

# Uma Revisão Sistematizada sobre Design Participativo Integrado à Engenharia de Software em Domínio Educacional

Amanda Meincke Melo<sup>1</sup>, Jean Felipe Patikowski Cheiran<sup>1</sup>, Auri Gabriel Castro de Melo<sup>1</sup>, Gabriel Souza Rodrigues de Amorim<sup>1</sup>

<sup>1</sup> Universidade Federal do Pampa (Unipampa) – Campus Alegrete  
Av. Tiarajú, 810 – Ibirapuitã – 97.546-550 – Alegrete – RS – Brasil

{amandamelo, jeancheiran}@unipampa.edu.br

{aurimelo.aluno, gabrielamorim.aluno}@unipampa.edu.br

**Abstract.** *Pieces of software support a variety of tasks in the educational domain. They must meet observable quality requirements and promote good experiences for their users. Participatory Design (PD), integrated with Software Engineering (SE), has the potential to address the experiences and needs of end users in software development focusing on quality. The pivotal question is: How do PD and SE integrate in software development for this domain? When investigating this issue by conducting a systematized review, it was found that there is room for proposing a framework that guides Software Engineers in adopting PD in software development for the educational domain.*

**Resumo.** *Software apoiam uma variedade de tarefas no domínio educacional. Devem atender a requisitos de qualidade, assim como promover boas experiências aos seus usuários. O Design Participativo (DP), integrado à Engenharia de Software (ES), tem o potencial de contemplar as experiências e as necessidades dos usuários finais no desenvolvimento de software com foco na qualidade. Pergunta-se: Como está a integração do DP à ES no desenvolvimento de software para esse domínio? Ao investigar essa questão, a partir de uma revisão sistematizada, constatou-se haver espaço para a proposição de um framework que oriente profissionais de Engenharia de Software na adoção do DP no desenvolvimento de software para o domínio educacional.*

## 1. Introdução

Software estão incorporados pervasivamente a diferentes domínios, inclusive ao educacional. Neste domínio, apoiam uma variedade de atividades relacionadas ao ensino, à aprendizagem, à gestão, entre outras. Professores e estudantes são recorrentemente mencionados entre os usuários de software nesse domínio, sobretudo relacionados aos processos de ensino e de aprendizagem [Rosa and Matos 2016, Falcão et al. 2017, Tuhkala 2021].

Um bom produto de software deve atender a requisitos de qualidade observáveis com apoio de métodos de inspeção e de testes [Pressman and Maxim 2016, Sommerville 2019]. Além disso, deve promover boas experiências aos seus usuários [Barbosa and Silva 2010, Rogers et al. 2011]. Essas experiências afetam diretamente os usuários e seu desempenho na realização de uma tarefa apoiada por sistema computacional interativo, assim como influenciam na aceitação do produto final. O Design Participativo (DP) [Muller et al. 1997], integrado à Engenharia de Software (ES)

[Pressman and Maxim 2016, Sommerville 2019], tem o potencial de contemplar as experiências e as necessidades dos usuários finais no desenvolvimento de software. Isso porque a abordagem do DP propõe métodos e técnicas para diferentes etapas do desenvolvimento de software em que essas experiências e necessidades podem ser colocadas em perspectiva, assim como para que os usuários interfiram nas características do produto de software em desenvolvimento e o experimentem [Muller et al. 1997].

A adoção do DP, assim como da ES, no desenvolvimento de software para o domínio educacional não é novidade. Entretanto, com o objetivo de mapear e caracterizar os esforços de aproximação dessas áreas, define-se a seguinte questão de pesquisa para esta revisão: Como está a integração do DP à ES no desenvolvimento de software para esse domínio? Além disso, é possível estabelecer uma questão de pesquisa secundária relevante para apoiar desenvolvedores: Há *frameworks* que orientam Engenheiros de Software na adoção do Design Participativo durante o desenvolvimento de software para o domínio educacional, considerando toda a sua complexidade? Propõe-se com este trabalho responder a essas questões a partir de uma revisão sistematizada realizada com a adoção da abordagem de busca *snowballing* [Jalali and Wohlin 2012].

## 2. Trabalhos Relacionados

A adoção do Design Participativo pela comunidade de Interação Humano-Computador (IHC) ocorreu na década de 1990 [Hartson and Pyla 2019], promovendo um crescente interesse da comunidade acadêmica no tema que se observa pelo uso dessa prática em diversos domínios [Halskov and Hansen 2015] e pela promoção de conferências especializadas – *Participatory Design Conferences*<sup>1</sup> (PDC). Com uma abundante coleção de estudos que descrevem o uso do DP e os resultados desse uso, diversos estudos secundários foram conduzidos sob a perspectiva de diferentes domínios de aplicação.

Revisões sobre uso de DP em domínios específicos estão entre as mais são numerosas, embora existam também revisões mais amplas. Halskov e Hansen, por exemplo, analisaram 10 anos de publicações nas PDC para identificar aspectos, domínios, métodos, contribuições e definições em estudos que envolvem DP [Halskov and Hansen 2015]. Os autores descreveram alguns exemplos de domínios de software (e.g., assistência de saúde e mídias sociais), contribuições teóricas (e.g., desenvolvimento guiado por efeitos e entrevista estética) e métodos (e.g., personas e clipe de experiência) associados ao DP. Posteriormente, Poderi e Dittrich analisaram também artigos publicados nas PDC até o ano de 2016 com objetivo de mapear como a sustentabilidade é abordada em DP nas perspectivas de DP para sustentabilidade, sustentabilidade das práticas de DP e sustentabilidade dos resultados de DP [Poderi and Dittrich 2018]. Embora esses dois artigos relatem a identificação de *frameworks* como uma contribuição para a área de DP, os artigos originais que descrevem esses *frameworks* não são citados nas revisões ou materiais de apoio.

Dentre os domínios cobertos por revisões sobre uso do DP estão a assistência de saúde [Hamzah and Wahid 2016], sistemas de informação de *big data* [Whitman et al. 2018], gerontologia [Sumner et al. 2020], jogos sérios [Maheu-Cadotte et al. 2021] e biblioteconomia [Nesset et al. 2024]. Tendo em vista que o domínio educacional é alvo desta revisão, é significativo que os estudos secundários mais relevantes nessa temática sejam apresentados com mais detalhes.

<sup>1</sup>Disponíveis em <http://pdcproceedings.org>.

O trabalho de Tuhkala e colegas cobriu 10 anos de estudos sobre DP em 14 bases de dados científicas com objetivo de identificar agrupamentos temáticos [Tuhkala et al. 2018], coletando 2939 artigos para análise semiautomática. Embora o artigo seja uma revisão geral sobre DP, os autores destacaram apenas agrupamentos relacionados ao domínio educacional que incluem *game design*, crianças, estudantes, DP na aprendizagem, ensino de DP, *design* educacional, e *design* de tecnologias assistivas.

Em uma nova revisão, Tuhkala analisou 2943 artigos que envolviam professores em DP obtidos a partir de 14 bases científicas [Tuhkala 2021]. O artigo mapeia estudos que aplicam DP junto a professores da educação infantil ao ensino superior, detalhando seu envolvimento em relação a ambiente, práticas e tecnologias. As poucas ocorrências de *frameworks* relatadas na pesquisa de Tuhkala incluem um *framework* voltado para o *design* instrucional, a visão do DP como um *framework* no contexto de desenvolvimento profissional, e o uso de um *framework* de análise de interessados fundamentado no DP para o *design* de um jogo educacional para pessoas com necessidades especiais.

Observa-se que, nas revisões analisadas, embora abordem o desenvolvimento de software para diferentes domínios, inclusive o educacional, a perspectiva de integração entre DP e ES não está sistematizada, ainda que sejam reportados aspectos relacionados à qualidade, assim como processos, métodos e ferramentas relacionados à DP e ES. Nessas revisões há uma clara preocupação em identificar contribuições do Design Participativo ao design de novas tecnologias sem, contudo, estruturá-las nas práticas da Engenharia de Software e vice-versa.

### 3. Método de Pesquisa

Este estudo é caracterizado como uma revisão sistematizada que adota um passo de *backward* e *forward snowballing* a partir de artigos semente para formar a coleção de estudos analisados. Estes são apresentados e discutidos na seção 3.1.

Dentre os diferentes tipos de revisões, a revisão sistematizada tem objetivo de incluir elementos do processo de revisão sistemática, podendo simplificar ou flexibilizar processos [Grant and Booth 2009]. O foco de revisões sistemáticas e revisões sistematizadas é similar, porém a simplificação de algumas etapas (como a avaliação de qualidade dos estudos encontrados) ou a adoção de técnicas alternativas (como buscas baseadas em referências em vez de buscas em bases de dados) representa uma fonte adicional de vies nos resultados [Grant and Booth 2009].

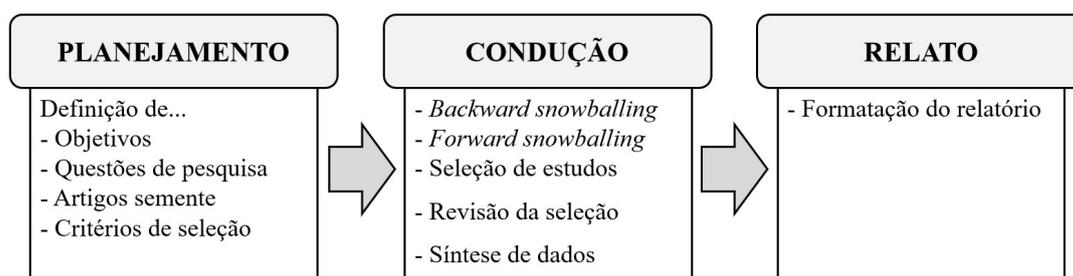
Estratégias de busca baseadas em referências costumam ser usadas para reduzir a quantidade de artigos pouco relevantes que comumente são retornados em buscas em bibliotecas digitais [Skoglund and Runeson 2009]. Dentre as abordagens de busca baseadas em referências, o *snowballing*<sup>2</sup> se destaca como uma prática originalmente complementar a outros métodos de busca que passou a ser adotada como estratégia principal no esforço de encontrar literatura relevante [Jalali and Wohlin 2012]. Enquanto o *backward snow-*

---

<sup>2</sup>O termo *snowballing*, que pode ser traduzido como ‘bola de neve’, representa uma analogia para uma pequena bola de neve que rola a partir de um lugar mais alto: conforme a bola desce, ela agrega mais flocos de neve e cresce de tamanho. Esse comportamento é similar a essa estratégia de busca, dado que um artigo (ou uma pequena coleção de artigos) é usado para aumentar o espaço de busca a partir de suas referências e citações e, conseqüentemente, aumentar a coleção de artigos descobertos que poderão ser usados da mesma forma para incrementar novamente o espaço de busca.

*balling* usa as referências de artigos escolhidos como base (chamados também de artigos semente) para aumentar a coleção de artigos sob análise, o *forward snowballing* usa as citações aos artigos escolhidos para ampliar a coleção [Jalali and Wohlin 2012].

Neste trabalho, as recomendações de Revisões Sistemáticas de Literatura (RSL) de Kitchenham [Kitchenham and Charters 2007] foram seguidas, embora tenham sido adotadas as seguintes adaptações: (1) a busca em bases de dados por meio de *strings* foi substituída por *backward snowballing* e *forward snowballing*, (2) a avaliação de qualidade dos artigos não foi realizada para ampliar a coleção de análise e (3) a extração dos dados dos artigos foi substituída pela elaboração de uma síntese direta dos trabalhos com informações relevantes para responder às questões de pesquisa. Os passos executados nessa revisão sistematizada são apresentados na Figura 1.



**Figura 1. Visão geral das etapas executadas na Revisão Sistemática.**

Considerando que intenciona-se a identificação de estudos que proponham *frameworks* para integração de DP e ES em domínio educacional, esta revisão foi guiada pelas seguintes questões de pesquisa que são respondidas na apresentação dos resultados:

1. Que softwares são desenvolvidos nesses estudos?
2. De que modo a qualidade é abordada nesses estudos?
3. Que processos, métodos e ferramentas de Engenharia de Software são adotados?
4. Que processos, métodos e ferramentas de Design Participativo são adotados?
5. Como se dá a integração do Design Participativo à Engenharia de Software?
6. Quais são as partes interessadas envolvidas nesses estudos?
7. De que modo essas partes interessadas são envolvidas nesses estudos?

Os artigos semente usados nesse trabalho foram descobertos em bases de dados variadas e escolhidos a partir da experiência dos autores sobre o tema. Um critério essencial para as sementes era descrever uma proposta de *framework* baseado em Design Participativo para o desenvolvimento de software.

Para a etapa de *backward snowballing*, as referências dos artigos semente foram coletadas a partir das páginas das bases de dados ou dos próprios artigos. Na etapa de *forward snowballing*, o recurso *Cited By* da plataforma *Google Scholar* foi utilizado para identificar trabalhos que referenciassem uma ou mais sementes. Visando apoiar a análise da questão de pesquisa secundária, uma filtragem automatizada foi, então, aplicada na coleção completa de artigos encontrados para serem mantidos apenas aqueles que incluíssem as *strings* “framework” e “participatory design” (ou “design participativo”) em seus meta-dados. A coleta dos artigos para *snowballing* ocorreu em julho de 2023.

Já os critérios de inclusão e exclusão da Tabela 1 foram definidos para serem aplicados a partir da leitura do título e do resumo dos artigos. Enquanto os critérios de

inclusão estão vinculados diretamente aos objetivos dessa revisão, os critérios de exclusão estão vinculados a limitações operacionais da pesquisa ou reduzem significativamente um viés indesejado. Naturalmente, os artigos deviam atender a todos os critérios de inclusão para serem mantidos e os artigos que caíam em um ou mais critérios de exclusão foram eliminados. Além disso, antes da aplicação desses critérios, os artigos duplicados e os próprios artigos semente foram excluídos da coleção, pois não há necessidade de avaliar o mesmo artigo mais de uma vez e porque os artigos semente já seriam analisados.

**Tabela 1. Critérios de inclusão e exclusão de artigos.**

| <b>Critérios de inclusão</b>                                  |
|---|
| CI1. O artigo aborda desenvolvimento ou evolução de software. |
| CI2. O artigo inclui um estudo em domínio educacional.        |
| <b>Critérios de exclusão</b>                                  |
| CE1. O artigo não está publicado em inglês ou português.      |
| CE2. O artigo não é um estudo primário.                       |
| CE3. O artigo possui menos de 6 páginas.                      |
| CE4. O documento não é um artigo de conferência ou periódico. |
| CE5. Não foi possível obter acesso ao documento completo.     |

O processo de aplicação dos critérios de inclusão e exclusão foi realizado por dois dos autores deste artigo. No caso de divergências, uma terceira autora mais experiente entrou como juíza. Os artigos selecionados foram, então, integralmente lidos e analisados.

### 3.1. Artigos Semente

A Tabela 2 apresenta os títulos e as referências para os seis artigos semente, que são apresentados e discutidos a seguir.

**Tabela 2. Artigos semente usados no processo de *snowballing*.**

|  |                               |
|--|-------------------------------|
| CUTA4UML: bridging the gap between informal and formal requirements for dynamic system aspects                   | [Erfurth and Rossak 2010]     |
| Designing educational software with students through collaborative design games: The We!Design&Play framework    | [Triantafyllakos et al. 2011] |
| Software engineering for 'social good': integrating action research, participatory design, and agile development | [Ferrario et al. 2014]        |
| Semio-participatory framework for interaction design of educational software                                     | [Rosa and Matos 2016]         |
| PDEduGame: Towards participatory design process for educational game design in primary school                    | [Ismail and Ibrahim 2017]     |
| A participatory framework proposal for guiding researchers through an educational mobile app development         | [Tavares et al. 2020]         |

Erfurth e Rossak, com ênfase no subprocesso de Engenharia de Requisitos (ER), descrevem um *framework* de métodos, ferramentas e ciclos de *feedback*, que denominam CUTA4UML [Erfurth and Rossak 2010]. Baseado no Design Participativo, o *framework* estende uma técnica conhecida como CUTA (do inglês, *Collaborative User's Task Analysis*), de modo que esta possa incorporar a notação UML (e.g., Diagrama de Atividades). Seu propósito é melhorar a elicitación e a especificação de requisitos em

diferentes domínios, apoiando o envolvimento, forte e direto, de todas as partes interessadas, incluindo pessoas não técnicas.

Triantafyllakos, Palaigeorgiou e Tsoukalas apresentam um *framework* para o desenvolvimento de jogos digitais colaborativos, denominado *We!Design&Play* [Triantafyllakos et al. 2011]. Segundo os autores, inspirado na teoria de geração de ideias e na literatura de jogos de *design*, este pode ser adotado em sessões de DP para o projeto de aplicações educacionais com a participação de estudantes. O *framework* propõe uma abordagem exploratória baseada nas três perspectivas a seguir: Convergente, que busca garantir que conhecimentos prévios de *design* sejam considerados, contemplando o domínio da aprendizagem e soluções pré-existentes; Contextual, que leva em conta aspectos como lugares, configurações e condições específicas assim como perspectiva temporal; Divergente, que contempla aspectos tecnológicos e de criatividade, explorando o tensionamento e o rompimento de paradigmas.

Ferrario e demais autores apresentam um *framework*, denominado *Speed-play*, para gestão de projetos de software, baseado no DP e na pesquisa-ação, que tem como propósito apoiar o desenvolvimento de software voltado ao bem social [Ferrario et al. 2014]. Esse *framework* propõe uma abordagem ágil e iterativa, focada no usuário e na participação. Para os autores, os processos de software existentes nem sempre correspondem às necessidades dos projetos de software sociais, que buscam mudanças sociais e geralmente envolvem comunidades vulneráveis. O *framework*, organizado em quatro etapas – preparação, *design*, construção e sustentação –, prevê a participação de partes interessadas, inclusive usuários finais, na geração de ideias e *feedback*.

Rosa e Matos apresentam um *framework* semi participativo para o (re)*design* de interação de software educacional com o objetivo de melhorar a qualidade da interação a partir de uma abordagem culturalmente informada [Rosa and Matos 2016]. Esse *framework* está organizado nas três fases do Design Centrado na Comunicação (DCC), da Engenharia Semiótica, e incorpora técnicas do DP: i. Análise de Contexto, que tem como objetivo conhecer o contexto do usuário, adotando as técnicas *contextual inquiry* e *brainstorm* para que o usuário aborde problemas e soluções; ii. Engenharia da Interface, na qual são desenvolvidos protótipos de interface com apoio da técnica *braindraw*; e iii. Avaliação, que se propõe a avaliar a interface e a interação projetada com auxílio da técnica *think-aloud*. A partir da aplicação do *framework* proposto, os autores puderam conhecer o contexto do usuário (professores e estudantes) e suas práticas na interação com o software TecCiências, obtendo contribuições, como sugestões, críticas, novos protótipos de interfaces e retorno sobre esses protótipos.

Ismail e Ibrahim, a partir da meta-análise de uma revisão sistematizada de literatura sobre processos de DP, propõem um *framework* denominado PDEduGame com o propósito de guiar o *design* de jogos educacionais para os ensino fundamental [Ismail and Ibrahim 2017]. Este possui cinco processos principais: i. estudo inicial (5 atividades); ii. planejamento de oficinas de projeto de jogos educacionais (4 atividades); iii. oficinas de jogos educacionais – modelagem conceitual inicial (4 atividades), *storyboard* do jogo (3 atividades), finalização do *mock-up* do jogo (3 atividades); iv. prototipação do jogo educacional (2 atividades); v. teste do protótipo (5 atividades). As autoras sugerem sua adoção em escolas de ensino fundamental da Malásia.

Tavares, Vieira e Pedro, fundamentados na EDR (do inglês, *Educational Design Research*), propõem um *framework*, baseado no DP e no Design Centrado no Usuário (DCU), para organizar o desenvolvimento de aplicativos móveis educacionais e os produtos esperados desse desenvolvimento, a partir da percepção, ideias e necessidades de seus usuários finais [Tavares et al. 2020]. O referido *framework* organiza um processo iterativo de três etapas: i. Pesquisa preliminar, na qual são especificados o público-alvo, a abordagem de aprendizagem, o tópico de educação em ciência a ser tratado, a tipologia dos recursos educacionais digitais, os componentes de gestão da aprendizagem e o conceito do aplicativo; ii. Prototipação, no qual é produzido um guia de autoria com especificações didáticas, de funcionalidades e gráficas; e iii. Avaliação, no qual o protótipo é implementado e testado. Para apoiar a elaboração dos produtos mencionados, os autores explicitam as técnicas adotadas e as partes interessadas envolvidas, incluindo pesquisadores, estudantes, professores e especialistas. Os autores argumentam que o *framework* proposto, construído a partir da revisão de literatura, é suficientemente flexível e compreensível para ser adotado para o desenvolvimento de outras soluções educacionais.

Em síntese, entre os artigos semente, tem-se *frameworks* voltados ao desenvolvimento de software para diferentes domínios de aplicação [Erfurth and Rossak 2010] ou para o bem social [Ferrario et al. 2014], ao desenvolvimento de jogos digitais educacionais [Triantafyllakos et al. 2011, Ismail and Ibrahim 2017], ao desenvolvimento de aplicativos móveis [Tavares et al. 2020] e para o (re)design da interação de software educacional [Rosa and Matos 2016]. Considerando-se a integração do Design Participativo à Engenharia de Software, o *framework* proposto por Erfurth e Rossak apresenta ênfase no subprocesso de ER [Erfurth and Rossak 2010], enquanto que o de Ferrario e outros propõe uma abordagem ágil e iterativa [Ferrario et al. 2014]. Ambos os *frameworks*, entretanto, não são voltados ao domínio educacional.

## 4. Resultados e Discussão

Na etapa *backward snowballing*, obtiveram-se como resultados 50 artigos, e 133 artigos na etapa *forward snowballing*, considerando o filtro descrito na seção 3. Um artigo semente foi encontrado na etapa de *backward snowballing* e removido. Quatro duplicatas foram removidas também na etapa de *forward snowballing*. A combinação desses resultados totalizou 178 artigos identificados. Uma vez aplicados os critérios de inclusão e exclusão, chegou-se a cinco artigos provenientes da etapa *backward snowballing* e seis da etapa *forward snowballing*. A Figura 2 sumariza o processo de busca, triagem e seleção dos artigos. Os artigos selecionados são apresentados e discutidos a seguir.

### 4.1. Backward snowballing

Scaife e outros propõem um *framework* denominado *informant design* para maximizar as contribuições de diferentes partes interessadas – professores, crianças e membros da equipe de *design* – em diferentes etapas do processo de *design* [Scaife et al. 1997]. Ilustram sua aplicação no desenvolvimento de um software multimídia para ensinar conceitos de ecologia considerados difíceis. O referido *framework* é apresentado com apoio de uma tabela, que possui como cabeçalhos de coluna: Fase do Design, Informante/Contribuinte da Equipe de Design, Entrada, Métodos. Contudo, o texto não faz menção à qualidade de software ou à ES e, embora faça referência ao DP, dentre os métodos indicados, não explicita na tabela quais métodos adotam abordagem participativa. Quanto à relação entre

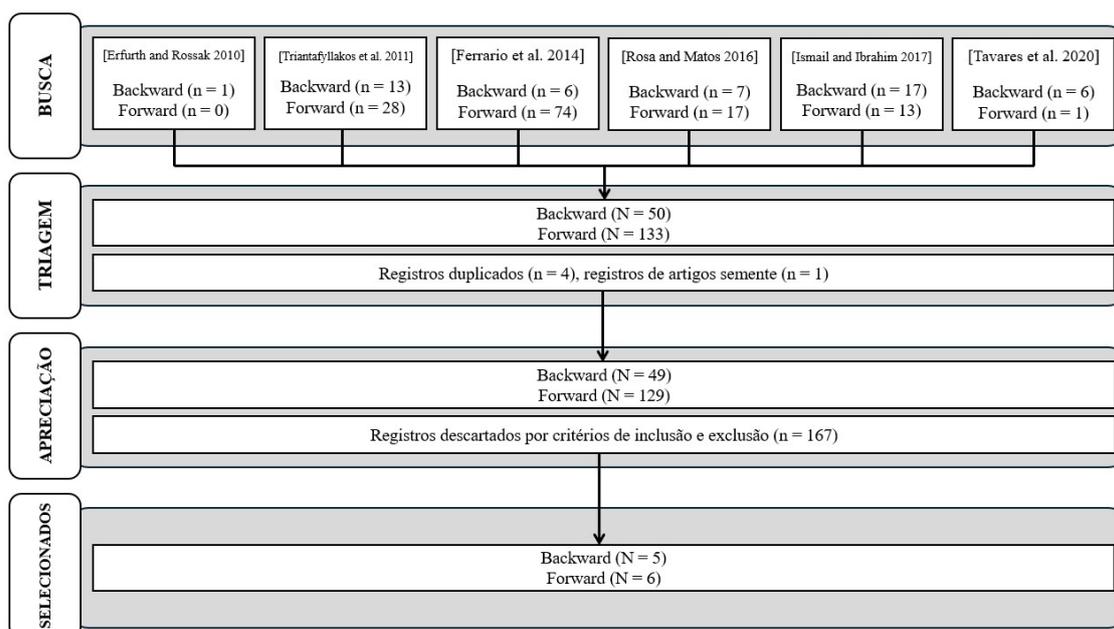


Figura 2. Diagrama com fluxo da revisão sistemática.

crianças e *designers*, assim como entre professores e *designers*, os autores se posicionam entre o DCU e o DP. Para os autores, crianças e professores, embora não sejam tratados como parceiros de *design*, podem revelar informações desconhecidas. Crianças conversam com outras crianças no contexto escolar com apoio de materiais existentes, desenhando cenários e jogos, realizando tarefas de aprendizagem. Já os professores são entrevistados e também experimentam protótipos, sugerindo como poderiam ser adotados no ensino.

Triantafyllakos, Palaigeorgiou e Tsoukalas apresentam uma metodologia, baseada em Design Participativo, considerada pelos próprios autores um *framework*, para o desenvolvimento de aplicações educacionais [Triantafyllakos et al. 2008]. No estudo em tela, abordam o *design* de um ambiente de avaliação eletrônica e de um *site* de um curso por estudantes de graduação. Denominada *We!Design*, a metodologia, organizada em duas fases, possibilita a cooperação entre estudantes computacionalmente letrados e *designers*: a primeira fase é dedicada à formulação de necessidades, tarefas e protótipos de interface; já a segunda envolve a análise e a integração das sugestões dos estudantes. Quanto à qualidade, os autores mencionam a usabilidade como responsabilidade de um *expert* em IHC. A única menção à ES diz respeito à experiência limitada dos estudantes com a área.

Siozos e demais autores discutem desafios envolvidos na elaboração de testes apoiados por computador e reportam o desenvolvimento de uma aplicação para a plataforma *Tablet PC*, denominada *MyTest*, conduzido com a metodologia participativa *We!Design* [Triantafyllakos et al. 2008], envolvendo 31 estudantes de ensino médio e professores na elaboração de propostas de *design* e avaliação formativa dessas propostas [Siozos et al. 2009]. Embora não mencionem explicitamente contribuições da ES, a partir das avaliações dos protótipos, abordam qualidade como promoção da utilidade e facilidade de uso, fazendo referência a aspectos como legibilidade e flexibilidade. Propõem, então, um *framework* para orientar o *design* de ferramentas voltadas à elaboração de testes apoiados por computador, que envolve: *design* com estudantes e professores; seleção

da plataforma de mídia adequada; e planejamento de uma evolução, ao invés de uma revolução, de práticas existentes.

Triantafyllakos, Palaigeorgiou e Tsoukalas apresentam uma abordagem que consideram inovadora de adoção de personagens fictícios no *design* colaborativo de software educacional. Trata-se da técnica *design alter egos*, que envolve a criação e o uso de personagens ficcionais na elicitação de requisitos e de ideias de *design*. A abordagem foi experimentada e avaliada no *design* de um *site* de curso ideal em sessões de DP com estudantes de graduação sem experiência em IHC, sendo que a maioria apresentava conhecimento limitado sobre ES [Triantafyllakos et al. 2010]. A qualidade é abordada do ponto de vista das contribuições realizadas pelos estudantes durante as sessões de *design*. Os autores avaliaram que a técnica contribuiu para minimizar o receio dos participantes de se sentirem expostos, valorizou e apoiou sua introspecção, estimulou sua criatividade, além de desenvolver uma atmosfera informal e construtiva nas sessões de *design*.

Israel e demais autores envolveram estudantes do 5º ano do ensino fundamental, com 11 anos de idade em média, na concepção de um aplicativo móvel educacional para o ensino de circuitos e eletricidade [Israel et al. 2013]. Para os autores, a qualidade é entendida a partir da percepção do usuário, que a consideram crítica para a sua satisfação. Além disso, ainda que não pormenorizados, acessibilidade e usabilidade são mencionados pelos autores. Adotam um processo de DP, particularmente de prototipação em papel, para compreender como esses estudantes concebem tecnologia para apoiar seu próprio aprendizado, contemplando seus conhecimentos e experiências com jogos e tecnologias, com o conteúdo em tela, além de suas preferências de aprendizagem, competências e dificuldades. O trabalho não menciona a ES.

#### 4.2. Forward snowballing

Tracy e Jordan relatam a experiência de um grupo de pesquisa interdisciplinar, denominado Ensemble (do inglês, *Semantic Technologies for the Enhancement of Case Based Learning*), de engajamento de estudantes universitários em colaboração com pesquisadores no desenvolvimento de tecnologias de websemântica para a educação no ensino superior [Tracy and Jordan 2012]. De acordo com as autoras, a abordagem de *design* adotada não está alinhada a qualquer processo pré-definido cientificamente de resolução de problema com fases gerais, mas combina uma série de papéis de pesquisa e de *design* de modo comunicativo e produtivo. Nessa perspectiva, a proposta não menciona qualquer metodologia em particular de DP tão pouco de ES. Os estudantes, contudo, envolveram-se como *designers* em usos pedagógicos das tecnologias relacionadas, interdisciplinaridade, modelagem, além da criação de conhecimento e elementos artísticos e práticos de visualização de dados. A qualidade é mencionada ao fazerem referência a trabalhos relacionados, contemplando fatores de qualidade como eficácia e eficiência, sendo a qualidade abordada do ponto de vista da confiabilidade da origem do conteúdo e dos metadados.

Perry e Schnaid investigam estratégias adotadas por *designers* digitais/gráficos no projeto de interfaces na área de química ao colaborarem com professores experientes em etapas iniciais do processo de *design* [Perry and Schnaid 2012]. Os autores mencionam usabilidade ao se referirem ao caráter interdisciplinar das interfaces de software educacionais, não sendo a qualidade de software abordada de forma explícita. Em duas equipes, formadas por um *designer* e um professor, foram realizados desenhos, adotando folha A4

e lápis de diferentes tipos e cores, que deveriam servir de ponto de partida para o desenvolvimento de *software* educacional voltado a estudantes do ensino médio. Os tópicos de conteúdo eram apresentados apenas no início da sessão, conduzida com apoio do método de pensar em voz alta (*think-aloud*). Em ambos os casos foram propostas soluções *web*.

Perry e demais autoras abordam o *design* de objetos de aprendizagem envolvendo a cooperação entre *designers* e professores, dando ênfase à comunicação entre essas partes interessadas [Perry et al. 2014]. O processo metodológico apresentado pelas autoras possui as seguintes etapas: Compreensão, que envolve o planejamento pelo professor; Preparação, na qual o professor entrega documentações relacionadas ao projeto, como mapas de *sites* ou *storyboards*; Projeto, no qual a equipe interdisciplinar analisa o planejamento do professor, chamando-o para uma entrevista, e iterativamente projeta a interface em contato com esse professor; Desenvolvimento, em que o projeto é implementado por um membro da equipe de *design* em contato com o professor; e Apresentação, quando o objeto de aprendizagem está pronto, é avaliado quanto a sua usabilidade e submetido à aprovação do professor antes de sua publicação.

Ekanem e outros apresentam um sistema *web* de gestão de resultados de estudantes e a metodologia adotada para fazê-lo, denominada modelo de processo incremental e participativo – PIP Model (do inglês, *Participatory Incremental Process Model*) [Ekanem et al. 2017]. O modelo propõe uma etapa inicial de ER com o envolvimento de partes interessadas, como desenvolvedor de software, supervisor, estudantes, oficiais de exames, gestores escolares e palestrantes. Nessa etapa são aplicadas técnicas como entrevistas, questionários, discussões em grupo, análise social, avaliação e inspeção de documentos para elicitare requisitos de usuário e de sistema. Então, iterativamente, são conduzidas as etapas de Design, Desenvolvimento, Teste e Entrega. Durante todo o ciclo de vida, é realizada a gestão de requisitos. Uma série de modelos é adotada para expressar aspectos como a decomposição funcional do sistema, processos e casos de uso. Fatores de qualidade como segurança e compatibilidade são mencionados.

Falcão e demais autores abordam congruências entre DP e Comunidades de Práticas, particularmente no contexto da Comunidade de Prática a DEMULTS – Desenvolvimento Educacional de Multimídias Sustentável, no *design* de jogos digitais educacionais [Falcão et al. 2017]. Estes são desenvolvidos por estudantes do ensino médio, denominados novatos, em colaboração com *experts* em *design*, programação, conteúdos curriculares diversos, para estudantes do ensino médio. A qualidade é aludida pelos autores do ponto de vista das expectativas para os jogos produzidos. Técnicas do DP, como *brainstorming* e *braindrawing*, são adotadas, especialmente no início de cada ciclo no DEMULTS. A ES é mencionada na apresentação do DP para caracterizá-lo como um campo multidisciplinar.

Farah e outros, visando a apoiar desenvolvedores na criação de aplicações *web* para a Educação, propõem um *framework*, de código aberto, baseado em software [Farah et al. 2022]. Este, de arquitetura similar ao *framework* JavaScript Gatsby, apresenta cinco componentes principais: (i) contexto, (ii) API (do inglês, *Application Program Interface*), (iii) modelos, (iv) CLI (do inglês, *Command Line Interface*) e (v) repositório. A avaliação do *framework* proposto foi realizada por doze desenvolvedores que o adotaram, julgando-o útil tecnicamente para o desenvolvimento de aplicações no domínio educacional. Contudo, os autores chegaram à conclusão da importância de sua adoção

em um contexto de colaboração com diferentes partes interessadas, como pesquisadores da área da Educação e educadores.

### 4.3. Discussão

Os estudos analisados envolvem a criação de diferentes tipos software para o domínio educacional, como software multimídia (1 artigo), diferentes tipos de soluções *web* (6), aplicativo móvel para testes apoiados por computador (1) e para o ensino de circuitos e eletricidade (1), jogos digitais (1) e objetos de aprendizagem (1). Os níveis de ensino contemplam tanto a educação básica (4 artigos) quanto o ensino superior (3).

Fatores de qualidade mencionados de forma explícita nos trabalhos analisados limitam-se à usabilidade (3 artigos), facilidade de uso (1), utilidade (1), acessibilidade (1), segurança (1) e compatibilidade (1). Ademais, a qualidade está associada às contribuições realizadas pelos usuários em sessões de *design* (1 artigo), a sua percepção (1) e expectativas (1), assim como à confiabilidade da origem do conteúdo e metadados (1). Percebe-se que, mesmo que a qualidade do produto de software seja mencionada na maioria dos estudos, diferentemente de uma abordagem sistematizada de ES, não é alvo de atenção.

Ainda que os trabalhos analisados proponham o desenvolvimento de produtos software, mesmo quando adotam abordagens sistematizadas, a maioria não faz referência explícita às contribuições da Engenharia de Software. A menção à área ocorre em dois artigos que se referem à experiência limitada em ES dos usuários finais envolvidos e em um artigo para caracterizar o DP como um campo multidisciplinar. Ademais, um artigo apresenta um modelo de processo com etapas e atividades bem delimitadas para o desenvolvimento de software e um artigo propõe um *framework* baseado em software. Por outro lado, é possível identificar contribuições do DP pela adoção de métodos (4 artigos) ou proposição de métodos (2) que contemplam a colaboração explícita dos usuários finais.

Percebe-se, assim, que embora os trabalhos analisados tratem do desenvolvimento de produtos de software, em sua maioria não há uma contribuição explícita da área de ES na sistematização dos processos com foco na qualidade, que é abordada de forma incipiente nesses trabalhos. Em uma perspectiva de ES, processos, métodos e ferramentas contribuem ao desenvolvimento de software visando a sua qualidade [Pressman and Maxim 2016, Sommerville 2019]. O desenvolvimento dos software mencionados é sistematizado a partir da valorização da contribuição das partes interessadas, com apoio de metodologias e técnicas que servem a esse propósito. Em Ekanem e outros, percebe-se a apresentação de um modelo para o desenvolvimento de software onde a participação dos usuários ocorre sobretudo na ER [Ekanem et al. 2017].

Os usuários finais, que atuam como informantes ou assumem papéis de coautoria, são sobretudo estudantes de diferentes níveis de ensino – ensino fundamental (2 artigos), ensino médio (2) e ensino superior (3) –, sendo mencionada também a participação de professores (4 artigos), supervisor (1), oficiais de exames (1), gestores escolares (1) e palestrantes (1). Estes colaboraram com *designers*, *experts* ou desenvolvedores de software.

As técnicas mencionadas para envolver os usuários finais como *codesigners* incluem prototipação em papel (2 artigos), *brainstorming* (1) e *braindrawing* (1). Além dessas técnicas, também foram propostas e validadas técnicas e metodologias pelos próprios autores, como *design alter egos* [Triantafyllakos et al. 2010] e *We!Design* [Triantafyllakos et al. 2008], sendo esta última adotada em um artigo adicional. Ainda, o

envolvimento dos usuários ocorre como informantes e não como parceiros de *design* em aproximação ao que seria uma abordagem de DCU (1 artigo), de partes interessadas que contribuem em atividades de ER (1), e de entrevistados e consultados iterativamente (1).

Dentre os *frameworks* identificados a partir da análise dos trabalhos encontrados das etapas de *backward* e *forward snowballing*, percebem-se contribuições em variados níveis de abstração: promover a participação de diferentes partes interessadas como informantes (1 artigo), orientar o *design* de testes apoiados por computador envolvendo estudantes e professores (1) e apoiar desenvolvedores na criação de aplicações *web* para a Educação (1). Além disso, o próprio Design Participativo é compreendido como um *framework* para o desenvolvimento de software [Triantafyllakos et al. 2008].

Nesses *frameworks* e demais trabalhos analisados, pode-se concluir que a integração entre Design Participativo e Engenharia de Software no desenvolvimento de software para o domínio educacional é ainda incipiente. Isso porque o desenvolvimento de software para o domínio educacional pode envolver diretamente, além de professores e estudantes, outros usuários e partes interessadas; não está restrito a um tipo de aplicação e transcende a adoção de *frameworks* baseados em software. Além disso, a qualidade de *software*, para além dos fatores mencionados, não é abordada de forma sistemática com apoio de métodos, processos e ferramentas.

## 5. Considerações Finais

Este artigo apresenta um estudo, baseado em uma revisão sistematizada de literatura, que investigou a integração do Design Participativo à Engenharia de Software no desenvolvimento de software para o domínio educacional. Para tanto, foi adotada a abordagem de busca *snowballing*, com um passo de *backward* e outro de *forward*. Partindo-se de seis artigos semente, foram selecionados cinco artigos da etapa *backward snowballing* e seis artigos da etapa *forward snowballing*.

Constatou-se na análise empreendida que a integração entre Design Participativo e Engenharia de Software no desenvolvimento de software para o domínio educacional é incipiente. Há espaço, portanto, para a proposição de um *framework* que oriente Engenheiros de Software na adoção do Design Participativo, abordando qualidade com apoio de processos, métodos e ferramentas para fazê-lo, inclusive com a participação de diferentes partes interessadas no produto de *software* em desenvolvimento. Espera-se que as reflexões apresentadas neste artigo contribuam nesse sentido.

Este trabalho apresenta uma pequena coleção de ameaças à validade comuns a revisões: (1) o viés de publicação (i.e., tendência de publicação de artigos com resultados positivos) é uma questão pertinente nessa revisão, porque optou-se por não incluir literatura cinzenta que poderia comprometer a confiabilidade de trabalhos selecionados e que não foram avaliados por pares; (2) embora as sementes escolhidas sejam representativas para as questões de pesquisa desta revisão, um único passo de *snowballing* pode ter levado à omissão de trabalhos relevantes que podem ser identificados por revisões complementares futuras; (3) de forma a reduzir os efeitos relacionados à imprecisão da seleção de estudos, a classificação de estudos selecionados (aplicação de critérios de inclusão e exclusão) foi realizada de forma independente por dois autores e os conflitos foram analisados e resolvidos por uma terceira autora; e (4) uma vez que os autores revisaram os resultados exaustivamente, erros de classificação e análise de dados são improváveis.

## Referências

- Barbosa, S. D. J. and Silva, B. S. d. (2010). *Interação Humano-Computador*. Elsevier, Rio de Janeiro.
- Ekanen, A. J., Ozuomba, S., and Jimoh, A. J. (2017). Development of students result management system: A case study of university of uyo. *Mathematical and Software Engineering*, 3(1):26–42.
- Erfurth, I. and Rossak, W. (2010). Cuta4uml: bridging the gap between informal and formal requirements for dynamic system aspects. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 2*, page 171–174, New York, NY, USA. ACM.
- Falcão, T. P. F., Oliveira, G. d. S., Andrade e Peres, F. M. d., and Morais, D. C. S. (2017). Design participativo de jogos digitais educacionais por adolescentes imersos em uma comunidade de prática. *Revista de Sistemas e Computação*, 7(2):189–205.
- Farah, J. C., Ingram, S., and Gillet, D. (2022). Supporting developers in creating web apps for education via an app development framework. In *HEAd'22 Conference Proceedings*, page 883–890. Valencia, Spain, Editorial Universitat Politècnica de València.
- Ferrario, M. A., Simm, W., Newman, P., Forshaw, S., and Whittle, J. (2014). Software engineering for 'social good': integrating action research, participatory design, and agile development. In *Companion Proceedings of the 36th International Conference on Software Engineering*, pages 520–523, New York, NY, USA. ACM.
- Grant, M. J. and Booth, A. (2009). A typology of reviews: an analysis of 14 review types and associated methodologies. *Health Information & Libraries Journal*, 26(2):91–108.
- Halskov, K. and Hansen, N. B. (2015). The diversity of participatory design research practice at pdc 2002–2012. *International Journal of Human-Computer Studies*, 74:81–92.
- Hamzah, A. and Wahid, F. (2016). Participatory design in the development of healthcare systems: a literature review. In *Proceedings of the 2nd International Conference on Communication and Information Processing*, page 60–64, New York, NY, USA. ACM.
- Hartson, R. and Pyla, P. (2019). *The UX book: Agile UX design for a quality user experience*. Morgan Kaufmann, second edition.
- Ismail, R. and Ibrahim, R. (2017). Pdedugame: Towards participatory design process for educational game design in primary school. In *2017 International Conference on Research and Innovation in Information Systems*, pages 1–6, New York, NY, USA. IEEE.
- Israel, M., Marino, M. T., Basham, J. D., and Spivak, W. (2013). Fifth graders as app designers. *Journal of Research on Technology in Education*, 46(1).
- Jalali, S. and Wohlin, C. (2012). Systematic literature studies: Database searches vs. backward snowballing. In *International Symposium on Empirical Software Engineering and Measurement*, pages 29–38, New York, NY, USA. ACM.
- Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical report, Keele University and University of Durham, UK.

- Maheu-Cadotte, M.-A., Dubé, V., Cossette, S., Lapierre, A., Fontaine, G., Deschênes, M.-F., and Lavoie, P. (2021). Involvement of end users in the development of serious games for health care professions education: Systematic descriptive review. *JMIR Serious Games*, 9(3).
- Muller, M. J., Haslwanter, J. H., and Dayton, T. (1997). *Handbook of Human-Computer Interaction*, chapter Participatory Practices in the Software Lifecycle. Elsevier, Amsterdam, second edition.
- Nesset, V., Vanderschantz, N., Stewart-Robertson, O., and Davis, E. C. (2024). Advocating for a more active role for the user in his participatory research: a scoping literature review. *Journal of Documentation*, 80(2):446–468.
- Perry, G. T., Bock, M. S., and Carneiro, M. L. F. (2014). It’s all about communication: Designing learning objects for an entire university. In *Proceedings of Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*, volume 1.
- Perry, G. T. and Schnaid, F. (2012). A case study on the design of learning interfaces. *Computers & Education*, (59):722–731.
- Poderi, G. and Dittrich, Y. (2018). Participatory design and sustainability: a literature review of pdc proceedings. In *Proceedings of the 15th Participatory Design Conference: Short Papers, Situated Actions, Workshops and Tutorial - Volume 2*, New York, NY, USA. ACM.
- Pressman, R. S. and Maxim, B. R. (2016). *Engenharia de Software: uma abordagem profissional*. Editora AMGH, Porto Alegre, eighth edition.
- Rogers, Y., Sharp, H., and Preece, J. (2011). *Design de Interação: além da interação humano-computador*. Bookman, third edition.
- Rosa, J. C. S. and Matos, E. (2016). Semio-participatory framework for interaction design of educational software. In *Proceedings of the 15th Brazilian Symposium on Human Factors in Computing Systems*, New York, NY, USA. ACM.
- Scaife, M., Rogers, Y., Aldrich, F., and Davies, M. (1997). Designing for or designing with? informant design for interactive learning environments. In *Proceedings of CHI 97*, pages 22–27.
- Siozos, P., Palaigeorgiou, G., Triantafyllakos, G., and Despotakis, T. (2009). Computer based testing using ”digital ink”: Participatory design of a tablet pc based assessment application for secondary education. *Computers & Education*, 52(4):811–819.
- Skoglund, M. and Runeson, P. (2009). Reference-based search strategies in systematic reviews. In *13th international conference on Evaluation and Assessment in Software Engineering*, pages 31–40, Durham, UK. BCS Learning & Development Ltd.
- Sommerville, I. (2019). *Engenharia de Software*. Pearson Universidade, São Paulo, tenth edition.
- Sumner, J., Chong, L., Bunde, A., and Wei Lim, Y. (2020). Co-designing technology for aging in place: A systematic review. *The Gerontologist*, 61(7):e395–e409.
- Tavares, R., Vieira, R., and Pedro, L. (2020). A participatory framework proposal for guiding researchers through an educational mobile app development. *Research in Learning Technology*, 28:1–21.

- Tracy, F. and Jordan, K. (2012). Students as designers of semantic web applications. *Technology, Pedagogy and Education*, 21(2):171–188.
- Triantafyllakos, G., Palaigeorgiou, G., and Tsoukalas, I. A. (2010). Fictional characters in participatory design sessions: Introducing the “design alter egos” technique. *Interacting with Computers*, 22(3).
- Triantafyllakos, G., Palaigeorgiou, G., and Tsoukalas, I. A. (2011). Designing educational software with students through collaborative design games: The we!design & play framework. *Computers & Education*, 56(1):227–242.
- Triantafyllakos, G. N., Palaigeorgiou, G. E., and Tsoukalas, I. A. (2008). We!design: A student-centred participatory methodology for the design of educational applications. *British Journal of Educational Technology*, 39(1):125–139.
- Tuhkala, A. (2021). A systematic literature review of participatory design studies involving teachers. *European Journal of Education*, 56(4):641–659.
- Tuhkala, A., Kärkkäinen, T., and Nieminen, P. (2018). Semi-automatic literature mapping of participatory design studies 2006–2016. In *Proceedings of the 15th Participatory Design Conference: Short Papers, Situated Actions, Workshops and Tutorial - Volume 2*, New York, NY, USA. ACM.
- Whitman, M., Hsiang, C.-y., and Roark, K. (2018). Potential for participatory big data ethics and algorithm design: a scoping mapping review. In *Proceedings of the 15th Participatory Design Conference: Short Papers, Situated Actions, Workshops and Tutorial - Volume 2*, New York, NY, USA. ACM.