012]

Além do apoio às atividades do processo de estudo secundário, a ferramenta disponibiliza alguns gráficos com informações sobre quantidade de artigos selecionados em todo o estudo secundário, apresentando o quantitativo para cada critério de inclusão. Por fim, a ferramenta possibilita exportar o protocolo e resultados do estudo secundário.

5. Exemplo de Revisão Sistemática em Informática na Educação

A título de ilustração, um dos campos que têm utilizados estudos secundários em Informática na Educação é o ensino de algoritmos e programação, sobretudo motivado pelo fato de que este conteúdo sustenta a formação acadêmica básica em Computação e

áreas relacionadas. Vários métodos, técnicas e ferramentas têm sido desenvolvidos para apoiar o ensino-aprendizagem, com destaque para o uso de jogos digitais. Por exemplo, um trabalho recentemente publicado no Simpósio Brasileiro de Informática na Educação (SBIE 2018) teve como objetivo replicar uma revisão sistemática da literatura referente a 2008-2012 a fim de investigar a utilização de jogos digitais no ensino de programação para iniciantes em Computação no nível superior no Brasil, cobrindo a última década (2008-2017). Este trabalho, de Silva *et al.* (2018), compara ainda os resultados obtidos nos dois quinquênios da última década e identifica ferramentas e teorias pedagógicas utilizadas no ensino-aprendizagem de programação.

Em seu planejamento, o trabalho utiliza os critérios PICO para definir a questão de pesquisa (*Qual o panorama do uso de jogos digitais no ensino de programação para iniciantes em Computação no nível superior na última década?*), bem como sub-questões para apoiar a sua investigação: (1) Sub-Q1: Como os jogos digitais têm sido aplicados no ensino de programação? (2) Sub-Q2: Que ferramentas têm sido empregadas? (3) Sub-Q3: Quais os efeitos observados nos alunos com a introdução de atividades de programação baseadas em jogos digitais? (4) Sub-Q4: Que teorias pedagógicas estão sendo utilizadas? (5) Sub-Q5: Quais são as instituições de pesquisas envolvidas na área e como elas estão distribuídas no Brasil? (6) Sub-Q6: Quais os avanços obtidos no uso de jogos no ensino de programação para iniciantes em Computação no nível superior na última década? Além disso, as questões foram pautadas pelo uso do método GQM (*Goal-Question-Metric*) [Basili 1992] para apoiar a definição dos objetivos do estudo.

Uma vez que o estudo se voltou para o cenário nacional e replica a revisão sistemática anterior de Medeiros *et al.* (2013), foram executadas buscas nas seguintes bases de artigos: (1) Simpósio Brasileiro de Informática na Educação (SBIE); (2) Workshop de Desafios da Computação Aplicada à Educação (DesafIE); (3) Workshop de Informática na Escola (WIE); (4) Workshop sobre Educação em Computação (WEI); (5) Revista Brasileira de Informática na Educação (RBIE); (6) Revista de Informática Teórica e Aplicada (RITA); e (7) Revista Novas Tecnologias na Educação (RENOTE). Dessa forma, não houve uma *string* de busca propriamente dita. Adicionalmente, foram estabelecidos critérios de inclusão e exclusão, bem como definidos e avaliados critérios de qualidade, conforme recomendado para revisões sistemáticas.

Após a realização da busca manual, obteve-se um total de 34 artigos, sendo 12 selecionados. Por ser uma replicação, oito estudos analisados no trabalho de Medeiros *et al.* (2013) foram incluídos, totalizando 20 artigos para análise. Para responder a cada subquestão de pesquisa, optou-se por explorar o cenário mais atual (2013-2017) – não tratado no estudo anterior – e fazer ainda uma comparação dos resultados obtidos nos dois períodos (2008-2012 *versus* 2013-2017). Ao final, na seção de discussão, foram apontadas ainda tendências observadas para a última década no que se refere à utilização de jogos digitais no ensino de programação para iniciantes em Computação no nível superior no Brasil. Por exemplo, após responder a Sub-Q6, conclui-se que, nos últimos anos, oficinas para ensino de programação com jogos digitais foram exploradas de maneira crescente, partindo de uma (2008-2012) para cinco (2013-2017) iniciativas.

Além disso, novas formas de avaliar os trabalhos dos alunos apareceram com o passar do tempo, tais como avaliação por especialista e estudos experimentais.

Por fim, o trabalho apresenta algumas limitações da replicação da revisão sistemática. Entre elas, está a quantidade de artigos no tema (total = 20), considerando o escopo do trabalho (utilização de jogos digitais para ensino de programação aos iniciantes em Computação no nível superior no Brasil na última década). Reforça-se, contudo, que ameaças à validade e limitações são elementos importantes para serem discutidos em estudos secundários a fim de assegurar reaplicações futuras ou para apoiar pesquisadores que os utilizem para estudos terciários.

6. Considerações Finais

A execução de estudos secundários traz inúmeros benefícios para as pesquisas científicas e benefícios na proposição de novas tecnologias. Através dos resultados desses estudos é possível identificar lacunas que oportunizarão novas pesquisas e tecnologias. Além disso, a execução adequada de um estudo secundário também possibilita a publicação de um artigo em veículos científicos. Este capítulo buscou apresentar os conceitos de estudos secundários levando em consideração aspectos práticos de sua condução. Deste modo, espera-se que os conhecimentos fornecidos contribuam na melhoria nos trabalhos publicados pela comunidade da área de Informática na Educação.

Há diversos desafios na execução de estudos secundários. Esses desafios precisam ser tratados de modo que tragam benefícios para todas as áreas da computação, incluindo Informática na Educação. Alguns desafios são:

- (1) definição adequada das bibliotecas digitais – quais são as bibliotecas digitais que trazem publicações relevantes para determinadas áreas da computação?
- (2) definição da *string* de busca de modo que ela cubra o maior número de publicações de interesse – quais palavras usar? Quais sinônimos devem ser usados para conseguir coletar o maior número de publicações possíveis?
- (3) seleção e extração adequada dos estudos primários – como selecionar os estudos primários? Quais informações são relevantes para extrair? Quantos pesquisadores podem apoiar na realização dessa atividade? Como garantir que todos os pesquisadores envolvidos no processo de execução do estudo irão selecionar e extrair os dados de forma semelhante?
- (4) síntese dos resultados do estudo secundário – como sintetizar os dados obtidos? Utilizar esquemas gráficos sempre é uma ótima opção?
- (5) definição adequada do tipo de revisão – qual técnica de estudo secundário deve ser utilizada e quais as suas vantagens e desvantagens?

Visto esses desafios, este capítulo apresentou uma reflexão e exemplos práticos para maior compreensão sobre a condução dos estudos secundários, objetivando sempre melhorar a qualidade dos estudos secundários publicados pela comunidade de Informática na Educação.

Referências

- Bargas-Avila, J. and Hornbæk, K. (2011). "Old wine in new bottles or novel challenges: a critical analysis of empirical studies of user experience". In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2689-2698.
- Basili, V. (1992). "Software Modeling and Measurement: The Goal/Question/Metric Paradigm". Technical Report CS-TR-2956, University of Maryland.
- Battistela, P. E., Wangenheim, C. G. V., & Fernandes, J. M. (2014). "Como jogos educacionais são desenvolvidos? Uma revisão sistemática da literature". In XXII Workshop sobre Educação em Computação (WEI 2014). Sociedade Brasileira de Computação.
- Bellotti, F., Kapralos, B., Lee, K., Moreno-Ger, P., & Berta, R. (2013). "Assessment in and of serious games: an overview". *Advances in Human-Computer Interaction*, 2013, 1.
- Cascini, D., & Campos, S. (2015, October). "Avaliação de jogos educacionais multiusuários: Uma revisão sistemática da literature". In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)* (Vol. 26, No. 1, p. 519).
- Connolly, T. M., Boyle, E. A., MacArthur, E., Hainey, T., & Boyle, J. M. (2012). "A systematic literature review of empirical evidence on computer games and serious games". *Computers & Education*, 59(2), 661-686.
- Cruzes, D. S., & Dybå, T. (2011). "Research synthesis in software engineering: A tertiary study". *Information and Software Technology*, 53(5), 440-455.
- Detroz, J. P., Hinz, M., & da Silva Hounsell, M. (2015). "Uso de Pesquisa Bibliográfica em Informática na Educação: um Mapeamento Sistemático". *Revista Brasileira de Informática na Educação*, 23(1).
- Fabbri, S., Octaviano, F., Silva, C., Di Thommazo, A., Hernandez, E., and Belgamo, A. (2016). "Improvements in the Start tool to better support the systematic review process". In *Proc. of the 20th International Conference on Evaluation and Assessment in Software Engineering (EASE'16)*, Limerick, Ireland, June 2016.
- Fowler, M., & Highsmith, J. (2001). "The agile manifesto". *Software Development*, 9(8), 28-35.
- Hernandes, E., Zamboni, A., Fabbri, S., & Thommazo, A. D. (2012). "Using GQM and TAM to evaluate StArt-a tool that supports Systematic Review". *CLEI Electronic Journal*, 15(1), 3-3.
- Imtiaz, S., Bano, M., Ikram, N., & Niazi, M. (2013, April). "A tertiary study: experiences of conducting systematic literature reviews in software engineering". In *Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering* (pp. 177-182). ACM.
- Iosup, A., & Epema, D. (2014, March). "An experience report on using gamification in technical higher education". In *Proceedings of the 45th ACM technical symposium on Computer science education* (pp. 27-32). ACM.

- Kitchenham, B. A., and Charters, S. (2007). "Procedures for performing systematic literature reviews in software engineering". Keele University & Durham University, UK.
- Kitchenham, B. A., Budgen, D., & Brereton, P. (2016). "Evidence-based software engineering and systematic reviews". CRC Press: New York. 426p.
- Kuhrmann, M., Fernández, D. M., & Daneva, M. (2017). "On the pragmatic design of literature studies in software engineering: an experience-based guideline". *Empirical Software Engineering*, 22(6), 2852-2891.
- Law, E. L. C., & Sun, X. (2012). "Evaluating user experience of adaptive digital educational games with Activity Theory". *International Journal of Human-Computer Studies*, 70(7), 478-497.
- Mafra, S., Barcelos, R., Travassos, G.: "Aplicando uma Metodologia Baseada em Evidência na Definição de Novas Tecnologias de Software". In: Proceedings of the 20th Brazilian Symposium on Software Engineering (SBES 2006). SBC.
- Magalhães, C. V., Santos, R. E., da Silva, F. Q., & Gomes, A. S. (2013). "Caracterizando a pesquisa em informática na educação no Brasil: um mapeamento sistemático das publicações do SBIE". In: Anais do Simpósio Brasileiro de Informática na Educação (SBIE 2013). SBC.
- Marshall, C., Brereton, P., & Kitchenham, B. (2014, May). "Tools to support systematic reviews in software engineering: a feature analysis". In Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering (p. 13). ACM.
- Medeiros, T. J., da Silva, T. R., & da Silva Aranha, E. H. (2013). "Ensino de programação utilizando jogos digitais: uma revisão sistemática da literature". *RENOTE*, 11(3).
- Nacke, L. E., Drachen, A., & Göbel, S. (2010). "Methods for evaluating gameplay experience in a serious gaming context". *International Journal of Computer Science in Sport*, 9(2), 1-12.
- Nakagawa, E. Y., Scannavino, K. R. F., Fabbri, S. C. P. F., & Ferrari, F. C. (2017). "Revisão Sistemática da Literatura em Engenharia de Software: Teoria e Prática". Elsevier Brasil. 144p.
- Rivero, L., & Conte, T. (2017, October). "A systematic mapping study on research contributions on UX evaluation technologies". In Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems (p. 5). ACM.
- Schiehl, E. P., & Gasparini, I. (2017). "Modelos de Ensino Híbrido: Um Mapeamento Sistemático da Literatura". In: Anais do Simpósio Brasileiro de Informática na Educação (SBIE 2017). SBC.
- Shull, F., Carver, J. and Travassos, G. (2001). "An empirical methodology for introducing software processes". *ACM SIGSOFT Software Engineering Notes*, 26(5), 288-296.

- Silva, R. R., Fernandes, J. C., Santos, R. P. (2018). “Panorama da Utilização de Jogos Digitais no Ensino de Programação no Nível Superior na Última Década: Uma Revisão Sistemática da Literatura”. In Anais do SBIE 2018.
- Tell, P., Cholewa, J. B., Nellesmann, P., & Kuhrmann, M. (2016). “Beyond the spreadsheet: Reflections on tool support for literature studies”. In: Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering (EASE 2016). ACM.
- Travassos, G. H., Gurov, D., & Amaral, E. A. G. G. (2002). “Introdução à engenharia de software experimental” (p. 53). UFRJ.
- Usman, M., Mendes, E., Weidt, F., & Britto, R. (2014, September). “Effort estimation in agile software development: a systematic literature review”. In Proceedings of the 10th International Conference on Predictive Models in Software Engineering (pp. 82-91). ACM.
- Vermeeren, A., Law, E., Roto, V., Obrist, M., Hoonhout, J. and Väänänen-Vainio-Mattila, K. (2010). “User experience evaluation methods: current state and development needs”. In Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries, 521-530.
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., & Wesslén, A. (2012). “Experimentation in software engineering”. Springer Science & Business Media.

Sobre os autores



Luis Rivero – É doutor em Informática pela Universidade Federal do Amazonas (título obtido em 2017) e Professor Adjunto A do Departamento de Informática da Universidade Federal do Maranhão, onde atua em cursos de graduação e como membro colaborador do Programa de Pós-Graduação em Ciência da Computação. É membro do grupo de pesquisa Desenvolvimento de Software para Sistemas Embarcados Complexos da UFAM e do grupo Núcleo de Computação Aplicada NCA-UFMA, possuindo experiência na área de Engenharia de Software, Interação Humano Computador e Informática na Educação, atuando principalmente nos seguintes temas: Engenharia de Aplicações Móveis, Engenharia de Software Experimental, Qualidade de Software, Avaliação de Usabilidade, Experiência do Usuário e Desenvolvimento de Jogos Educacionais. Tem experiência na docência de cursos de graduação e pós-graduação pelo Centro Universitário do Norte – UNINORTE e a Universidade Federal do Amazonas. Atua como revisor de periódicos da área de Engenharia de Software: Journal of Universal Computer Science, Information and Software Technology, e International Journal of Software Engineering and Knowledge Engineering. Atua como revisor e/ou membro do comitê de programa de conferências da área de computação como: CIBSE, SEMISH, SBSI, WASHES, IHC, entre outros.

CV Lattes: <http://lattes.cnpq.br/8534667641706692>



Davi Viana – É Doutor em Informática pela UFAM (título obtido em 2015) e Professor Adjunto da Universidade Federal do Maranhão, onde atua em cursos de graduação (Bacharelado Interdisciplinar em Ciência e Tecnologia e Bacharelado em Engenharia da Computação) e no Programa de Pós-Graduação em Ciência da Computação (PPGCC). É membro do Laboratório de Sistemas Distribuídos Inteligentes (LDSi). Além disso, é Diretor da Divisão de Difusão do Empreendedorismo da UFMA e membro do Comitê Gestor da Comissão Especial de Sistemas de Informação (CE-SI) da SBC. Possui experiência na área de Sistemas de Informação, Engenharia de Software e Informática na Educação, atuando principalmente nos seguintes temas: Desenvolvimento de Jogos Educacionais, Engenharia de Software Experimental, Qualidade de Software. Atua como revisor e/ou membro do comitê de programa de conferências da área de computação como: SBQS, SBSI, CIBSE, WASHES, entre outros. Também é editor da sessão de Sistemas de Informação da Revista Eletrônica de Iniciação Científica (REIC).

CV Lattes: <http://lattes.cnpq.br/9297257833779277>



Rodrigo Santos – Professor Adjunto do Departamento de Informática Aplicada (DIA) e membro efetivo do Programa de Pós-Graduação em Informática (PPGI) da Universidade Federal do Estado do Rio de Janeiro (UNIRIO). Doutor em Engenharia de Sistemas e Computação pela COPPE/UFRJ, onde realizou também o seu Pós-doutorado (2016). Atuou como pesquisador visitante na University College London (2014-2015). Atuou como consultor em projetos de pesquisa e desenvolvimento de engenharia de sistemas na indústria nacional pela Fundação Coppetec entre 2008 e 2017. É editor-chefe da iSys: Revista Brasileira de Sistemas de Informação e organizou edições especiais em periódicos como JBCS e JISA, além de volume na série Springer CCIS. É membro da SBC desde 2006 e membro do Comitê Gestor da Comissão Especial de Sistemas de Informação (CE-SI). É avaliador de cursos superiores em Computação e Informática MEC desde 2011. Tem experiência na área de Ciência da Computação, com ênfase em Engenharia de Software, pesquisando sobre Ecossistemas de Software, Engenharia de Requisitos e Educação em Engenharia de Software. É/foi coordenador científico do SBSI 2019, ACM ECSA/WDES 2018, SBSI/Minicursos 2018, IHC/Pôsteres-Demos 2018, SBGames/CTD 2018, CBSOFT/WDTSoft 2018, CSBC/BraSNAM 2017/2018, IEEE/ACM ICSE/SESOS-WDES 2017, ICSR/WASHES 2017, IHC/WAIHCWS 2017/2018, SBQS/WASHES 2016 e CBSOFT/WDES 2014/2015. Ministrou palestras, minicursos e tutoriais no SBSI (2010/2011/2013/2017/2018), CLEI (2016/2018), CBSOFT (2012/2013/2016/2017), SBGames (2017), SBQS (2009/2015/ 2016), Webmedia (2016), IHC (2016), CIBSE (2012), SBIE (2010) e ICTAC (2010).

CV Lattes: <http://lattes.cnpq.br/8613736894676086>

Capítulo

3

Desenvolvimento do Pensamento Computacional através do Uso de Tutoriais Interativos

Gilson Pedroso dos Santos, Ronilson dos Santos Bezerra

Resumo

Este trabalho discute sobre o Pensamento Computacional (PC), sua introdução no processo de ensino-aprendizagem e sua relevância para o indivíduo do século XXI. Apresenta também duas ferramentas que podem ser muito importantes, porque podem auxiliar no processo educativo visando a disseminação do Pensamento Computacional. Uma dessas ferramentas denomina-se Tutorial Interativo (TI), recurso computacional que objetiva facilitar a compreensão do conteúdo pelo receptor. A outra ferramenta é um software de programação em blocos denominado Scratch. Esta já é popularizada em mais de 150 países e possui mais de 28 milhões de usuários. Por ser embasada nos princípios do construcionismo, esta pode ser utilizada nas mais diversas atividades pedagógicas, desenvolvendo no aluno, a capacidade do pensar computacionalmente. Para a elaboração deste trabalho, foi feita uma minuciosa pesquisa bibliográfica sobre o assunto em tela, sob a ótica de vários pesquisadores e percebeu-se que todos convergem para uma necessidade urgente: a necessidade do cidadão do século XXI aprender a utilizar a tecnologia como sua aliada, de forma consciente e pensando computacionalmente.

Palavras-chave: Tutoriais Interativos. Scratch. Pensamento Computacional. Ensino.

Abstract

The Computational Thinking (CP), its introduction to the teaching-learning process, as well as its relevance to the 21st century individual is what is being discussed in this paper. It also presents two learning tools which can be very important as they facilitate the educational process in order to disseminate Computational Thinking skills. One of these tools is called Interactive Tutorial (IT), a computational resource whose aim is help the receiver to understand the learning material sent to him/her. The other tool is a blocks (drag and drop) programming software called Scratch. Based on the constructivism, the latter has already been popularized in more than 150 countries and has more than 28 million users as is it can be used in several educational activities

which lead students to develop computational thinking skills. To accomplish this work, a detailed literature survey was conducted on the reported subject according to the point of view of various researchers and, as a result, it was noticed these researchers are all convergent to an urgent need: the 21st individual's need to learn how to use technology to self-improvement in a conscious way and thinking computationally.

Keywords: Interactive Tutorials. Scratch. Computational Thinking. Teaching.

1. Introdução

O presente trabalho trata sobre a utilização da ferramenta Scratch e de Tutoriais Interativos (TI) no processo de ensino-aprendizagem.

O mesmo busca discutir ainda o desenvolvimento do Pensamento Computacional (PC) aliado aos TI's e como elaborar estas ferramentas através de programas como *Microsoft Office Power Point* e *Microsoft Paint*. A importância deste se reflete na necessidade de se investigar as possibilidades de aplicações do PC na educação. Para auxiliar nessa discussão é apresentado um *template* contendo as principais partes que um tutorial interativo precisa ter.

Como procedimentos metodológicos utilizados, destacam-se as revisões bibliográficas sobre Scratch, Tutoriais Interativos, bem como sobre o Pensamento Computacional e TIC. A partir disso, foi elaborado e documentado o *template* para basilar a produção de Tutoriais Interativos. É apresentada ainda a proposta de desenvolvimento de um TI através de desafios.

A seguir é informado como está estruturado o presente trabalho.

Na seção 2 é apresentado o referencial teórico, em três subseções, onde: Na subseção 2.1 consta uma explanação sobre o PC, sob a ótica de vários pesquisadores, e sua aplicação no processo de ensino-aprendizagem; Em seguida, na subseção 2.2, é exposto o conceito de TI e sua importância no processo da comunicação e na subseção 2.3 são discutidos sobre o Scratch e sua aplicabilidade no processo educativo.

Na seção 3 foi abordado sobre as TIC no processo de ensino-aprendizagem, importância do professor para a apropriação dos recursos computacionais, dentre outros.

Já na seção 4 é evidenciada a construção de um tutorial interativo e foi abordado ainda sobre o processo de atribuição de interatividade à ferramenta.

Apresenta-se na seção 5 uma proposta de desenvolvimento de um tutorial interativo através da execução de desafios.

Por fim, na seção 6, os autores teceram suas considerações finais sobre este trabalho.

2. Referencial Teórico

2.1. Pensamento Computacional

Esta seção aborda sobre o Pensamento Computacional, bem como sobre a sua aplicação no processo de ensino-aprendizagem.

Nas últimas décadas, os dispositivos eletrônicos passaram a fazer parte do cotidiano das pessoas. Se antes os computadores eram restritos às forças militares,

universidades e grandes empresas, atualmente eles estão presentes em todos os lugares. Com isso, a computação, enquanto ciência, foi ganhando espaço na vida das pessoas e por mais que estas não tenham conhecimento sobre esta área, diariamente, elas fazem uso de algum dispositivo computadorizado, seja um caixa eletrônico, celular, *tablet*, computador de bordo, *desktops*, *notebooks* e até mesmo nos eletrodomésticos.

Este acesso diário aos computadores, somado à popularização da internet, mudou completamente a forma como as pessoas obtêm informações. Se antes as informações passavam primeiro pelas redações dos jornais impressos e apenas horas ou dias depois eram publicadas, hoje as pessoas têm acesso às informações quase que de forma instantânea. Ademais, com o advento da *Web 2.0*¹, além de consumidoras de informações, estas pessoas também passaram a atuar na produção e compartilhamento de conteúdo.

Além disso, a geração atual se depara, cotidianamente, com a necessidade de saber lidar com soluções computacionais, uma vez que elas estão presentes nas mais diversas áreas. Desse modo, estar familiarizado com essas soluções pode facilitar ou não a inserção de uma pessoa no mercado globalizado atual.

Diante da ascensão do uso dessas tecnologias, tornou-se imprescindível a discussão sobre a inserção das mesmas no processo de ensino-aprendizagem. Desde então o uso das Tecnologias da Informação e da Comunicação (TIC) ganhou mais destaque e tornou-se objeto de estudo de diversos pesquisadores. E neste sentido é necessário fazer uma distinção entre o ensino da informática - ensino restrito ao “adestramento” para uso de alguns aplicativos, tais como editores de texto, acesso à internet e outros - do ensino da computação enquanto ciência. Na concepção de França et al. (2014 apud Nunes, 2008) esta diferenciação é imprescindível.

Ao abordar sobre a computação no processo educativo, Geraldes (2017 apud CSTA, 2005) destaca que a computação enquanto ciência não se refere apenas às habilidades de apontar e clicar, mas sim de utilizar um conjunto de fundamentos da computação na resolução sistemática de problemas do mundo real. O que segundo o autor é tão essencial ao estudante do presente século, quanto as ciências tradicionais.

Blikstein (2008) complementa destacando que atualmente não basta apenas saber somar, subtrair, ler e escrever. Segundo o autor, para exercer plenamente a cidadania no século XXI, faz-se necessário também a obtenção de diversos outros conhecimentos/habilidades. E dentre estes o autor destaca o “Pensamento Computacional”, que segundo Wing (2006) é uma habilidade indispensável para todos e não apenas para profissionais da área da computação. A referida autora define o Pensamento Computacional como algo muito além da capacidade de saber utilizar um aplicativo de computador e sim como uma habilidade de saber utilizar o computador como um instrumento de aumento do poder cognitivo e operacional humano, o que é capaz de ser convertido em resultados como aumento da produtividade, inventividade e criatividade humana.

¹ Considerada “a segunda geração de serviços *on-line* que se caracteriza por potencializar as formas de publicação, compartilhamento e organização de informações, além de ampliar os espaços para a interação entre os participantes do processo” (PRIMO, 2007, p. 1).

Apesar do pioneirismo de Wing, Seymour Papert também teve uma influência significativa no desenvolvimento do Pensamento Computacional. O trabalho de Papert sobre o Construcionismo e a linguagem de programação LOGO foi fundamental para o desenvolvimento do Pensamento Computacional, uma vez que nele foi apresentado o computador como instrumento capaz de apoiar a construção do conhecimento. Trabalhou-se ainda na hipótese de que a programação de computadores poderia facilitar o processo de resolução de problemas e o aprendizado em diversas áreas [Rodrigues 2015].

Na concepção de Blikstein (2008) o Pensamento Computacional pode ser definido como “saber usar o computador como um instrumento de aumento do poder cognitivo e operacional humano”. Já Furber (2012) preconiza o Pensamento Computacional como o processo de reconhecimento dos aspectos computacionais no mundo que nos cerca, além da aplicação de ferramentas e técnicas da Ciência da Computação para compreender e raciocinar sobre os sistemas e processos naturais e artificiais. Assim, o Pensamento Computacional não deve ter necessariamente como resultado final a produção de *software* e *hardware*. Reconhece ainda que os conceitos fundamentais da Computação estarão presentes para resolver problemas em vários contextos do cotidiano [Barcelos e Silveira 2012].

Para Blikstein (2008) o “pensar computacionalmente” pode ser dividido em duas etapas: a) Identificação das tarefas cognitivas que podem ser executadas com mais rapidez e eficiência fazendo o uso da computação; b) Programação do computador para que este realize as tarefas cognitivas identificadas na etapa anterior. Desse modo, os computadores passam a auxiliar na execução dos processos produtivos, pesquisas científicas, dentre outros, de forma a evitar perda de tempo, por parte das pessoas, com a execução de tarefas repetitivas.

2.1.1 Habilidades do Pensamento Computacional

Segundo Barr e Stephenson (2011) compõe a parte central do Pensamento Computacional nove conceitos que são: Coleta de dados; Análise de dados; Representação de dados; Decomposição de problemas; Abstração; Algoritmos e Procedimentos; Automação; Paralelização e Simulação.

Os mesmos autores também pontuam o que significa cada um desses nove conceitos:

- **Coleta de dados:** pressupõe um processo de coleta de dados ou informações sobre determinado problema de forma adequada;
- **Análise de dados:** é um processo onde se atribui significação aos dados, obtêm-se padrões, além de se tirar conclusões;
- **Representação de dados:** processo onde se representa e se organiza dados através de gráficos, tabelas, imagens, textos ou figuras;
- **Decomposição de problemas:** processo no qual se decompõe certo problema complexo em tarefas menores e mais facilmente gerenciáveis;
- **Abstração:** processo necessário para reduzir a complexidade do problema e para definir ideias principais;
- **Algoritmos e Procedimentos:** sequência de passos necessários para resolução de um problema ou para se atingir determinado objetivo;

- **Automação:** processo que indica a utilização de computadores ou máquinas para realizar tarefas repetitivas;
- **Paralelização:** refere-se à organização de recursos para realizar tarefas, simultaneamente, para alcançar um objetivo;
- **Simulação:** representação ou modelo de um processo. A simulação também envolve experimentos sendo executados usando modelos. A aquisição do PC acontece por meio do estudo de fundamentos de Ciência da Computação, podendo ou não ser integrados a disciplinas de currículos escolares para formação básica.

A CSTA (2011) também apresenta os nove conceitos essenciais para o Pensamento Computacional classificados em três grupos:

- Coleta de dados, análise de dados, representação de dados;
- Decomposição de problemas, abstração, algoritmos;
- Automação, simulação e paralelismo;

Já Karen Brennan e Mitchel Resnick, pesquisadores do MIT - Massachusetts Institute of Technology -, têm realizado nos últimos anos muitos estudos relacionados com o PC no que diz respeito à criação de produtos multimídia. Os resultados dessas pesquisas deram origem ao quadro de referências para estudar e avaliar o desenvolvimento do Pensamento Computacional, e que envolve três dimensões: (i) conceitos computacionais; (ii) práticas computacionais; e (iii) perspectivas computacionais [Sousa e Lencastre 2014].

Brennan e Resnick (2012) também utilizam uma abordagem tripartida para a análise do Pensamento Computacional, contudo, com o foco voltado para práticas de programação. Dessa forma, o framework desenvolvido pelos pesquisadores do Media Lab do MIT é composto pelas seguintes dimensões: conceitos, práticas e perspectivas computacionais.

2.1.1 O que não é o Pensamento Computacional

Ciência da computação é área de conhecimento que estuda a computação, isto é, investiga o que pode ser computado e como pode ser computado. O PC possui diversas características como conceptualização. Entretanto ciência da computação não é sinônimo de programação. Pensar como um cientista da computação envolve mais do que ser capaz de programar um computador [Wing 2006]. Assim, saber programar é uma habilidade dos cientistas da computação e está relacionada como o Pensamento Computacional, todavia o Pensamento Computacional vai muito além disso.

O PC não é saber navegar na internet, enviar e-mail, publicar um blog, operar um processador de texto ou planilha eletrônica. Pensamento Computacional pressupõe a utilização do computador como um instrumento capaz de aumentar o poder cognitivo e operacional humano, ou seja, é usar computadores, e redes de computadores, para expandir a produtividade, inventividade e criatividade [BLIKSTEIN 2008]. Inventar, produzir e criar são ações inerentes ao PC, pois permitem ao indivíduo um protagonismo na sua vida em sociedade. Um indivíduo capaz de inventar coisas novas ou se reinventar como pessoa é um indivíduo mais apto a sobreviver numa época onde mudanças profundas ocorrem a todo o momento.

Saber manusear uma ferramenta de escritório como processador de texto ou planilha, ou até mesmo outros aplicativos não significa que a pessoa está “pensando computacionalmente”, porque o Pensamento Computacional propõe ir além do simples executar algo no computador. Contudo, se a pessoa utiliza o programa computacional, como o processador de texto, por exemplo, pra criar, inventar ou aumentar seu conhecimento, ela passa então a “pensar computacionalmente”.

Pensamento Computacional é uma forma para as pessoas resolverem problemas; não é tentar fazer com que seres humanos pensem como computadores. Computadores são tediosos e enfadonhos; humanos são espertos e imaginativos. Os seres humanos transformam a computação em algo empolgante [Wing 2006].

2.1.3 Pensamento Computacional no ensino

Nos últimos anos os Estados Unidos e alguns países da Europa introduziram o ensino da computação na grade curricular. Todavia, França e Tedesco (2015) observam que o mesmo não acontece ainda no Brasil, onde este ensino é ofertado, mais especificamente, à estudantes das áreas de tecnologias (dos níveis técnico e superior). Com exceção para algumas iniciativas envolvendo ensino médio e fundamental.

Dentre estas iniciativas França e Tedesco (2015) destacam algumas olimpíadas de programação, cursos de lógica de programação fazendo uso do *App Inventor* (www.appinventor.mit.edu), projetos visando o desenvolvimento do raciocínio lógico, peças teatrais e construção de jogos digitais.

No Brasil, nas abordagens utilizadas que visam o desenvolvimento do Pensamento Computacional, utilizando-se a lógica de programação como um meio, destacam-se as ferramentas visuais de programação. Dentre as quais sobressaem-se o Scratch, *App Inventor*, *AgentSheets* e *RoboMind*, conforme destacam França e Tedesco (2015).

Neste trabalho optou-se pelo uso de Tutoriais Interativos e da ferramenta Scratch. Esta, foi escolhida pois, além de ser gratuita, é bastante conhecida e possui uma excelente documentação, o que possibilita aos usuários iniciantes melhor compressão da ferramenta. Além disso, ela funciona tanto *online* quanto *off-line*, facilitando a utilização por usuários que ainda não dispõem de acesso à internet. Por sua vez por meio de Tutoriais Interativos é possível apresentar e ensinar a utilizar este ambiente de forma lúdica, aumentando o interesse pelo conteúdo e facilitando a aprendizagem.

2.2. Tutorial Interativo

Esta seção apresenta a definição do que são Tutoriais Interativos, bem como suas principais características.

Tutorial pode ser definido como um programa que busca instruir, podendo envolver leitura, conexão com outros sítios, perguntas e testes [Francisco e Oliveira 2006]. A instrução através do tutorial visa potencializar o processo de construção do conhecimento. Para tanto, o aprendiz terá que ler e exercitar o conhecimento adquirido. A conexão com outros sítios disponíveis na internet, pode apresentar informações auxiliares que nortearão no processo de ensino-aprendizagem.

Tutorias também são textos que visam a instrumentalização de todo indivíduo que está envolvido em uma atividade que irá exigir determinada prática para atuar em uma área específica [Santiago 2010]. Vale lembrar que eles podem ser utilizados para fins de ensino ou não, e também, nas mais diversas áreas.

São também ensinamentos transmitidos na internet que possuem a finalidade de ensinar como se realiza alguma tarefa através da explicitação do seu desenvolvimento em etapas [Reis, Romão e Leite 2011]. Dessa forma, nota-se o “passo-a-passo”, isto é, a sequência de etapas que é inerente ao tutorial. É importante que a descrição dos passos sejam claros e objetivos, para que o aprendiz, ao tentar reproduzir a sequência, consiga facilmente executá-las.

A sociedade da qual fazemos parte possui um caráter extremamente imediatista e o tutorial visa atender estas demandas desse tipo de sociedade. Ele pode permitir a aquisição de conhecimento de uma maneira mais rápida e lúdica do que por um meio “mais institucionalizado”. O tutorial também só é possível numa configuração sociotécnica como a da cibercultura. Porque é nesta configuração onde tecnologias e demandas sociais resultam em conexão colaborativa e em todas as características do presenteísmo pós-moderno [Reis, Romão e Leite 2011].

Como afirma os autores, a sociedade atual possui diversas singularidades que décadas atrás eram inimagináveis. Dentre estas, pode-se citar o advento e popularização das Tecnologias da Informação e Comunicação (TIC). Dessa forma, hoje encontra-se disponíveis, nos mais variados sítios, recursos computacionais que podem ser utilizados para enriquecer e auxiliar no processo de ensino-aprendizagem.

Portanto, tutorial pode ser considerado como um mecanismo informal de auxílio na aquisição de conhecimento e informações. Destaca-se seu caráter didático e pormenorizado, uma vez que descreve uma sequência de passos.

Os programas tutoriais são uma versão computacional da instrução programada. Eles dispõem de características que não são permitidas no papel como: animação, som e a manutenção do controle da performance do aprendiz. São muito utilizados devido permitirem a introdução do computador na escola sem provocar muita mudança, pois representa uma versão computadorizada do que já ocorre na sala de aula [Valente 1993]. Dentre as características, ressalta-se a animação, que permite mostrar de forma diferente da tradicional, como a informação sobre determinado conteúdo pode ser apresentada.

Vale lembrar que na internet há predominância de informalidade na produção de tutoriais, uma vez que se pode encontrar em sítios como o Youtube (<http://youtube.com>) diversos vídeos tutoriais sobre os mais variados assuntos, feitos por quaisquer pessoas independentemente de ter um conhecimento científico. Em outros domínios que envolvem tecnologia e o mundo digital, também pode-se encontrar uma grande gama de tutoriais, desde um tutorial sobre remover e/ou instalar um programa a um tutorial sobre como fazer um bolo. Assim, qualquer indivíduo, considerando seus conhecimentos e experiências, é capaz de produzir um tutorial. Todavia, a qualidade deste vai depender da sua capacidade de expressão e conhecimento das características dessa ferramenta.

Quanto ao termo interatividade, este é um fenômeno da “sociedade da informação”. Tem origem nos anos de 1970 e ganha notoriedade a partir dos anos de 1980 [Silva 2010]. Atualmente, é muito comum ouvir que determinado produto é interativo, ou seja, dotado de interatividade.

Alguns elementos necessários às tecnologias interativas são a participação, a intervenção, a bidirecionalidade e a multiplicidade de conexões [Silva 2010]. Dessa forma, pode-se afirmar que um tutorial interativo, como produto dotado de interatividade, deve apresentar características como participação, intervenção, bidirecionalidade e multiplicidade de conexões.

Na interatividade é essencial a relação do emissor com o receptor. Deve haver a participação-intervenção do receptor, onde participar vai muito além do responder sim ou não, ou de apenas escolher uma opção dada. Essa participação tem que ser capaz de modificar e interferir na mensagem. A comunicação existente é produção conjunta da emissão e da recepção, na qual o emissor é receptor em potencial e o receptor é emissor em potencial o que significa que ambos codificam e decodificam [Silva 2010].

Ressalta-se o papel do emissor e do receptor na utilização do tutorial interativo. O receptor deve se preocupar com quem será o seu receptor e como este poderá se comportar.

Em vista do que foi mencionado, conclui-se que no tutorial interativo deve haver a possibilidade do receptor participar através da navegação ao longo deste recurso e poder intervir. Essa intervenção, por exemplo, pode se dar quando o receptor escolhe a forma como vai usar o tutorial.

2.3. Scratch

Esta seção aborda sobre o *software* Scratch e sua utilização na disseminação do Pensamento Computacional.

O ensino da programação e da computação é importante pois permite que o indivíduo compreenda como são criados os sistemas operacionais com os quais ele interage e assim ele passa a entender as intenções deste sistema [Omine 2014]. Na concepção de Omine (*apud* Lemos, 2013) o conhecimento da programação é essencial, inclusive para a manutenção da democracia.

Steve Jobs, durante uma entrevista em 1995, afirmou que todos deveriam aprender a programar uma linguagem computacional, pois ensina a pensar. “É como ir para a faculdade de direito. Não acho que todos devem ser advogados, mas acho que fazer direito pode ser útil, pois ensina a pensar de um certo modo”, defendeu ele [Steve Jobs: The Lost Interview, 2012].

Na concepção de Rodrigues (2018) o mundo está ingressando na revolução industrial 4.0 e isso causará mudanças na economia e em especial no mercado de trabalho, com a automação de muitos trabalhos que atualmente exigem esforços repetitivos. Diante deste cenário, segundo o autor, duas competências serão fundamentais: resolver problemas complexos e gerar resultados a partir do uso de tecnologia. Caso contrário a sociedade estará diante de um abismo social entre os que possuírem estas competências e aqueles que não aprenderam a usar as tecnologias, muito menos a pensar de forma autônoma. Rodrigues (2018) alerta ainda para a urgência da existência de um sistema que familiarize as pessoas às tecnologias, pois não saber programar será o analfabetismo do futuro. Para ele é preciso que as crianças aprendam a programar nas escolas e que os adultos também sejam inseridos neste processo de desenvolvimento, uma vez que a expectativa de vida destes é cada vez maior.

Todavia, grande parte dos usuários têm dificuldade durante o processo de aprendizagem de programação, especialmente os que são iniciantes, pois ainda não dominam a lógica de programação e não possuem conhecimento sobre algoritmos. Diante disso, desde o início da década de 70 pesquisadores buscam desenvolver linguagens e ambientes de programação mais compreensíveis para um maior número de pessoas, deixando a linguagem de programação um pouco mais próximo da linguagem natural [Scaico et al. apud Paush e Kelleher 2005].

Neste sentido o Scratch destaca-se por utilizar o conceito de desenvolvimento orientado ao *design*. Nesta ferramenta, o iniciante não precisa dominar a sintaxe para ter suas primeiras experiências com programação. Isso porque o Scratch utiliza o que chama-se de programação em blocos. Ou seja, para criar seus projetos, o usuário não precisa digitar linhas de código. Em vez disso, ele vai arrastando blocos de código e criando suas histórias, jogos, animações e outros. Este projeto pode ser iniciado “do zero”, recomendado para usuários que já possuem uma certa experiência, ou pode ser criado a partir de outros projetos criados e disponibilizados por outros usuários.

O Scratch é uma ferramenta de programação, disponibilizada no *link* <www.scratch.mit.edu> (ver Figura 1), que permite que seus usuários programem suas histórias, jogos e animações interativas por meio de uma interface gráfica utilizando blocos. Tal ferramenta faz parte de um projeto surgido em 2007 e mantido pelo *Lifelong Kindergarten* do grupo *Massachusetts Institute of Technology (MIT) Media Lab* [LIFELONG KINDERGARTEN GROUP 2007].

Segundo o Lifelong Kindergarten Group (2007), o Scratch foi concebido especialmente para a faixa etária de 8 a 16 anos de idade, todavia atualmente é usado por pessoas de todas as idades (ver Gráfico 1). Milhões de usuários criam projetos Scratch numa grande variedade de contextos, incluindo lares, escolas, museus, bibliotecas e centros comunitários” [Lifelong Kindergarten Group 2007].

Atualmente o Scratch está disponível em mais de 150 países (Figura 2) e em 40 línguas e possui mais de 32 milhões de projetos² que foram criados por seus 28 milhões de usuários e compartilhados para que outras pessoas possam reutilizá-los, dando origem a novas histórias. Desse total de usuários, 1,39% são brasileiros, o que equivale a mais de 413 mil usuários. Este número ainda é baixo, quando comparado com os Estados Unidos que possui mais de 13 milhões de usuários. Todavia, é importante ressaltar que o Brasil é um dos países da América Latina com a maior quantidade de usuários de Scratch.

Ademais, é importante ressaltar que o número de usuários do Scratch cresce a cada ano no mundo inteiro, conforme mostra o Gráfico 2. Portanto, é uma ferramenta que a cada dia torna-se mais popular e, conseqüentemente, cresce o número de projetos que são disponibilizados, facilitando assim, para que novos usuários façam uso destes durante o processo de iniciação ao uso da ferramenta.

Utilizando o Scratch, o usuário além de aprender a lógica de programação, também pode desenvolver outras habilidades como o pensamento criativo, análise crítica e aprendizagem contínua. Isso porque esta ferramenta é embasada no

² Segundo dados disponíveis no *hiperlink* <https://scratch.mit.edu/statistics>. Acesso em 14 de Junho de 2018.

construcionismo, uma vez que a mesma foi desenvolvida a partir da linguagem LOGO (criada por Seymour Papert). Isso faz com que o Scratch se apresente como um importante aliado para o ensino-aprendizagem, com vistas ao desenvolvimento do Pensamento Computacional.

Ao manuseá-lo, o docente pode fazer uso de várias possibilidades de utilizações pedagógicas, podendo realizar atividades elaboradas de acordo com o conteúdo programático de certa disciplina, com a faixa etária dos estudantes, com o contexto histórico-cultural e etc. No presente trabalho, optou-se por adotar a referida ferramenta, juntamente com Tutoriais Interativos, no processo de ensino-aprendizagem, com o objetivo de desenvolver o Pensamento Computacional.



Figura 1: Página inicial do website do Scratch. Fonte: <https://scratch.mit.edu>. Acesso em 01 de Julho de 2018

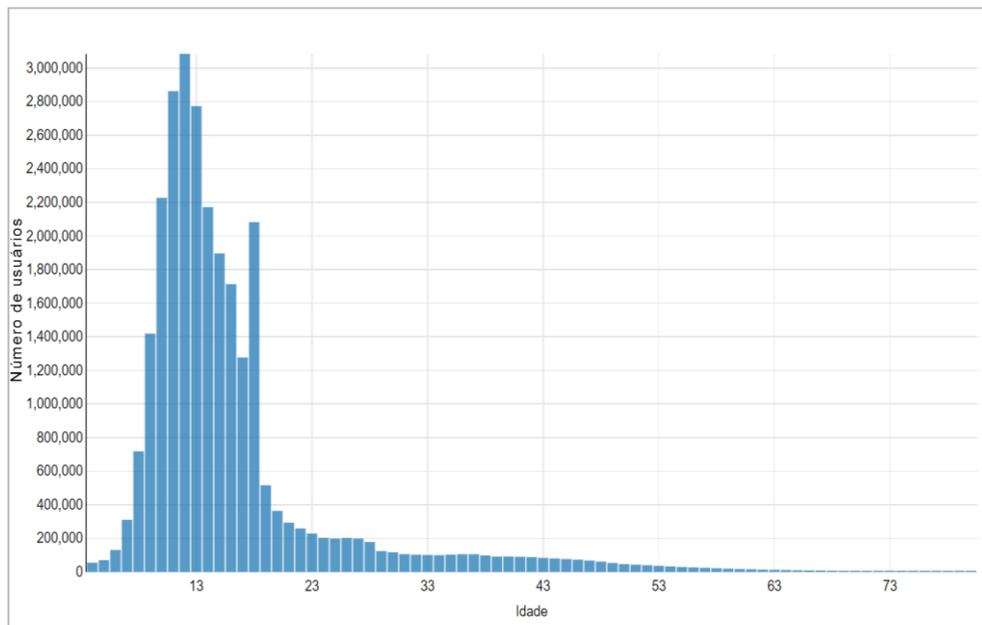


Gráfico 1: Distribuição por idade dos novos usuários de Scratch. Fonte: www.scratch.mit.edu. Acesso em 01 de Setembro de 2018.

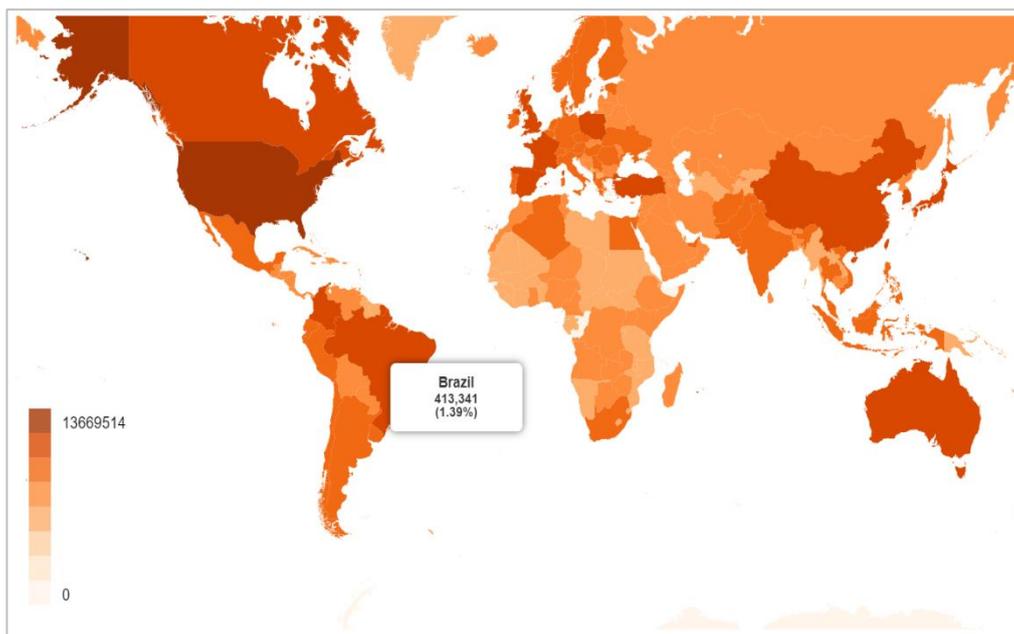


Figura 2: Distribuição dos usuários de Scratch no mundo, com destaque para a quantidade de usuários no Brasil. Fonte: www.scratch.mit.edu. Acesso em 01 de Setembro de 2018

Além do mais, a ferramenta apresenta inúmeras vantagens aos discentes, pois a mesma possibilita que seus usuários absorvam vários conhecimentos enquanto criam seus projetos. Alguns destes conhecimentos estão elencados no artigo “*Learning with Scratch*”³, dentre os quais destacam-se:

³ Learning with Scratch – the Scratch Documentation Website - MIT. Disponível em: <https://llk.media.mit.edu/scratch/Learning-with-Scratch.pdf>. Acesso em 01 de Agosto de 2018.

- **Conhecimento matemático e computacional** – enquanto criam seus projetos os usuários absorvem os conhecimentos básicos da computação, tais como iteração e condicionais. Além destes os usuários também aprendem a lidar com conceitos matemáticos como coordenadas, variáveis e números aleatórios;
- **Processo do “design”** – durante a criação de um projeto no Scratch o usuário cria uma nova ideia. Posteriormente ele faz um protótipo desta ideia, realiza testes, executa a depuração quando surgem erros, além da possibilidade de receber comentários de outros usuários. Isto faz com que o usuário revise o seu projeto por diversas vezes durante o processo criativo. Essa sequência de passos constituem um ciclo contínuo (ver Figura 3) onde um projeto nasce a partir de uma ideia e este dar origem à outros projetos e assim sucessivamente. Com isso ao utilizar o Scratch o usuário está em constante processo de aprendizagem e construção do conhecimento. Por sua vez este processo auxilia no desenvolvimento de cidadãos mais críticos, que tenham habilidade de comunicar-se, de realizar análises sistemáticas, colaborar efetivamente, além de aprender de forma contínua.
- **Maior “fluência” com a tecnologia digital** – ser fluente em tecnologia digital não significa saber realizar tarefas básicas em um computador, com enviar um *e-mail* ou digitar um texto. E sim utilizar o computador para criar algo com ele.

Em vista do que foi mencionado, pode-se afirmar que o Scratch apresenta um grande potencial para fins de ensino. Além do mais, a popularização da ferramenta, somada à sua facilidade de acesso e uma vasta documentação, fez com que a mesma se tornasse uma importante aliada no processo de ensino-aprendizagem. Segundo o Lifelong kindergarten group (2007) diversos educadores têm utilizado a ferramenta no ensino básico, secundário e até mesmo em universidades. Essa utilização tem sido aplicada nas mais diversas áreas do conhecimento. Seja para o ensino-aprendizagem de ciências, matemáticas, línguas estrangeiras, artes, estudos sociais e letras.

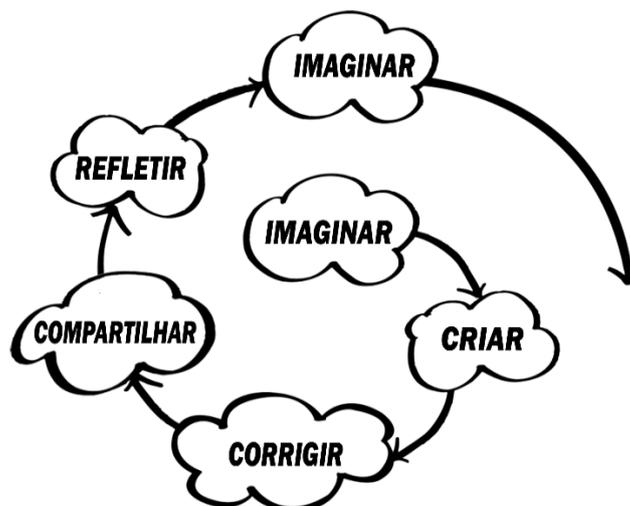


Figura 3: Ciclo de criação no Scratch³

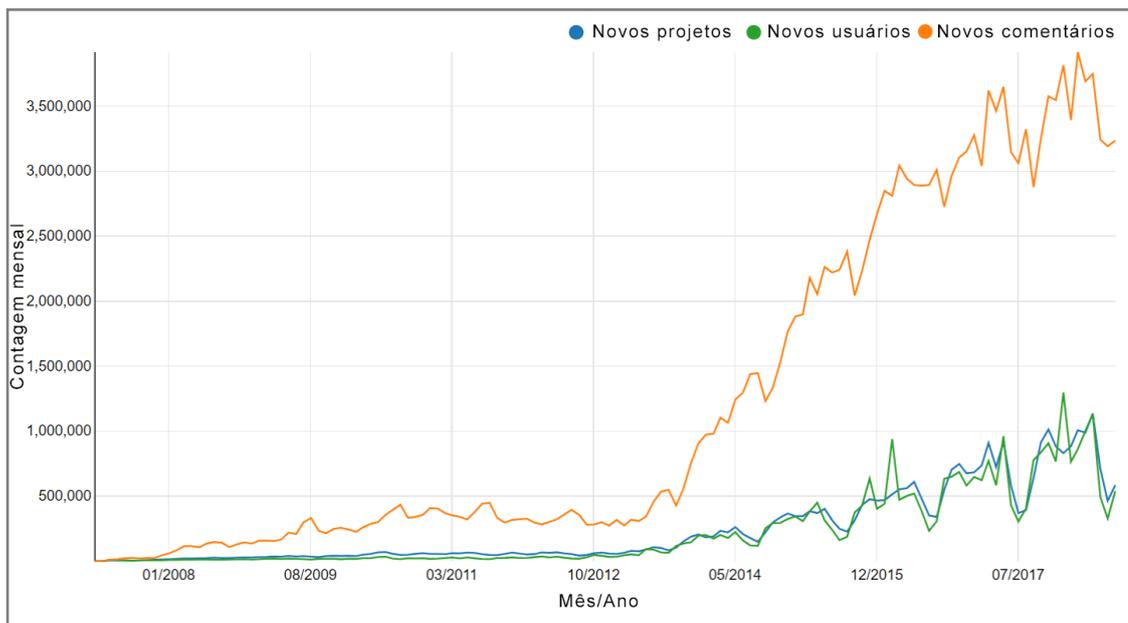


Gráfico 2: Atividades mensais no website no Scratch. Traduzido pelos autores.
Fonte: <https://scratch.mit.edu/statistics/>. Acesso em 01 de Agosto de 2018.

3. O uso de Tutoriais Interativos no desenvolvimento do “pensar computacionalmente”

Nesta seção será abordada de que maneira pode ser elaborado um Tutorial Interativo (TI), apresentando suas telas principais e como atribuir interatividade a esta ferramenta.

Para elaboração dos tutoriais, podem ser usados programas como o *Microsoft Office Power Point* e *Microsoft Paint*, que são *softwares* que estão disponíveis nos computadores de muitas pessoas. A utilização do *Power Point* é recomendada porque possui recursos como botões de ação, *hiperlinks*, dentre outros, que possibilitam atribuir interatividade aos TIs. Já o *Paint* é interessante para tratamento das imagens que serão utilizadas no tutorial. Todavia, é importante destacar, que caso o usuário não disponha destes recursos computacionais, por estes não serem de licença livre, o mesmo pode fazer uso de outras ferramentas que possibilitem a criação de apresentações (similares às criadas no *Power Point*) e edição de imagem (tal qual é possibilitado pelo *Paint*).

A Página de Apresentação, como pode ser observada na Figura 4, será a primeira tela que o usuário visualizará. Para tanto, devem ser colocadas informações como o título do tutorial interativo, botão *play* e nome do autor ou autores. O título deverá ser objetivo e sucinto. O botão *play* servirá para o usuário prosseguir para a próxima tela.

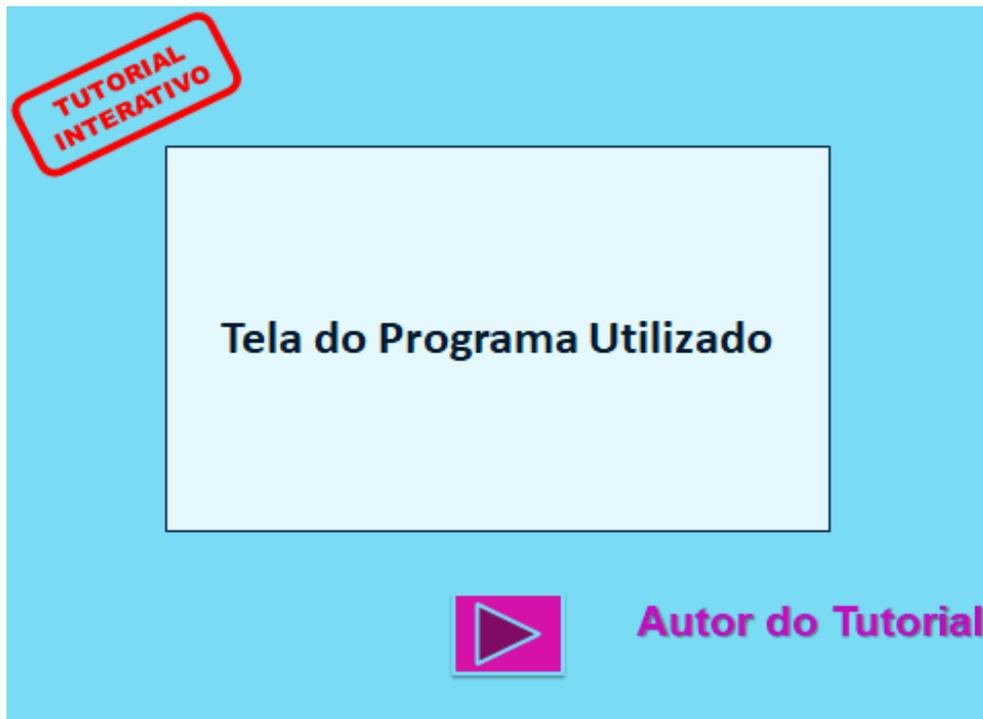


Figura 4: Página de Apresentação. Fonte: Autores (2018)

Na Página Inicial, apresentada na Figura 4, deverá apresentar o título do tutorial, além de três botões localizados na parte inferior da tela: Como Usar, Menu Principal e Ajuda. Estes serão úteis para facilitar a exploração do ambiente, permitindo uma rápida navegação.

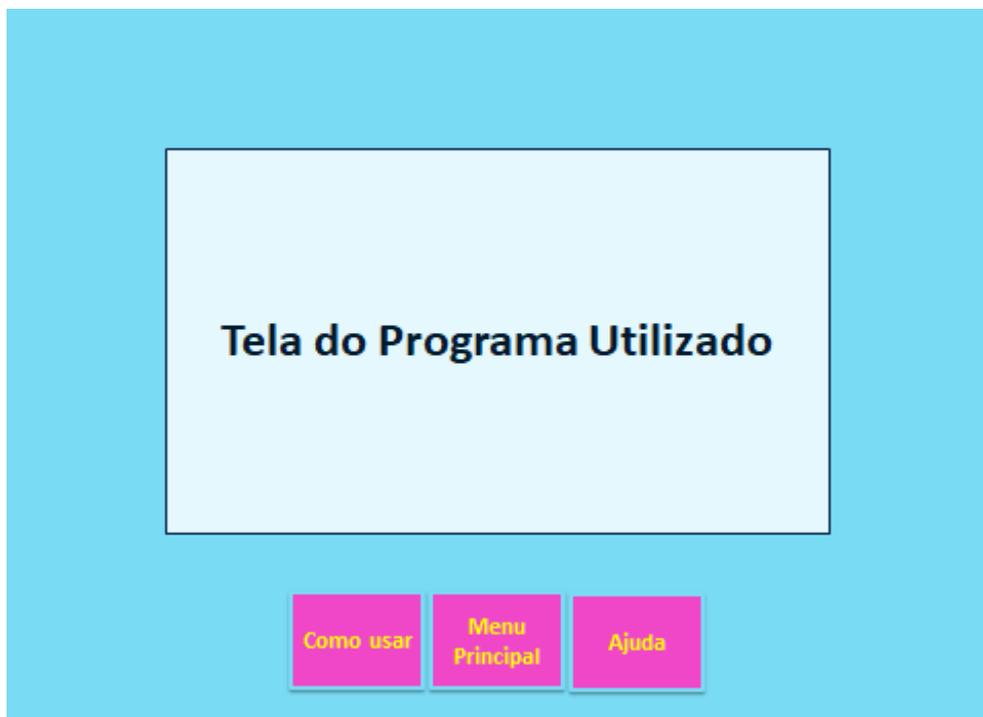


Figura 5: Página Inicial. Fonte: Autores (2018)

Na Página Como Usar, mostrada na Figura 6, deve-se apresentar botões de atalho na parte superior direita da tela, o título da página, balões explicativos e botões voltar e avançar. Estes permitirão também uma navegação rápida e eficiente pelo ambiente. Os balões explicativos servem para informar o que determinada funcionalidade faz.



Figura 6: Página Como Usar. Fonte: Autores (2018)

No Menu Principal, tela apresentada na Figura 7, há botões de atalho, título da página, botões voltar e avançar, além dos *links* das principais partes da ferramenta.



Figura 7: Página Menu Principal. Fonte: Autores (2018)

Em O que é o Programa, Figura 8, pode-se expor sobre o que é o *software* escolhido, suas funcionalidades e possibilidades de uso.

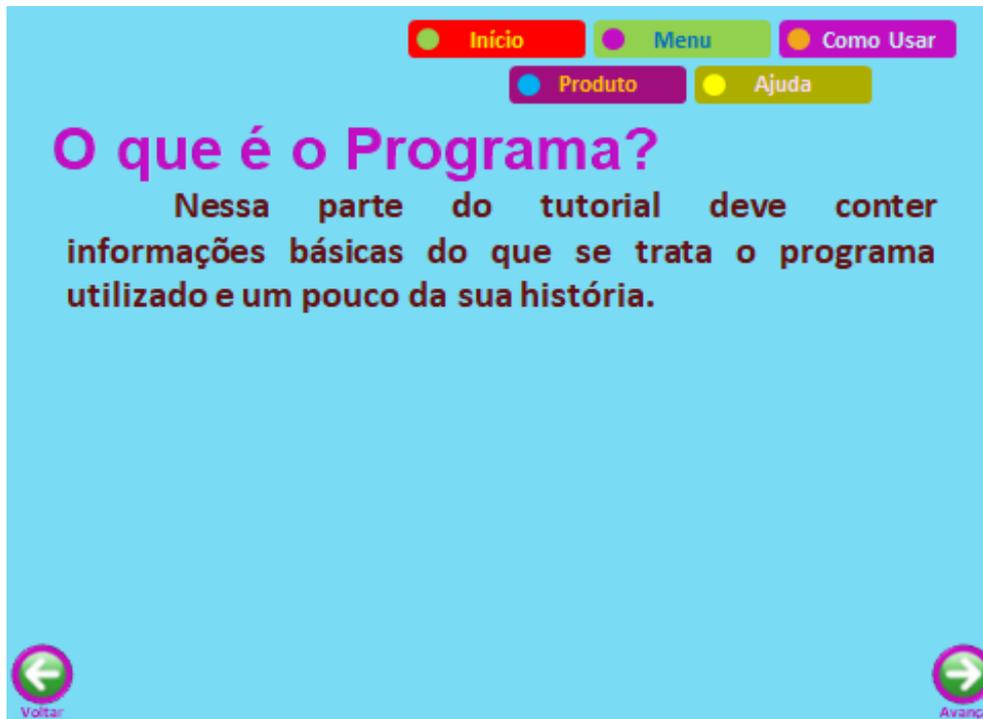


Figura 8: Página O que é o Programa. Fonte: Autores (2018)

Em Baixando o Programa, Figura 9, é interessante fornecer informações de onde se pode baixar o programa.

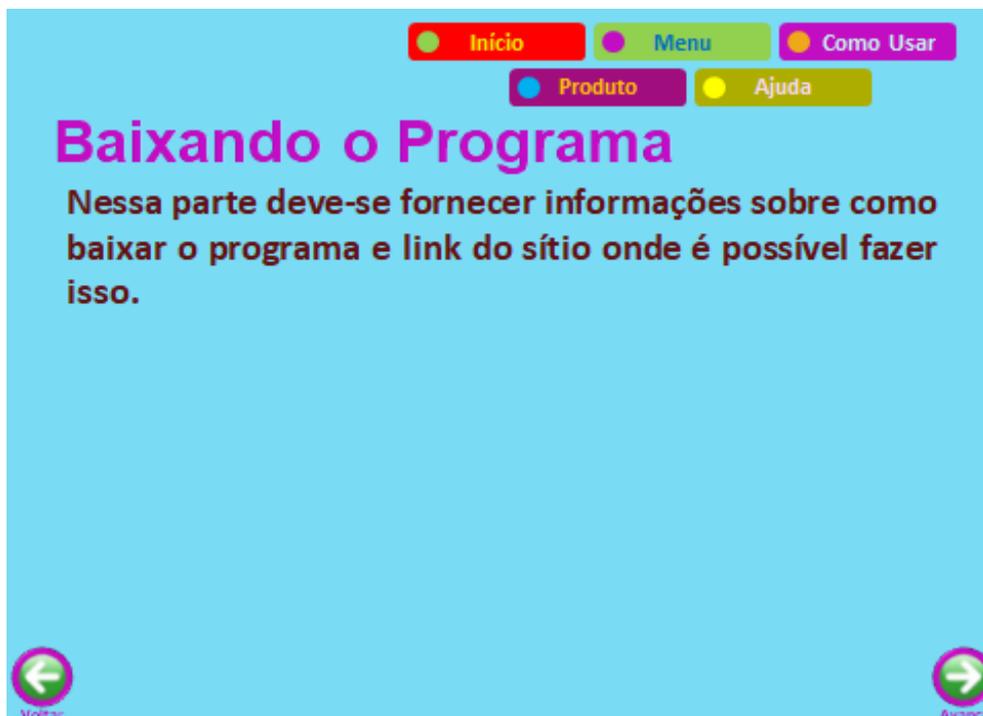


Figura 9: Página Baixando o Programa. Fonte: Autores (2018)

Já em Instalando o Programa, Figura 10, é necessário destacar em quais sistemas operacionais o programa pode ser instalado, quais requisitos são necessários, dentre outros.

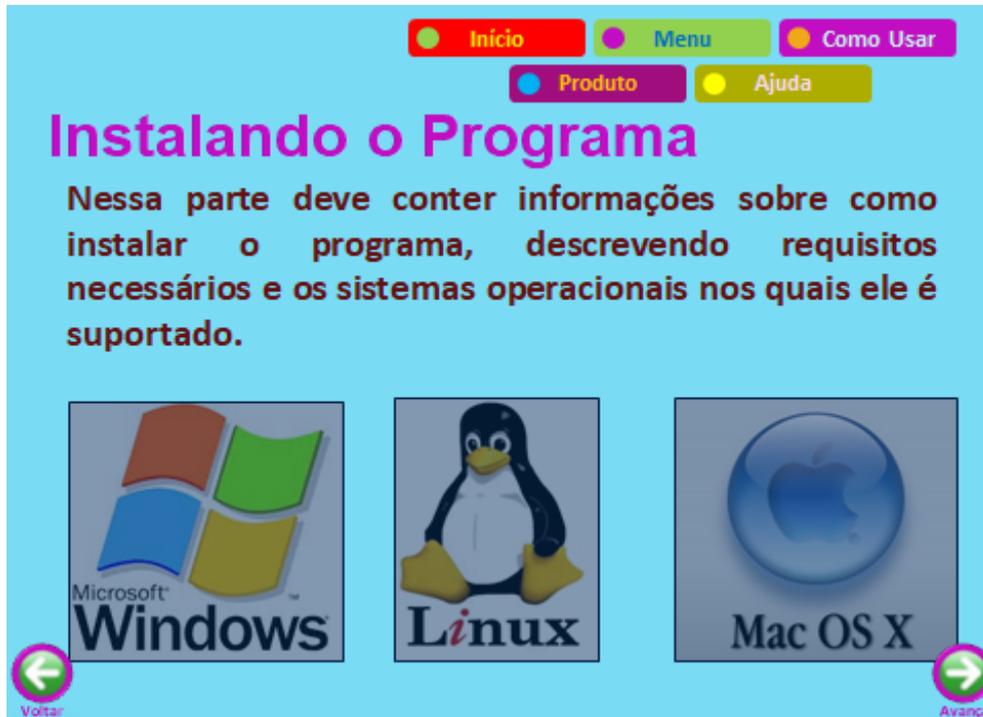


Figura 10: Página Instalando o Programa. Fonte: Autores (2018)

Na Página Conhecendo o Ambiente, Figura 11, o usuário poderá conhecer de forma mais detalhada as funcionalidades do programa. Sugere-se a inserção de imagens ilustrativas do ambiente, bem como animação de como utilizar as funcionalidades.



Figura 11: Conhecendo o Ambiente. Fonte: Autores (2018)

Criando o Produto, Figura 12, é uma das partes mais importantes do tutorial interativo, porque deve apresentar como é elaborado o produto do programa. Por exemplo, se for um programa de planilha eletrônica, deverá elencar os passos

necessários para se criar uma planilha, se for um editor de simulação deverá mostrar as etapas para se utilizar a simulação.

Em Ajuda, Figura 13, há informações para sanar dúvidas do usuário quanto à utilização do tutorial.



Figura 12: Página Criando o Produto. Fonte: Autores (2018)



Figura 13 – Página Ajuda. Fonte: Autores (2018)

4. Proposta de Desenvolvimento de um Tutorial Interativo

A seguir são apresentadas propostas de desafios para nortear e estimular o desenvolvimento de um tutorial interativo. Destaca-se que o recurso computacional utilizado para a construção do tutorial será o Scratch. As figuras 14 a 26 ilustram o resultado esperado das atividades realizadas nos desafios a seguir.

4.1 Desafios

Embora os desafios estejam relacionados ao Scratch, eles podem ser aplicados aos mais diversos recursos computacionais.

4.1.1 Desafio 1

1. Analise a ferramenta Scratch;
2. Verifique as partes que compõe o ambiente;
3. Pesquise informações adicionais sobre o Scratch;
4. Visite e explore a aplicação Teste de Matemática 2 (Figura 14), disponível em <https://scratch.mit.edu/projects/3295166/#editor>

É importante a análise do recurso que se pretende utilizar, porque através disso, o desenvolvedor terá uma visão geral de como o programa funciona. A partir de então, poderá ser feita a reflexão de como o TI será estruturado e de como planejá-lo. Esse planejamento influenciará a qualidade do tutorial. Ao visite a aplicação Teste de Matemática, espera-se que o usuário tenha algumas noções de como se organiza um código no ambiente Scratch.

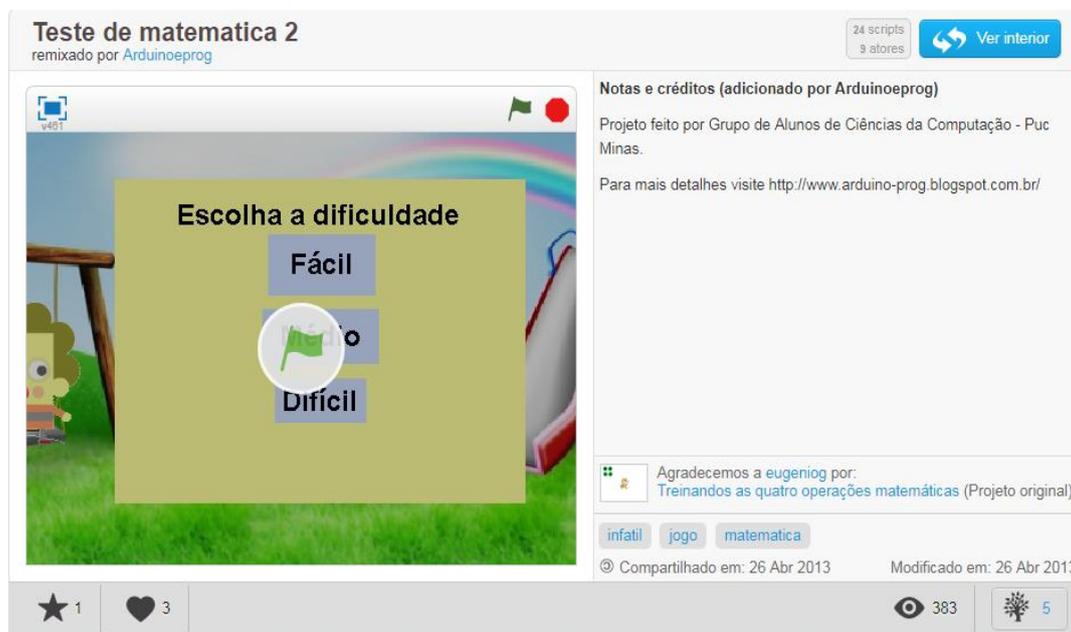


Figura 14: Teste de matemática. Disponível em: <https://scratch.mit.edu/projects/3295166/#editor>. Acesso em 10 de Setembro de 2018

4.1.2 Desafio 2

1. *Desenvolva a Página de Apresentação com as seguintes partes: nome do autor do tutorial, imagem da tela do ambiente e botão play;*

A página de apresentação será a primeira tela que o usuário terá contato. Dessa forma, há a necessidade de despertar a atenção e desejo do sujeito em utilizar o tutorial. Essa tela é essencial, uma vez que através dela, as primeiras informações serão repassadas ao usuário.



Figura 15: Página de Apresentação. Fonte: Autores (2018)

4.1.3 Desafio 3

1. *Crie a Página Inicial que deverá abarcar os botões Como Usar, Menu Principal e Ajuda, além da imagem do recurso computacional utilizado;*

A página inicial, ao apresentar esses três botões, facilita a navegação pelo TI.



Figura 16. Início. Fonte: Autores (2018)

4.1.4 Desafio 4

1. Elabore as telas Como Usar, incluindo os botões de atalho, balões explicativos, imagens das funcionalidades explicadas e botões voltar e avançar;



Figura 17. Como Usar. Fonte: Autores (2018)

Diferentemente de outras partes do tutorial, que apresentam apenas uma tela, o Como Usar traz diversas telas, dependendo apenas da intenção de quem está desenvolvendo o tutorial.

4.1.5 Desafio 5

1. Produza o Menu Principal para facilitar a navegação pelo tutorial. Para isso, insira os botões de atalho, título da página, botões voltar e avançar, além dos links das principais partes do ambiente;

Por apresentar as principais partes que o tutorial está organizado, pode-se afirmar que é uma das telas mais importantes e que precisa ser bem elaborada.



Figura 18: Menu Principal. Fonte: Autores (2018).

4.1.6 Desafio 6

1. Faça a tela O que é o Programa, destacando as principais informações sobre o ambiente;

Essa tela é relevante devido ser através dela que serão mostradas noções do potencial do recurso escolhido.

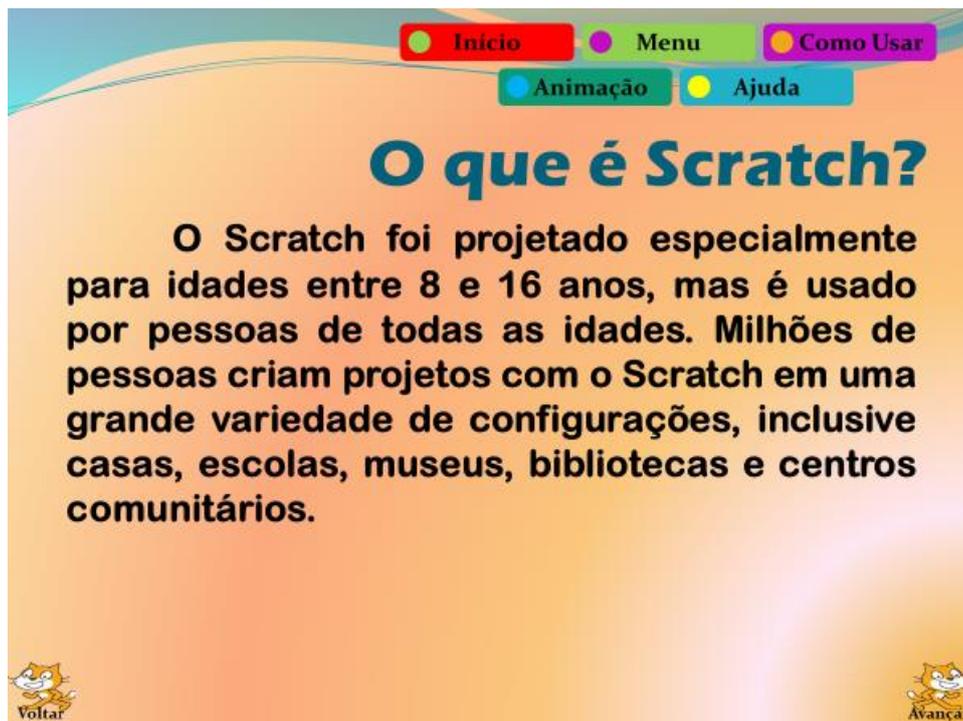


Figura 19: Menu Principal. Fonte: Autores (2018)

4.1.7 Desafio 7

1. Informe em Baixando o Programa o sítio onde o recurso está disponível para ser baixado;



Figura 20: Ensinando como baixar o Scratch. Fonte: Autores (2018)

Nessa parte deve conter indicações de onde pode-se baixar o programa. Isso é relevante pois, muitas das vezes, o usuário não tem o software instalado no seu computador. Vale ressaltar também que alguns ambientes podem ser utilizados *online*.

4.1.8 Desafio 8

1. *Apresente em Instalando o Programa como o usuário poderá proceder com a instalação do software;*

Após baixar o recurso computacional, torna-se primordial saber como instalá-lo. Como algumas instalações não são tão simples e intuitivas, deve-se informar quais recursos extras são necessários e em quais sistemas operacionais o programa pode ser instalado.



Figura 21: Instalando o Scratch. Fonte: Autores (2018)

4.1.9 Desafio 9

1. *Para a tela Conhecendo o Ambiente do Programa, demonstre quais as principais funcionalidades da aplicação;*

As funcionalidades mais imprescindíveis devem ser destacadas, para quando o usuário for criar o produto, ele terá conhecimento mínimo de como proceder.



Figura 22: Conhecendo o Scratch. Fonte: Autores (2018)



Figura 23: Exemplo de uma funcionalidade do programa. Fonte: Autores (2018)

4.1.10 Desafio 10

1. Ao elaborar a tela Criando um Produto Prático do Programa, apresente as etapas necessárias para a criação do produto que pode ser desenvolvido através do ambiente;

Sugere-se a apresentação de uma sequência de passos numerada para que o usuário saiba qual etapa ele deve executar antes de passar para a próxima.

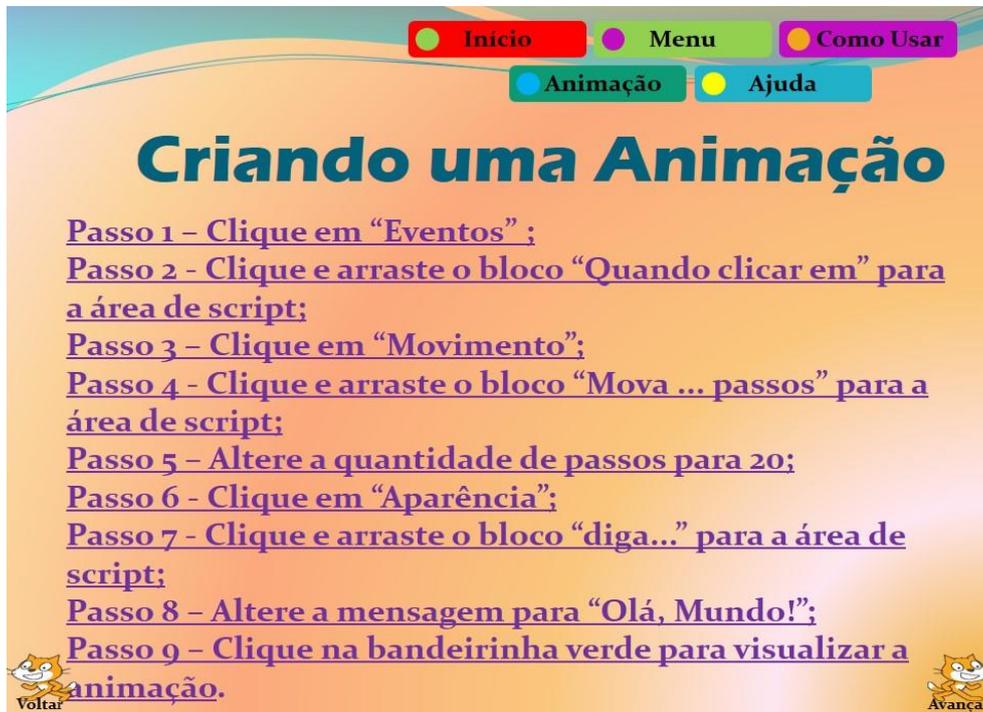


Figura 24: Criando uma Animação. Fonte: Autores (2018)



Figura 25: Exemplo de uma etapa. Fonte: Autores (2018)

4.1.11 Desafio 11

1. *Implemente a parte Ajuda com a finalidade de fornecer ao usuário, informações capazes de sanar possíveis dúvidas.*

Como, ao longo da exploração do ambiente, o utilizador da ferramenta pode ficar em dúvida sobre algo, essa parte deve prever as possíveis incertezas que poderão surgir.



Figura 26: Ajuda. Fonte: Autores (2018)

4.1.12 Desafio 12

1. *Verifique se os links estão funcionando corretamente, bem como as animações e botões de ação.*

Para finalizar a construção do tutorial, é necessário revisar todo o tutorial com a finalidade de se verificar se há algum erro e, conseqüentemente, corrigi-lo.

5. Considerações Finais

Durante a elaboração deste ficou evidente a importância da disseminação do Pensamento Computacional, uma vez que este é tão importante para o indivíduo do presente século.

Sabe-se que em algumas escolas do Brasil a disciplina de “Informática” já está presente. Todavia, é preciso que algo a mais seja feito para que as aulas possam ir além de ensinar como utilizar programas de computadores, tais como editores de texto, de imagem ou navegar na internet. É necessário que hajam iniciativas no sentido do ensino da computação enquanto ciência, de modo que o aluno consiga programar o computador e utilizá-lo como um “agente” potencializador de suas habilidades.

Entretanto, ensinar sobre o Pensamento Computacional ainda pode ser um desafio no Brasil, seja por falta de laboratórios - em várias escolas, ou até mesmo

devido este ainda não está na grade curricular. Dessa forma, parte dos professores tem se utilizado do espaço de outras disciplinas para disseminar este conhecimento tão importante.

Neste sentido, o Scratch apresenta-se como uma excelente ferramenta, pois a mesma representa um suporte multidisciplinar, permitindo que o docente trabalhe com os mais variados assuntos, sem, contudo, deixar de lado o ensino da lógica de programação e o desenvolvimento do Pensamento Computacional.

E aqui convém destacar a importância dos Tutoriais Interativos para facilitar este processo de ensino. Os TI facilitam no processo comunicativo, fazendo com que o conteúdo fique mais dinâmico e mais atrativo.

Portanto, estas duas ferramentas podem ser importantes aliadas dos educadores na disseminação do Pensamento Computacional. São de fácil acesso e não exigem que o educador tenha um grande conhecimento em informática para manuseá-las. Além disso, podem ser utilizadas para o ensino-aprendizagem de quase todos os conteúdos, das mais diversas disciplinas, nos mais diversos níveis de escolaridade.

Referências

- Barcelos, T. S., & Silveira, I. F. (2012) “Pensamento computacional e educação matemática: Relações para o ensino de computação na educação básica”. In XX Workshop sobre Educação em Computação, Curitiba. Anais do XXXII CSBC (Vol. 2, p. 23).
- Barr, V., & Stephenson, C. (2011) “Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?”. *Acm Inroads*, 2(1), 48-54.
- Blikstein, P. (2008) “O Pensamento Computacional e a reinvenção do computador na educação”.
- Brennan, K., & Resnick, M. (2012) “New frameworks for studying and assessing the development of computational thinking”. In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada (Vol. 1, p. 25).
- Charlot, B. (2008) “O professor na sociedade contemporânea: um trabalhador da contradição”. *Revista da FAEEBA–Educação e Contemporaneidade*, Salvador, 17(30), 17-31.
- Francisco, A. C.; Oliveira, A. C. (2006) “Educação à distância no ensino superior: constatações e recomendações”. In XIII SIMPEP, 06 a 08 de novembro de 2006, Bauru, SP, Brasil.
- França, R. S., Ferreira, V. F. S., Almeida, L. D., & Amaral, H. D. (2014) “A disseminação do pensamento computacional na educação básica: lições aprendidas com experiências de licenciandos em computação”. In Anais do XXII Workshop sobre Educação em Computação (WEI-CSBC).
- França, R., & Tedesco, P. (2015) “Desafios e oportunidades ao ensino do pensamento computacional na educação básica no Brasil”. In Anais dos

- Workshops do Congresso Brasileiro de Informática na Educação (Vol. 4, No. 1, p. 1464).
- Geraldes, W. B. (2017) “O Pensamento Computacional no ensino profissional e tecnológico”. 81 f. Dissertação (Mestrado em Gestão do Conhecimento e Tecnologia da Informação) – Pós-Graduação Stricto Sensu em Gestão do Conhecimento e Tecnologia da Informação – Universidade Católica de Brasília, Brasília, 2017.
- Lifelong Kindergarten Group (2007) “Scratch: Imagine, Program, Share”. MIT MediaLab, 2007. Disponível em: <http://Scratch.mit.edu/>. Acesso em 01 maio 2017.
- Primo, Alex. (2007) “O aspecto relacional das interações na Web 2.0”. E- Compós (Brasília), v. 9, p. 1-21, 2007.
- Reis, L. V. P.; Romão, A. U.; Leite, J. C. (2011) “Tutoriais: Um Modo Tipicamente Contemporâneo de Troca de Conhecimentos”. In XIII Congresso de Ciências da Comunicação na Região Centro-Oeste – Cuiabá – MT, 8 a 10 de junho de 2011, Cuiabá, Mato Grosso, 2011.
- Rodrigues, M. B. “Tecnologia deve ser ensinada desde a infância”: superintendente Centro-Sul American Chamber (Amcham) (Brasil). Disponível em: <<https://gauchazh.clicrbs.com.br/opiniao/noticia/2018/06/tecnologia-deve-ser-ensinada-desde-a-infancia-cji3pt6j50e8a01qoj83jm3ti.html>>. Acesso em: 14 jun. 2018.
- Rodrigues, R. da S. et al. (2015) “Análise dos efeitos do Pensamento Computacional nas habilidades de estudantes no ensino básico: um estudo sob a perspectiva da programação de computadores”. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). p. 121.
- Santiago, M. S. (2010) “Aspectos linguísticos de fraseologias em tutoriais da educação a distância”. In III Coloquio do PPG-Letras/UFRGS. Cadernos do IL. Porto Alegre, n.o 40, junho de 2010. p. 95-105. Disponível em <<http://www.seer.ufrgs.br/cadernosdoil/>>. Acesso em: 18 jan. 2016.
- Silva, M. (2010) “Sala de aula interativa: educação, comunicação, mídia clássica...” 5ª edição, São Paulo: Loyola.
- Sousa, R. M.; Lencastre, J. A. (2014) “Scratch: uma opção válida para desenvolver o Pensamento Computacional e a competência de resolução de problemas”. 2º Encontro sobre Jogos e Mobile Learning, p. 256-267.
- Steve Jobs: The Lost Interview. Direção de Paul Sen. Produção de Robert X. Cringely (2012). Legendado. Disponível em: <<https://www.netflix.com/br/title/70243590>>. Acesso em: 14 jun. 2018
- Valente, J. A. (1993) “Diferentes usos do Computador na Educação”. Disponível em:<<http://www.mrherondomingues.seed.pr.gov.br/redeescola/escolas/27/1470/14/arquivos/File/PPP/Diferentesusosdocomputadoreducacao.PDF>>. Acesso em: 13 out. 2015.

Valente, J. A. (2005) “A espiral da espiral de aprendizagem: o processo de compreensão do papel das tecnologias de informação e comunicação na educação”. 238 p. Tese (livre-docência) - Universidade Estadual de Campinas, Instituto de Artes, Campinas, SP. Disponível em: <<http://www.bibliotecadigital.unicamp.br/document/?code=000857072>>. Acesso em: 1 abr. 2017.

The CSTA Standards Task Force, “CSTA K-12 Computer Science Standards,” ACM Computer Science Teachers Association, New York, 2011.

Sobre os autores

Gilson Pedroso dos Santos



Mestre em Educação com ênfase em Práticas Educativas, Linguagens e Tecnologias, pela Universidade Federal do Oeste do Pará. Possui especialização em Gestão Pública pela Faculdade Venda Nova do Imigrante (FAVENI), Bacharel em Sistemas de Informação pela Universidade Federal do Oeste do Pará (UFOPA) e licenciado em Letras pela Universidade Luterana do Brasil (ULBRA). Atualmente pesquisa sobre Pensamento Computacional e TIC no ensino da matemática. Integra o Grupo de Estudos e Pesquisas em Educação Matemática e Interdisciplinaridade na Amazônia (GEPEIMAZ). Tem experiência com a construção de objetos de aprendizagem e na utilização de tecnologias para auxiliar no processo de ensino-aprendizagem.

Ronilson dos Santos Bezerra



Cientista da Computação pela Universidade Federal do Oeste do Pará (Ufopa) e pós-graduação em Informática da Educação e Docência do Ensino Superior pelo Instituto de Ciência, Educação e Tecnologia de Votuporanga - Faculdade Futura. Servidor Público Federal, atuando desde 2012 na Coordenação Acadêmica do Instituto de Ciências da Sociedade – Ufopa. Possui interesse principalmente em áreas de Informática na Educação, Pensamento Computacional, Inclusão Digital, Modelagem de dados, Programação Web e Robótica.