

Ensinando Design De Interface De Usuário Na Educação Básica: Um Mapeamento Sistemático Do Estado Da Arte E Prática

Miriam Nathalie F. Ferreira¹, Fernando Pinheiro², Raul Missfeldt Filho²,
Christiane Gresse von Wangenheim²

¹ Departamento de Expressão Gráfica – Universidade Federal de Santa Catarina

² Departamento de Informática e Estatística – Universidade Federal de Santa Catarina

{nathalie.fortuna, fernando.pinheiro}@posgrad.ufsc.br,
raul.missfeldt.filho@grad.ufsc.br, c.wangenheim@ufsc.br

Abstract. *Teaching computing in K-12 is typically realized through practical activities in which students create software artifacts. This, however, may not cover other important competencies, such as user interface (UI) design. Thus, in order to get an overview on the state of the art and practice of teaching UI design competencies in K-12, we conducted a systematic mapping study. Sixteen instructional units were identified, approaching diverse concepts. However, we observed a lack of detail as well as the availability of the instructional units. Even so, applications provide first indications that the teaching of UI design competences can be beneficial in K-12.*

Resumo. *O ensino de computação na Educação Básica é tipicamente realizado por meio de atividades práticas nas quais os alunos criam artefatos de software. Isso pode não abranger outras competências importantes, como o design de interface do usuário (UI). Assim, a fim de obter uma visão geral do estado da arte e da prática do ensino de competências de design de UI na Educação Básica, realizamos um mapeamento sistemático. Foram encontradas 16 unidades instrucionais, abordando diversos conceitos. No entanto, identificamos uma carência de detalhes e acessibilidade a essas unidades. Mesmo assim, aplicações fornecem primeiras indicações de que o ensino de design de UI pode ser benéfico no contexto da Educação Básica.*

1. Introdução

Numa sociedade em constante mudança do século XXI, há uma demanda para preparar os alunos com competências que vão além do conhecimento cognitivo (P21, 2017). Devido a isso, é necessário ensinar essas habilidades desde a Educação Básica (CSTA, 2011; P21, 2017). Atualmente, a introdução do ensino de computação nas escolas é uma tendência mundial apoiada por diversas iniciativas, voltadas, principalmente, para o ensino da programação por meio de ambientes de programação baseados em blocos.

Porém, essas iniciativas muitas vezes não abrangem outras competências importantes, como o design de interface de usuário, que é essencial para o desenvolvimento de software e faz parte do corpo de conhecimento da área de computação (ACM, 2013). As competências relacionadas ao design de interface do usuário são essenciais não apenas para a formação dos profissionais de TI, mas para

todos (AIGA, 2013). Geralmente, essas competências são ensinadas somente no nível do ensino superior (ACM, 2013). Entretanto, guias curriculares, como o *K-12 Computer Science Framework* (CSTA, 2017), indicam a sua importância também na Educação Básica, incluindo a Interação Humano-Computador como área transversal. Alguns países também já começaram a incorporar o ensino de design na Educação Básica, como, por exemplo, a China e a Coreia do Sul (West-Knights, 2017; Ahn, 2012).

O design, como área de conhecimento na Educação Básica, é capaz de promover vários benefícios, desde a aprendizagem de métodos de pesquisa, habilidades de visualizar e de apresentar informações, até o incentivo à análise crítica, colaboração e formação de equipes. Soma-se a isso a vantagem de facilitar o aprendizado de habilidades cognitivas criativas e de habilidades manuais produtivas (AIGA, 2013). O ensino do design pode atuar, então, como incentivo aos alunos, pois os torna mais criativos e imaginativos e também os ensina a aproveitar e a aplicar essa inventividade, além de colocá-la em prática (AIGA, 2013). Dessa forma, o aprendizado de competências do design propicia soluções e maneiras de se envolver com o mundo, as quais permitem que os indivíduos atuem como agentes de mudança e de criação do século XXI (Christensen et al., 2016).

Assim, surge a questão: se o design de UI é abordado atualmente no ensino de computação na Educação Básica e, se é, de que modo é feito. Vários artigos apresentam revisões gerais do ensino da computação nesse nível de educação. Grover e Pea (2013) apresentam uma revisão sistemática do ensino do pensamento computacional na Educação Básica. Outros resumem como a ciência da computação é tratada em diversos países na Educação Básica (Hubwieser et al., 2015; Heintz et al., 2016) ou fornecem visões gerais sobre a adoção de ambientes de programação (Moreno-León e Robles, 2016) ou sobre os tipos de conteúdo (robótica) (Bascou e Menekse, 2016). No entanto, nenhuma dessas revisões direcionadas à Educação Básica aborda especificamente a questão do ensino de design de UI.

Por outro lado, existem revisões do estado da arte do ensino de Interação Humano-Computador (Churchill, Bowser e Preece, 2013) no contexto de design de UI (Chiam e Ghazali, 2012) ou de países específicos (Boscarioli et al., 2016). Entretanto, elas enfocam na Educação Superior. Outras visam rever o estado da arte do ensino de competências relacionadas (como criatividade), porém, no contexto de outras áreas, como artes (Sawyer, 2017), e não na computação. Dessa forma, sabe-se pouco sobre o modo efetivo de ensinar essas competências, especialmente dentro de restrições típicas dentro do currículo escolar. Portanto, é necessário explorar quais e como as competências referentes ao design de interface do usuário podem ser integradas ao ensino de computação na Educação Básica.

A fim de obter uma visão geral do estado da arte e da prática do ensino de competências de design de UI na Educação Básica realizamos um mapeamento sistemático. Na seção 2 apresentamos a definição e execução do mapeamento sistemático e na seção 3 apresentamos a análise dos dados e discussão. Na seção 4 é apresentada a conclusão.

2. Definição e Execução do Mapeamento Sistemático

Para levantar o estado da arte e a prática sobre *se e como* o ensino de design de interface de usuário é abordado na Educação Básica, realizamos um estudo de mapeamento sistemático seguindo o procedimento proposto por Petersen et al. (2008).

2.1 Definição do Mapeamento Sistemático

Questão de Pesquisa. Existem (e, quais as suas características) unidades instrucionais¹ que ensinam competências de design de UI no contexto de ensino de computação na Educação Básica? Essa questão de pesquisa é refinada nas seguintes perguntas de análise:

PA1. Quais unidades instrucionais existem?

PA2. Qual(is) competências de design de interface de usuário são ensinados?

PA3. Quais as características instrucionais das unidades instrucionais?

PA4. Quais as características de contexto das unidades instrucionais?

PA5. Como as unidades instrucionais foram desenvolvidas?

PA6. Como a qualidade das unidades instrucionais é avaliada?

Critérios de inclusão/exclusão. Consideramos apenas artigos revisados por pares cujo foco é ensinar computação adotando práticas ou atividades de design de interface de usuário na Educação Básica voltadas ao desenvolvimento de *software*. São excluídos artigos que focam em robótica, em ensino superior ou em artigos que apresentam unidades instrucionais sem abordar conceitos de design de UI. Incluímos, também, literatura secundária descoberta por meio das referências da literatura primária.

Critério de Qualidade. Consideramos apenas artigos que apresentam informações substanciais em relação ao ensino de conceitos de design de UI, indicando, por exemplo, conteúdo das aulas, materiais didáticos, etc.

Fonte de dados. A busca foi feita no Scopus, Science Direct, Google Scholar, ERIC, representando as principais editoras científicas, e, também, em sites de MOOCs, incluindo *Udemy*, *Edx*, *Khanacademy*, *Coursera*. Com o intuito de abranger uma maior gama de publicações, foram realizadas buscas no Google Scholar e no Google (Haddaway et al., 2015).

Definição da *string* de busca. A *string* de busca é composta de conceitos relacionados à questão de pesquisa, considerando também sinônimos (Tabela 1).

Tabela 1: Strings de busca

Fonte de Dados	String de busca
Scopus	("user interface design" OR "graphic design" OR "visual design" OR "user experience" OR UX OR "design thinking") AND (computing OR coding OR programming) AND ("K-12" OR school OR kids OR children)
Science Direct	("user interface design" OR "graphic design" OR "visual design" OR "user experience" OR UX OR "design thinking") AND (computing OR coding OR programming) AND ("K-12" OR school OR kids OR children)
ERIC	("user interface design" OR "graphic design" OR "visual design" OR "user experience" OR "UX" OR "design thinking") AND ("coding" OR "programming") AND ("K-12" OR "school" OR "kids" OR

¹ Unidade Instrucional é um conjunto de aulas planejadas para ensinar determinados objetivos de aprendizagem a um público-alvo específico. Ela é constituída por um conjunto de materiais instrucionais, tanto para professores quanto para alunos, desenvolvidos para oportunizar o aprendizado em um contexto específico. (Hill; Rowan; Ball, 2005).

	"children")
Google scholar	("design thinking" OR "user interface design") AND ("programming" OR "coding" OR "apps") AND ("K-12" OR "school" OR "kids" OR "children")
Google	"design thinking" (computing OR coding OR programming) (kids OR K-12 OR school)
	"user experience" design (computing OR coding OR programming) (kids OR K-12 OR school)
	"user interface design" (computing OR coding OR programming) (kids OR K-12 OR school)

2.2 Execução da busca

A busca foi realizada em maio de 2018 em duas etapas. Na primeira, a busca foi feita via Scopus, Science Direct, ERIC, Google Scholar e Google (Tabela 2). Em relação às buscas realizadas no Google e no Google Scholar analisamos somente resultados mais relevantes. A partir desses, selecionamos 89 artigos potencialmente relevantes, analisando o título, o resumo e as palavras-chave. Na segunda etapa, analisamos o texto completo dos artigos pré-selecionados, identificando 16 artigos relevantes.

Tabela 2: Quantidade de artigos por etapa de seleção por repositório.

Fonte de dados	Resultado da pesquisa inicial	Resultados analisados	Resultado da seleção depois do 1º estágio	Resultado da seleção depois do 2º estágio
Scopus	160	160	13	4
Science Direct	83	83	1	0
ERIC	57	57	3	0
Google Scholar	17.800	300	22	7
Google	Design Thinking: 1.160.000 UX: 8.770.000 Interface design: 7.150.000	Design Thinking: 200 UX: 150 Interface design: 200	Design Thinking: 25 UX: 6 Interface design: 19	5
Total				16

Numa segunda etapa da busca, também foram procurados cursos de ensino *on-line* adotando os mesmos critérios de inclusão/exclusão. Como resultado desta pesquisa adicional, foi encontrada uma unidade instrucional (Collofello, 2002).

3. Análise de dados

Para responder à questão de pesquisa, extraímos informações relevantes às perguntas de análise. As informações que foram extraídas dos artigos são detalhadas em Ferreira et al. (2018).

3.1. Quais unidades instrucionais existem?

Como resultado da pesquisa, identificamos 16 unidades instrucionais voltadas ao ensino de computação, que, de alguma forma, abordam, também, o ensino de design de interface do usuário no nível da Educação Básica (Tabela 3).

Tabela 3: Artigos relevantes

Referência	Título
(Chen e Huang, 2017)	Design thinking in App inventor game design and development: A case study
(Sullivan et al., 2003)	Girls embrace technology: A summer internship for high school girls
(Ke e Im, 2014)	A case study on collective cognition and operation in team-based computer game design by middle-school children
(Robinson e Pérez-Quñones, 2014)	Underrepresented middle school girls: on the path to computer science through paper prototyping
(Van Wart et al., 2014)	Apps for social justice: Motivating computer science learning with design and real-world problem solving
(Denner et al., 2005)	The girls creating games program: Strategies for engaging middle-school girls in information technology

(EDUTOPIA, 2015)	Coding by design first approach
(CODELIKEAGIRL, 2017)	Learning design by making games in scratch
(TEKKIE UNI, 2018)	Build Your First App
(CREATELAB, 2017)	My createlab
(CODE, 2018)	Curriculum code
(Robinson et al., 2015)	Understanding the attitudes of African American middle school girls toward computer Science
(TECHNOVATION, 2018)	Technovation Challenge
(CODEHS, 2018)	Codehs
(CODE.ORG/APP LAB, 2018)	Applab
(GET STARTED WITH CODE 2, 2017)	Get started with code 2

3.2. Qual(is) competências de design de interface são ensinadas?

Observamos que são abordadas competências de design de interface, incluindo *design thinking*, design de interface do usuário, design visual, design centrado no usuário, usabilidade, interação humano-computador e UX. Como tendência atual, nessa área também se adota o *design thinking* como uma abordagem inovadora para design centrado no usuário com o intuito de explorar outras possibilidades, criar novas escolhas e trazer soluções inovadoras aos problemas (Brown, 2011). Algumas unidades instrucionais, além de visar a aprendizagem da lembrança e a compreensão desses conceitos, abordam, ainda, a aprendizagem de diversas técnicas, permitindo ao aluno entender, explorar e materializar, buscando ideias sobre a melhor forma de resolver um problema (Figura 1).

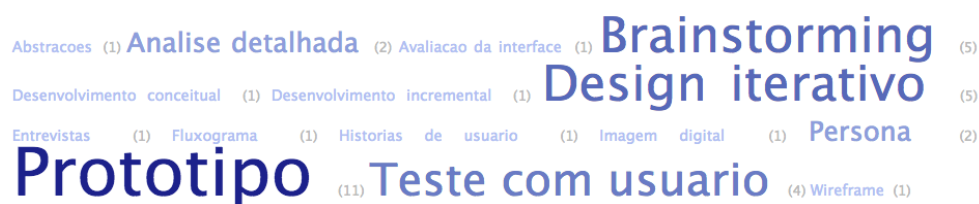


Figura 1: Frequência das técnicas utilizadas para ensinar design de interface

As unidades instrucionais fazem uso de ferramentas específicas para o design da interface, como Adobe Photoshop Macromedia Director, Pages e Mini Monet. Para o design de interface e a produção de mapas de navegação, algumas unidades instrucionais utilizaram Lingo, POP app, Invision app, editor *on-line* do CodeHS e Balsamiq.

3.3. Quais são as características instrucionais das unidades instrucionais?

No contexto do desenvolvimento de jogos, animações ou aplicativos, as unidades instrucionais ensinam competências do processo de *design thinking* e práticas de design de interface visual. Para o desenvolvimento desses *softwares* geralmente são adotados ambientes de programação visuais baseados em blocos. Observou-se também a adoção de linguagens de programação baseadas em textos, como HTML e CSS, que definem a parte visual de um website. Outras ferramentas foram utilizadas para a criação de animações e/ou jogos, como o Adobe Flash (Figura 2).

Em termos de métodos instrucionais, observou-se uma predominância de abordagens voltadas à aprendizagem ativa no nível de aplicação (Bloom, 1973). O ensino por meio de resolução de problemas foi utilizado como parte do ensino de *design thinking* com o objetivo de incentivar o pensamento criativo dos alunos. Diversas unidades instrucionais também incluem métodos de instrução direta, como aulas expositivas, vídeo

aula, demonstrações, principalmente na parte inicial da unidade instrucional. Usa-se também métodos indiretos como a investigação e os exercícios em sala de aula. Entre os métodos interativos, os mais utilizados foram *brainstorming*, desafios, entrevista, trabalho em grupo e discussões. Métodos de aprendizagem experimental também foram aplicados por meio de simulações, interpretação de papéis e condução de experimentos.

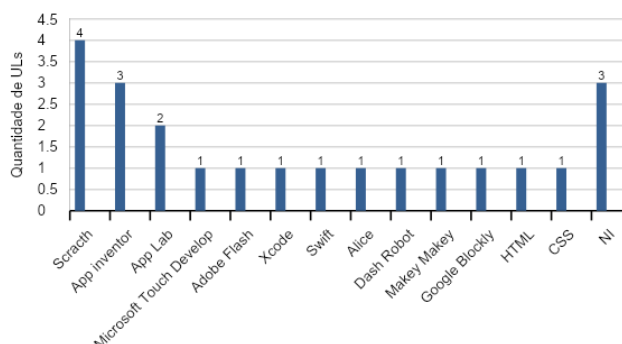


Figura 2: Ambientes de programação utilizados nas unidades instrucionais

Em virtude da variação de métodos instrucionais, são utilizados diferentes tipos de materiais. Os materiais mais utilizados para construir protótipo de papel ou *wireframes* são papéis, lápis, marcadores, etc. (Figura 7). Também são utilizados *post-its* para definir fluxogramas, registrar as ideias e requisitos de *software*. Artefatos do processo de *software* são utilizados para auxiliar no seu desenvolvimento, como, por exemplo, amostras de código e *user stories*. Além disso, foram utilizados slides e vídeos. No entanto, de forma geral, percebeu-se a apresentação de poucas informações em relação aos materiais nas publicações, suas disponibilidades e licenças de uso, o que dificulta o uso por outros interessados. A maioria dos materiais também está disponível em somente uma única língua (Inglês), o que pode limitar uma adoção mais ampla em outros países, já que necessitam tipicamente do material na língua nativa nesse nível escolar.

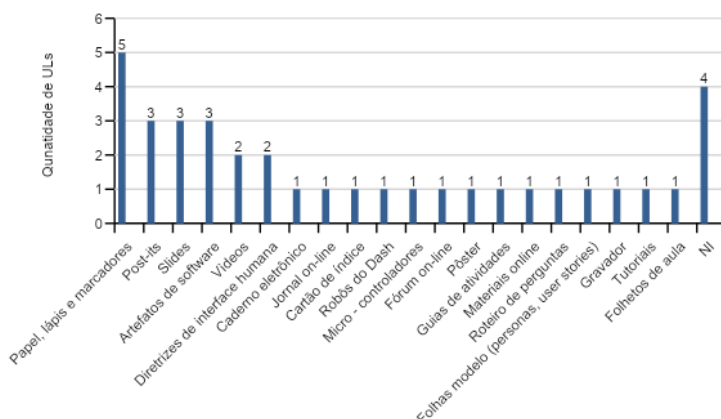


Figura 3: Materiais instrucionais utilizados

A aprendizagem dos alunos é avaliada principalmente por meio de avaliações baseadas no desempenho, analisando artefatos criados no contexto do ensino de design de interface. Essas avaliações contemplam a interação da interface de um *software* funcional, como jogos, websites ou apps. Observou-se, que também são aplicados exercícios com o intuito de fazer o aluno empregar os conceitos ensinados, como, por

exemplo, avaliar o design de objetos, projetar um produto por meio de protótipos em papel, realizar testes de usabilidade.

3.4 Quais são as características de contexto da unidade instrucional?

A maioria das unidades instrucionais encontradas não requisita conhecimento prévio e pode ser ensinada em qualquer nível escolar. Porém, observamos um maior número de unidades voltadas aos anos finais do Ensino Fundamental e do Ensino Médio. No que diz respeito ao período das unidades instrucionais, não existe um padrão, pois encontramos tanto oficinas de poucas horas quanto cursos de longa duração de 1 ano. Porém, a maioria das unidades instrucionais são de curta duração, variando de 4 a 18 h/a.

3.5. Como as unidades instrucionais foram desenvolvidas?

Percebe-se que os registros geralmente restringem-se na indicação das partes envolvidas por meio de cooperações entre escolas e/ou universidades, envolvendo professores, instrutores e tutores (Tabela 4). Dez unidades instrucionais não apresentam nenhuma informação em relação à forma como foram desenvolvidas.

Tabela 4: Métodos de desenvolvimento das unidades instrucionais

Referência	Método de desenvolvimento
(Sullivan et al., 2003)	Mentores de nível superior (não graduados) foram utilizados para auxiliar e controlar as equipes. Os materiais e o plano de ensino foram desenvolvidos de acordo com os objetivos de aprendizagem que também foram apresentados aos alunos e pais.
(Ke e Im, 2014)	Um grupo de estudantes graduados facilitou todas as sessões de design. Eles responderam perguntas, deram <i>feedback</i> entre as etapas do design e, ocasionalmente, levaram as crianças a participarem de explicações sobre o design ou o conteúdo matemático. Eles também forneceram ajuda na programação do Scratch durante o desenvolvimento do jogo.
(Denner et al., 2005)	O desenvolvimento se baseou em métodos que incorporam uma abordagem de pesquisa focada em design, no qual foi criada e testada atividades programáticas e estratégias instrucionais, depois foram refinadas para aumentar a eficácia. Essa abordagem fornece dados sobre contextos e mecanismos para descrever relações causais plausíveis entre estratégias e objetivos do programa. Foram treinadas instrutoras que trabalham como líderes de programa ou assistentes de professores na escola, instrutores de programação.
(CREATELAB, 2017)	A unidade Instrucional foi desenvolvida por uma cooperação de universidades alinhada as Etapas Chave 1-3 dos Programas Britânicos de Estudo de Computação.
(Robinson et al., 2015)	Foi utilizado a Teoria Cognitiva Social de Bandura
(TECHNOVATION, 2018)	Foram envolvidos mentores para auxiliar os grupos em todas as partes do desenvolvimento.

3.6. Como a qualidade da unidade instrucional é avaliada?

A avaliação é tipicamente realizada por meio de estudos empíricos, principalmente de estudos de caso. Apenas um estudo adotou um design de pesquisa mais rigoroso (Sullivan et al., 2003). Duas unidades instrucionais foram avaliadas de forma *ad-hoc*. Como resultado, esses estudos comentam somente o *feedback* informal dos alunos e/ou as observações feitas. As avaliações focam em demonstrar que os alunos conseguem aprender conceitos relacionados ao design de UI além de outros fatores, como utilidade, diversão, motivação, satisfação, etc. Essa diversidade de fatores de qualidade reflete a falta de um modelo de avaliação para esse tipo de unidade instrucional. Os dados são coletados mais por entrevista, questionários e observação. Dados via vídeos e *log files* de fóruns foram coletados em cursos *on-line* para analisar o engajamento do aluno no curso.

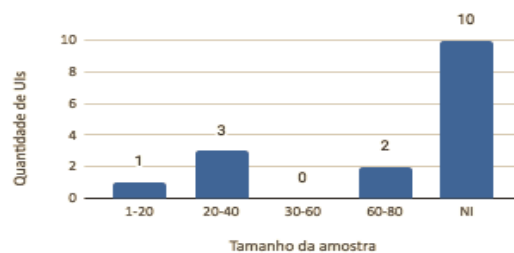


Figura 4: Quantidade de estudos por tamanho da amostra.

É importante salientar que a maioria das avaliações foi realizada com amostras pequenas, variando de 1 a 60 participantes. Esse baixo número de envolvidos corresponde ao tamanho típico de uma turma na qual a unidade instrucional é aplicada. Somente dois estudos utilizaram testes estatísticos para a análise dos dados coletados (Robinson, Pérez-Quinones e Scales, 2015; Ke e Im, 2014). Outros estudos, devido à adoção de designs de pesquisa menos rigorosos, não detalharam a análise de dados.

Ameaças à Validade. Existem várias ameaças à validade dos resultados dessa pesquisa. A fim de mitigar o risco da omissão de estudos relevantes, construímos a *string* de busca para ser o mais abrangente possível, considerando também, sinônimos. Para minimizar o risco de excluir unidades instrucionais existentes, as quais ainda não foram relatadas por meio de artigos científicos, realizamos uma busca de forma nos sites de MOOCs, Google scholar e Google e incluímos literatura secundária. Ameaças à seleção e à extração de dados foram mitigadas pela definição detalhada dos critérios. Definimos um protocolo rígido para a seleção do estudo. A extração de dados foi prejudicada em alguns casos, uma vez que as informações relevantes nem sempre foram apresentadas explicitamente e, por isso, tiveram que ser inferidas.

4. Conclusão

Considerando a relevância do design de UI no desenvolvimento de *software*, observa-se que há uma carência de ensino dessas competências na Educação Básica. Encontramos somente dezesseis unidades instrucionais que contemplam o design de UI. As unidades encontradas focam no ensino de *design thinking*, design visual, usabilidade, design de UI, interação humano computador, design centrado no usuário e UX. Porém, observou-se a ausência de conceitos mais concretos, como cores, tipografia, ícones, etc. Evidenciou-se também uma falta de informações sobre o acesso aos seus materiais e de modelos de avaliação da aprendizagem. A falta de relatos acerca de como as unidades instrucionais foram desenvolvidas e avaliadas também aponta a necessidade de pesquisas com maior rigor científico. Mesmo assim, as aplicações relatadas fornecem primeiras indicações de que o ensino de competências de design de UI pode ser benéfico tanto no ensino fundamental quanto no ensino médio como parte do ensino de computação.

Agradecimentos

Este trabalho foi apoiado pelo CNPq, uma entidade do governo brasileiro voltada para o desenvolvimento científico e tecnológico.

Referências

ACM/IEEE. (2013) Computer Science Curricula 2013: *Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. Nova York, EUA.

- AIGA. (2013). *The Professional Association for Design*. Disponível em: <<https://www.aiga.org/design-ed-k12>>. Acesso em: ago. 2019.
- Bascou, N. A.; Menekse, M. (2016) Robotics in K-12 formal and informal learning environments: A review of literature. In: *Proc. of the ASEE Annual Conf. e Exposition*. New Orleans, EUA.
- Boscarioli, C., et al. (2016). IHC Education in Brazil. In: *Proc. of the 15th Brazilian Symposium on Human Factors in Computing Systems*. ACM, New York, NY, USA.
- Bloom, B. S. et al. (1973). Taxonomia de objetivos educacionais: domínio afetivo. *Porto Alegre: Globo*.
- Brown, T., & Katz, B. (2011). Change by design. *Journal of Product Innovation Management*, 28(3), 381-383.
- Chen, P., Huang, R. (2017). Design thinking in App inventor game design and development: A case study. In: *Proc. of 17th Int. Conf. on Advanced Learning Technologies*. Timisoara, Romania.
- Chiam, N., Ghazali, M. A. (2012). Systematic Literature Review on the Role of Usability in Software Engineering Education. In *Proc. of Int. Conf. on Information and Knowledge Management*. Kuala Lumpur, Malaysia.
- Christensen, K. S. et al. (2016). Towards a formal assessment of design literacy: Analyzing K-12 students' stance towards inquiry. *Design Studies*, 46, 125-151.
- Churchill, E. F., Bowser, A., Preece, J. (2013). Teaching and learning human-computer interaction: past, present, and future. *Interactions*, 20(2), 44-53.
- CODE. (2018). *Code.org*. Disponível em: <<http://code.org>>. Acesso em: ago. 2019.
- CODE.ORG/APP LAB. (2018). Disponível em: <<https://code.org/educate/applab>>. Acesso em: ago. 2019.
- CodeClubWorld. (2012).). Disponível em: <<https://codeclubword.org>>. Acesso em: ago. 2019.
- CODEHS. (2018). Disponível em: <<https://codehs.com/info/curriculum>>. Acesso em: ago. 2019.
- Code like a Girl. (2017). Disponível em: <<https://code.likeagirl.io/>>. Acesso em: ago. 2019.
- Collofello, J. S. (2002) Creation, deployment and evaluation of an innovative secondary school Software development curriculum module. In *Proc. of the 32nd Annual Frontiers in Education*, Boston, EUA.
- CPUXF. (2018). *Curriculum and Glossary*. Version 3.15 EN, 23 March 2018.
- CREATELAB (2017). Disponível em: <<http://www.mycreatelab.com/ais/>>. Acesso em: ago. 2019.
- CSTA. (2017). *K-12 Computer Science Framework*. Disponível em: <<https://www.doe.k12.de.us/cms/lib/DE01922744/Centricity/Domain/176/CSTA%20Computer%20Science%20Standards%20Revised%202017.pdf>>.
- Denner, J., et al. (2005). The girls creating games program: Strategies for engaging middle-school girls in information technology. *Frontiers: A Journal of Women Studies*, 26(1), 90-98.
- Edutopia (2015). *Edutopia*. Disponível em: <<https://www.edutopia.org/blog/coding-by-design-first-approach-douglas-kiang>>. Acesso em: ago. 2019.
- Ferreira, M. N. F., et al. (2018). Material suplementar - Ensinando Design de Interface de Usuário na Educação Básica. Relatório Técnico 09.2018, INCoD/INE/UFSC, Florianópolis, Brasil.
- Francis, K., et al. (2017). Multidisciplinary perspectives on a video case of children designing and coding for robotics. *Canadian Journal of Science, Mathematics and Technology Education*, 17(3), 165-178.
- Garrett, J. J. (2011). *The Elements of User Experience: User-Centered Design for the Web and Beyond*, (2nd ed.). Berkeley: New Riders.

- Get started with code 2. (2017). Disponível em: <<https://itunes.apple.com/us/book/get-started-with-code-2/id1226776857>>. Acesso em: ago. 2019.
- Grover, S., e Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, 42(1), 38-43.
- Haddaway, N. R. et al. (2015). The role of Google Scholar in evidence reviews and its applicability to grey literature searching. *PloS one*, 10(9).
- Heintz, F., et al. (2016) A review of models for introducing computational thinking, computer science and computing in K-12 education. In: *Proc. of the Frontiers in Education Conference*, Erie, EUA.
- H. Hill, B. Rowan, D. Ball, Effects of Teachers' Mathematical Knowledge for Teaching on Student Achievement. *American Educational Research Journal*, 42(2), 2005, pp. 371-406.
- Hubwieser, P. et al. (2015) A global snapshot of computer science education in K-12 schools. In: *Proc. of the ITiCSE on Working Group Reports*. Nova York, EUA.
- Iniciativa Computação na Escola. (2017). Disponível em: <<http://www.computacaonaescola.ufsc.br>>. Acesso em: ago. 2019.
- ISO 9241-210: 2010. (2009). Ergonomics of human system interaction-Part, 210.
- Ke, F., e Im, T. (2014). A case study on collective cognition and operation in team-based computer game design by middle-school children. *International Journal of Technology and Design Education*, 24(2), 187-201.
- Moreno-León, J.; Robles, G. (2016) Code to learn with Scratch? A systematic literature review. In *Proc. of the Global Engineering Education Conf.*, Abu Dhabi, Emirados Árabes.
- Petersen, K. et al. (2008) Systematic mapping studies in Software engineering. In: *Proc. of the 12th Int. Conf. on Evaluation and Assessment in Software Engineering*, Swindon, Inglaterra.
- Robinson, A., e Pérez-Quinones, M. A. (2014). Underrepresented middle school girls: on the path to computer science through paper prototyping. In: *Proc. of the 45th ACM Technical Symposium on Computer Science Education*, Atlanta, EUA.
- Robinson, A. et al. (2015). Understanding the attitudes of African American middle school girls toward computer science. In: *Proc. of Research in Equity and Sustained Participation in Engineering, Computing, and Technology*, Charlotte, EUA.
- Sawyer, R. K. (2017). Teaching creativity in art and design studio classes: A systematic literature review. *Educational research review*, 22, 99-113.
- Sullivan, J. F. et al. (2003). Girls embrace technology: A summer internship for high school girls. In *Frontiers in Education*, Westminster, EUA.
- Technovation. (2018). *Technovation Challenge*. Disponível em: <<http://www.technovationbrasil.org/curriculo>>.
- Tekkie Uni (2018). *Build Your First App*. Disponível em: <<https://tekkieuni.com/courses/build-your-first-app/>>. Acesso em: ago. 2019.
- Van Wart, S. J., Vakil, S., e Parikh, T. S. (2014). Apps for social justice: Motivating computer science learning with design and real-world problem solving. In: *Proc. of the Conf. on Innovation e Technology in Computer Science Education*, Uppsala, Sweden.
- Verhoeff, T. (2006) A master class Software engineering for secondary education. In: *Proc. of the Int. Conf. on Informatics in Secondary Schools*. Heidelberg, Alemanha.
- West-Knights, I. (2017). Why are schools in China looking west for lessons in creativity. *Financial Times*.