

DOI:

A BIBLIOMETRIC ANALYSIS OF THE USE OF GAMES FOR TEACHING SOFTWARE ENGINEERING FROM A SYSTEMATIC LITERATURE REVIEW

UMA ANÁLISE BIBLIOMÉTRICA DA APLICAÇÃO DE JOGOS PARA ENSINO DE ENGENHARIA DE SOFTWARE A PARTIR DE UMA REVISÃO SISTEMÁTICA DA LITERATURA

Anderson Dos Santos Guerra

UFPA - UNIVERSIDADE FEDERAL DO PARÁ - ORCID: <https://orcid.org/0000-0001-7204-7041>

Edson Monteiro Neto

UNIFAP - UNIVERSIDADE FEDERAL DO AMAPÁ - ORCID: <https://orcid.org/0000-0003-0503-7200>

Julio Furtado

UNIVERSIDADE FEDERAL DO AMAPÁ - ORCID: <https://orcid.org/0000-0002-1984-9587>

Sandro Ronaldo Bezerra Oliveira

UFPA - UNIVERSIDADE FEDERAL DO PARÁ - ORCID: <https://orcid.org/0000-0002-8929-5145>

Abstract

Conducting a Systematic Literature Review (RSL) with the purpose of presenting a general assessment of the state of the art regarding the use of games and gamification to assist in teaching Software Engineering.

Identify what kind of games and gamified approaches are being used in the Software Engineering teaching process, in addition to identifying which areas of Software Engineering and competences each game or approach aimed to develop in the participants.

Systematic Literature Review

Through the RSL it was possible to recognize that there is a variety of games that are used for teaching HE, ranging from board games to role-playing games.

The RSL results that were presented in this work serve as a foundation for future works that aim to create games or approaches that help in teaching Software Engineering.

Show the main trends and ways used to deal with the teaching-learning process of ES with the use of games and gamified approaches.

Key words: Gamification, Serious Game, Software Engineering, Education, Teaching

Resumo

Realização de uma Revisão Sistemática da Literatura (RSL) com o propósito de apresentar um apurado geral sobre qual o estado da arte no que tange ao uso de jogos e gamificação para auxiliar no ensino de Engenharia de Software.

Identificar que tipo de jogos e abordagens gamificadas estão sendo utilizadas no processo de ensino de Engenharia de Software, além de identificar quais áreas da Engenharia de Software e competências cada jogo ou abordagem visou desenvolver nos participantes.

Revisão Sistemática da Literatura

Por meio da RSL foi possível reconhecer que existe uma variedade de jogos que são utilizados para o ensino de ES, variando desde jogos de tabuleiro até jogos de interpretação de personagens.

Os resultados da RSL que foram apresentados neste trabalho servem como alicerce para trabalhos vindouros que visam realizar a criação de jogos ou abordagens que auxiliem no ensino de Engenharia de Software.

Exibir as principais tendências e formas utilizadas para lidar com o processo de ensino-aprendizagem de ES com o uso de jogos e abordagens gamificadas.

Palavras-chave: Gamificação, Jogos Sérios, Engenharia de Software, Educação, Ensino

A Bibliometric Analysis of the Use of Games for Teaching Software Engineering from a Systematic Literature Review

ABSTRACT: Software engineering is a subject that allows academics to be able to carry out the planning and development of quality software. Among the difficulties encountered in their teaching, there is the identification of means of learning that can encourage academics, in addition to the need to find ways to exercise knowledge during classes through practices. In this context, it comes the importance of using games and effective gamified approaches so that the teaching-learning process can transmit its concepts. Thus, this work presents a systematic literature review on the use of games and gamification in the teaching of Software Engineering, identifying the main games and gamified elements used in their teaching. At the end of the review, 66 studies were identified; this result can serve for future research, indicating gamified approaches and types of games that can be created to assist in teaching Software Engineering.

Keywords: Gamification, Serious Game, Software Engineering, Education, Teaching.

Uma Análise Bibliométrica da Aplicação de Jogos para Ensino de Engenharia de Software a partir de uma Revisão Sistemática da Literatura

RESUMO: A Engenharia de software é uma disciplina que permite que acadêmicos consigam realizar o planejamento e desenvolvimento de um software de qualidade. Dentre as dificuldades encontradas no seu ensino, existe a identificação de formas de aprendizagem que consigam incentivar os acadêmicos, além da necessidade de encontrar formas de exercitar o conhecimento durante as aulas por meio de práticas. Nesse âmbito, entra a importância do uso de jogos e abordagens gamificadas eficazes para que o processo de ensino aprendizagem consiga transmitir os seus conceitos. Deste modo, esse trabalho apresenta uma revisão sistemática da literatura sobre o uso de jogos e gamificação no ensino da Engenharia de Software, identificando quais os principais jogos e elementos gamificados utilizados no seu ensino. Ao fim da revisão, foram identificados 66 trabalhos, este resultado pode servir para pesquisas futuras, indicando abordagens gamificadas e tipos de jogos que podem ser criados para auxiliar no ensino da Engenharia de Software.

Palavras-chave: Gamificação, Jogos Sérios, Engenharia de Software, Educação, Ensino.

Agradecimentos: Este trabalho pertence ao projeto SPIDER/UFPA (<http://www.spider.ufpa.br>).

1. INTRODUÇÃO

A Engenharia de Software (ES) é uma área abrangente, seu principal objetivo é fornecer uma base para o desenvolvimento profissional de software, indicando um conjunto de técnicas que auxiliem na especificação, projeto e evolução de programas (SOMMERVILLE, 2011). Além do aspecto técnico que envolve o desenvolvimento de um projeto, um profissional que atua na área de ES também precisa atentar-se a detalhes além da programação, que incluem a qualidade do software em desenvolvimento, os prazos estipulados e o orçamento estabelecido. Devido a isso, um engenheiro de software precisa tanto do conhecimento desses princípios quanto da experiência com o uso deles (IEEE/ACM, 2015).

Em um sentido amplo, uma aula dada por meio de uma abordagem expositiva não possui uma eficiência tão grande na efetivação do conhecimento, uma vez que apenas o sentido da audição costuma ser utilizado, em contraponto é possível afirmar que eventos que tentem simular problemas da realidade e atividades vivenciadas permitem que o participante consiga assimilar situações diferentes (PRIKLADNICKI *et al*, 2009).

O trabalho de WANGENHEIM e SILVA (2009) indica que a falta de profissionais de ES qualificados pode conter uma relação com as competências de um engenheiro de software não serem apropriadamente aprofundadas ao longo do ensino. Isso faz com que exista uma lacuna entre o que é ensinado na academia e o que a indústria necessita, uma vez que o profissional que se formou recentemente só irá efetivamente obter o conhecimento necessário quando começar a atuar na indústria de desenvolvimento de software.

O uso de jogos, tanto digitais quanto de tabuleiros, e atividades gamificadas podem influenciar de maneira positiva no processo de ensino-aprendizagem, uma vez que o participante tem a capacidade de adquirir novos conceitos, habilidades e conhecimentos por meio da experiência ao longo da execução dessas práticas (METTLER; PINTO, 2015). Com base nisso, esse trabalho tem como objetivo identificar que tipo de jogos e abordagens gamificadas estão sendo utilizadas no processo de ensino de Engenharia de Software, além de identificar quais áreas da Engenharia de Software e competências cada jogo ou abordagem visou desenvolver nos participantes.

Como forma de atingir o objetivo do trabalho, uma Revisão Sistemática da Literatura (RSL) foi realizada com o propósito de apresentar um apurado geral sobre qual o estado da arte no que tange ao uso de jogos e gamificação para auxiliar no ensino de Engenharia de Software. A RSL começou a ser projetada com a orientação de dois professores da área de Engenharia de Software. Inicialmente, foi definida uma questão de pesquisa principal, a qual foi guia para que fosse possível estabelecer outras seis questões específicas. Durante a coleta dos trabalhos foi feito o uso de critérios de exclusão e inclusão estabelecidos juntos dos profissionais da área. Uma RSL foi escolhida como a forma de pesquisa devido ser uma das formas mais sistemáticas e eficientes para realizar uma pesquisa em um determinado tema e compilar os resultados encontrados para que seja possível analisá-los (KITCHENHAM e CHARTERS, 2007).

Este trabalho encontra-se organizado da seguinte forma: a Seção 2 é responsável por apresentar a fundamentação teórica; a Seção 3 apresenta trabalhos que possuem relação com a pesquisa; a Seção 4 expõe a RSL e suas respectivas fases, que incluem o planejamento e criação do protocolo de revisão; na Seção 5 são apresentados os resultados obtidos através da execução da revisão; a Seção 6 apresenta as conclusões dessa pesquisa e

indicadores de trabalhos futuros; por fim, é possível visualizar um apêndice indicando os trabalhos que foram encontrados no processo da RSL.

2. FUNDAMENTAÇÃO TEÓRICA

Nesta seção de fundamentação teórica são apresentados tópicos importantes para o melhor entendimento do trabalho.

2.1. Engenharia de Software

A engenharia de software visa apoiar o desenvolvimento profissional de software, por meio das melhores práticas e métodos para a criação do produto de software (SOMMERVILLE, 2011). Deste modo, o ensino de aspectos técnicos e não técnicos da produção de software dão suporte para os profissionais e teóricos aplicarem e aperfeiçoarem as boas práticas de desenvolvimento e gerenciamento.

No entanto, a transmissão deste conhecimento é árdua e difícil, visto que o profissional necessita assimilar múltiplas técnicas e em quais projetos elas serão mais bem aplicadas, de modo que simular a complexidade dos projetos da vida real em um ambiente educacional pode ser impossível (GHEZZI e MANDRIOLI, 2005). Dessa forma, o objetivo geral da comunidade de educação de Engenharia de Software é prover uma sólida base pedagógica para os educadores, (CARVER *et al.*, 2003; GHEZZI e MANDRIOLI, 2005; PORTELA *et al.*, 2015).

2.2. Jogos e *Serious Game*

É possível definir um jogo como uma atividade recreativa e estimulante que envolve qualquer tipo de competição com um conjunto de regras, sendo estas regras executadas em um ambiente restrito, no qual existe envolvimento social em um espaço e tempo determinados (ZUCARELLI e COUTO, 2013).

Por outro lado, um *serious game* é propriamente um jogo, no entanto o seu propósito primário não é voltado para entretenimento, mas sim algum outro objetivo, normalmente o ensino de determinado assunto ou no treinamento de habilidades que podem ser utilizadas no mundo real, de modo que estes propósitos sejam atingidos por meio do uso de conflitos e desafios ao jogador (METTLER e PINTO, 2015; ZYDA, 2005).

2.3. Práticas gamificadas

Segundo DETERDING (2011a, 2011b), a gamificação pode ser definida como o uso de elementos de jogos em contextos que não estão necessariamente relacionados a jogos. Os elementos possuem fator de motivação que estimulam o foco do usuário. Diferentemente de uma atividade avaliativa tradicional, em um ambiente gamificado é possível gerenciar o nível de dificuldade, aumentar o engajamento por meio de competição ou cooperação, possibilitar a experimentação de novas soluções sem o risco de perda e outros fatores correlatos.

3. TRABALHOS RELACIONADOS

Entre os trabalhos relacionados a esta pesquisa, existe a pesquisa de ALHAMMAD e MORENO (2018) que é feita uma revisão da literatura para verificar as práticas de gamificação no desenvolvimento de software ágil. Entre os achados, a pontuação e as medalhas estão entre os elementos gamificados mais usados. A prática mais gamificada em

métodos ágeis são as *user stories*, apesar de não ser discutida nos trabalhos a razão por esta preferência. No presente trabalho os elementos gamificados utilizados na área de engenharia de software são explorados de maneira mais abrangente, não apenas para uma área de conhecimento específica.

O trabalho de ZAMBON e THIRY (2018) foca na realização de uma revisão sistemática que visa identificar como é realizada a documentação de atividades lúdicas que são utilizadas para jogos educativos no contexto do ensino de Engenharia de Software para, em um trabalho futuro, elaborar um catálogo que apresente um mapeamento dos objetivos de aprendizagem relacionados aos níveis de conhecimento da Taxonomia de Bloom. A principal diferença da pesquisa citada anteriormente com o atual trabalho é que este não possui um foco primário em analisar como é feita a documentação das práticas identificadas, mas sim em reconhecer o maior número não apenas de jogos, mas também de elementos gamificados utilizados no ensino de Engenharia de Software.

Na pesquisa de SOUZA *et al.* (2017) é realizado um mapeamento sistemático onde são retornados 106 trabalhos no uso de *serious game*, gamificação e desenvolvimento de jogos para a educação de engenharia de software. Além das áreas de conhecimento, é feito o comparativo do uso dos diferentes métodos de jogos, como gamificação, *serious game* (GBL – *Game Based Learning*) e aprendizagem baseada no desenvolvimento de jogos (GDBL – *Game Development Based Learning*). Como resultado, as áreas de conhecimento mais abordadas no uso de jogos para ensino são “Processo de Software”, “Design de Software” e “Práticas Profissionais”. GBL e GDBL são os métodos mais utilizados, porém suas aplicabilidades dependem da área de conhecimento, sendo GDBL mais usado em “Design de Software” e GBL em “Processo de Software”.

O diferencial deste trabalho com relação ao de SOUZA *et al.* (2017) é a atualização temporal sobre as matérias mais utilizadas; os elementos gamificados mais abordados; as formas de aplicação, sendo experimentação ou grupo de controle; e as competências e os assuntos específicos mais trabalhados, podendo ser por exemplo *soft skill*, GSE (*Global Software Engineering*), métodos ágeis e outros. Outra diferença é a exclusão de trabalhos que abordam GDBL, visto que, segundo os autores, o uso de jogos prontos para desenvolvimento não difere muita da abordagem de desenvolver outros projetos não relacionados a jogos, como por exemplo o desenvolvimento de um sistema de caixa ou uma calculadora, onde a regra de negócio já está estabelecida. Por outro lado, *serious game* e gamificação exigem do autor a capacidade de criar regras e mecânicas que apliquem elementos de jogos em ambientes não relacionados a jogos, fator que expõe a complexidade e flexibilidade frente a apenas usar um exemplo de jogo pronto para desenvolvimento.

4. A REVISÃO SISTEMÁTICA DA LITERATURA

Em direção ao alcance de um panorama abrangente sobre vários aspectos do uso de jogos para ensino de engenharia de software, foi escolhido o método de Revisão Sistemática da Literatura. O intervalo temporal dos trabalhos selecionados nesta pesquisa limita-se ao período de 07/2007 até 04/2021, por entendermos que um período de 15 anos traz novas amostragens para o contexto da pesquisa e para estender os períodos dos trabalhos relacionados.

4.1. Planejamento

Foi executado o planejamento da Revisão Sistemática da Literatura com o propósito de reconhecer os estágios essenciais para a realização clara da pesquisa, análise e sumarização dos artigos, assim como os artefatos concebidos em cada etapa. Este trabalho foi planejado e executado com a orientação de dois profissionais da área de Engenharia de Software que possuem experiência na realização de revisões sistemáticas da literatura, visando a elaboração de um Protocolo de Revisão Sistemática da Literatura (PRSL).

Os procedimentos do PRSL foram estabelecidos por meio das reuniões entre os pesquisadores e os professores. Os procedimentos podem ser divididos em 8 atividades:

- 1) Definição das questões de pesquisa;
- 2) Escolha das bases de dados;
- 3) Definição da *String* de Busca;
- 4) Definição dos critérios de inclusão e exclusão;
- 5) Execução das buscas nas bases de dados;
- 6) Análise e seleção dos artigos;
- 7) Fichamento dos artigos;
- 8) Sumarização dos dados.

Os passos supracitados podem ser visualizados na Figura 1 em suas respectivas ordens de execução, assim como o relacionamento dos artefatos com suas atividades no decorrer do processo.

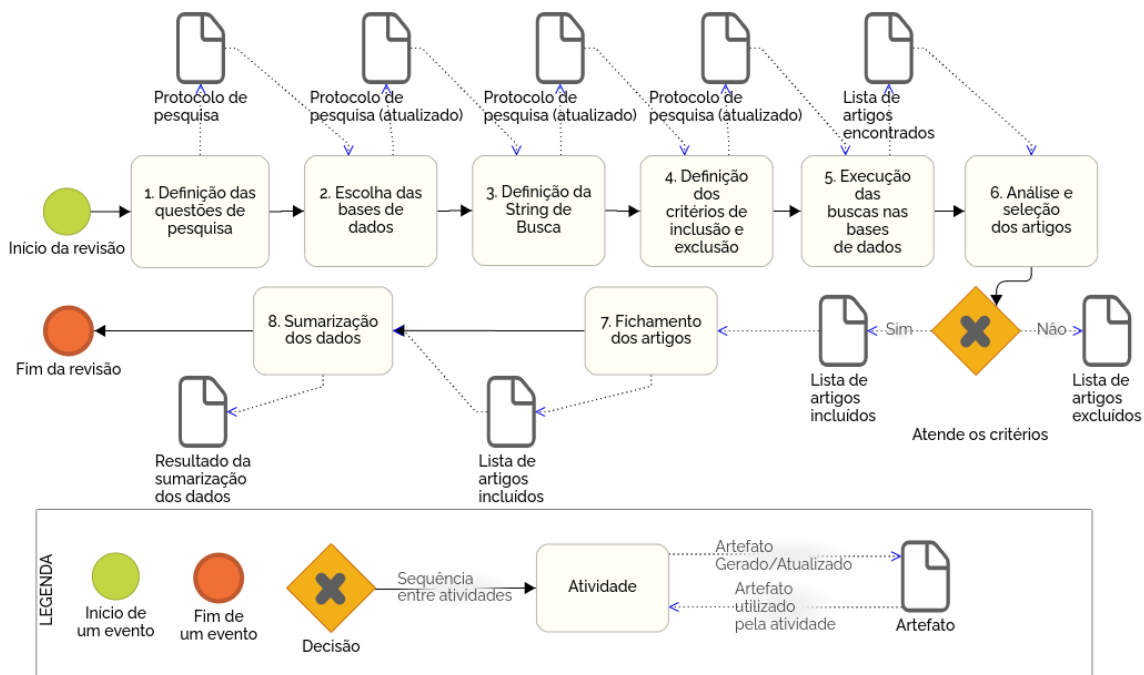


Figura 1 - Etapas da revisão sistemática

Fonte: Elaboração própria (2021).

4.1.1. Questões da Pesquisa

A aplicação de questões no processo de revisão sistemática busca desenvolver uma ferramenta norteadora capaz de delimitar o âmbito da revisão realizada, definida em

questão principal (QP) e questões secundárias (QS). A QP objetiva orientar a realização da pesquisa de forma geral e, tendo em vista que o estado da arte do uso de jogos no ensino de engenharia de software aborda vários aspectos, faz-se necessária a utilização de escopos com funções de delimitação para maior detalhamento. Deste modo, as QS possuem o propósito de especificar quais aspectos serão selecionados para que seja possível responder a questão principal. As seguintes questões foram realizadas:

- QP: Como se encontra o ensino de engenharia de software a partir do uso de jogos?
 - A QP tem como objetivo identificar qual o estado da arte em relação ao ensino de ES quando utilizados jogos (podendo tanto ser digital, quanto de tabuleiro) e elementos gamificados;
- QS1: Quais são as áreas da Engenharia de Software contempladas?
 - A QS1 é responsável por auxiliar na identificação de quais áreas da ES são contempladas pelos jogos e elementos gamificados encontrados nos trabalhos retornados;
- QS2: Quais são os tipos de jogos comumente usados?
 - A QS2 visa não somente especificar quais os tipos de jogos foram encontrados nos trabalhos retornados, mas também quais elementos foram utilizados caso tenha sido utilizada uma abordagem gamificada para o ensino;
- QS3: Quais as formas de aplicação dos jogos?
 - Esta questão secundária auxilia na identificação de quais são as formas de aplicação utilizadas nas abordagens dos trabalhos retornados na RSL;
- QS4: Qual tipo de conhecimento o jogo visa treinar?
 - Por meio da QS4 é realizado o reconhecimento de quais tipos de conhecimentos os trabalhos retornados possuíam como objetivo primário ao utilizarem os jogos e elementos gamificados;
- QS5: O jogo foi criado com o intuito de ser comercial ou acadêmico?
 - A QS5 propõe-se a discernir se os trabalhos retornados possuem um viés acadêmico ou voltado para a indústria.

4.1.2. Definição das Bases de Dados

As bases selecionadas para a busca foram ACM Digital Library e IEEE Xplore, devido à confiabilidade e grande quantidade de trabalhos publicados na área de Engenharia de Software; além de serem referências para a elaboração de base curricular em cursos de computação. Entre os critérios estabelecidos para a definição da base também é possível elencar:

- Disponibilidade de acesso por meio da internet;
- A disponibilidade destas bases para consulta/download gratuitos dos trabalhos para estudantes da UFPA.

Esta pesquisa executou buscas automatizadas nas bases ACM Digital Library e IEEE Xplore por meio do uso da *string* de busca estabelecida no processo.

4.1.3. String de Busca

Neste trabalho foi aplicado o método PIO - *Population* (Tipo de Participantes), *Intervention* (Tipo de Intervenção) e *Outcomes* (Tipo de resultados), para a busca das

palavras-chave que irão compor a *string* de busca. O Quadro 1 exibe a execução do método em acordo com os propósitos deste trabalho.

Quadro 1 - Método PIO.

<i>Population</i> (Participantes)	Profissionais e acadêmicos da área de engenharia de software
<i>Intervention</i> (Intervenção)	Uso de jogos e elementos gamificados para o ensino de engenharia de software
<i>Outcomes</i> (Resultados)	1) Áreas e conhecimentos de engenharia de software contemplados; 2) Tipos e formas de aplicação dos jogos; 3) Resultado das aplicações;

Fonte: Elaboração própria (2021).

Com base na questão principal e nas questões secundárias que foram estabelecidas, foi realizada uma combinação de palavras-chave para estabelecer uma *string* de busca que seja relevante. Para a validação da *string* foram utilizados os seguintes critérios: maior número de resultados obtidos por meio das bases utilizadas; e, para cada fonte, foram determinados trabalhos deveriam ser retornados, caso algum trabalho não fosse retornado, era necessário recalibrar a *string* de busca.

O conjunto de palavras-chave por meio de operadores lógicos “OR” e “AND” é o que representa uma *string* de busca. O emprego do operador “OR” é necessário para associar palavras-chave e sinônimos, baseado no método de pesquisa aplicado, ao mesmo tempo que o emprego do operador “AND” é indicado no momento que se pretende associar um grupo de palavras-chaves que foram definidos para todos os elementos (MALCHER *et al.*, 2015). Segue o Quadro 2 que apresenta a *string* de busca gerada.

Quadro 2 - *String* de busca

(Software engineering) AND (teach* OR learn* OR train* OR education OR tutoria OR evaluat* OR assess*) AND (Digital game OR electronic game OR table game OR board game OR Serious game OR Gamification)
--

Fonte: Elaboração própria (2021).

Entre os trabalhos pesquisados havia publicações de conferências, simpósios, workshops e periódicos científicos no tempo de julho de 2007 até abril de 2021, abrangendo um intervalo de 14 anos.

4.1.4. Critérios de Inclusão e Exclusão

Com a implementação da busca automatizada nas bases de dados, foi iniciada a etapa de análise e seleção dos trabalhos (6ª atividade). Dois pesquisadores efetuaram o processo de pré-seleção, onde na primeira etapa foi realizada a verificação dos títulos, dos resumos e das palavras-chave. Na segunda etapa foi realizado o processo de leitura vertical dos trabalhos com o objetivo de verificar se estes enquadravam-se nos critérios de exclusão e inclusão precedentemente acordados. O Quadro 3 apresenta os critérios.

Quadro 3 - Critérios de seleção dos estudos primários

ID	Critério de Inclusão (CI) ou Critério de Exclusão (CE)
CI 1	Aplicação de jogos em engenharia de software.
CI 2	Relatos de experiência da aplicação dos jogos na academia ou na indústria, a partir de uma pesquisa teórica ou experimental, considerando trabalhos completos ou em andamento.
CE 1	Trabalhos que não estão disponíveis para consulta/download gratuitos para estudantes da UFPA.
CE 2	Trabalhos que não foram encontrados com o uso das <i>strings</i> de buscas nas bases de pesquisa.
CE 3	Trabalhos que não estão incluídos nas bases de pesquisa.
CE 4	Trabalhos que violem os princípios de algumas das bases de pesquisa.
CE 5	Apenas a primeira aparição de um trabalho será incluída caso o mesmo trabalho surja repetidamente em mais de uma base de pesquisa.
CE 6	Somente será considerada a versão mais recente ou completa de um trabalho duplicado, exceto nos casos em que houver informações complementares.
CE 7	Trabalhos que não estejam na língua inglesa ou portuguesa.
CE 8	Trabalhos classificados como abstracts, <i>keynote speeches</i> , <i>courses</i> , <i>tutorials</i> ou <i>posters</i> .
CE 9	Trabalhos que não apresentem o uso de jogos/atividades gamificadas em engenharia de software.
CE 10	Trabalhos que não possuem relevância para as questões de pesquisa.

Fonte: Elaboração própria (2021).

5. RESULTADOS DA REVISÃO SISTEMÁTICA DA LITERATURA

A atividade realizada nesta seção faz referência aos resultados e à discussão obtidos por meio da execução do planejamento, objetivando o interesse de solucionar as QP e QS apresentadas anteriormente.

5.1. Resultados dos Estudos Primários

As *strings* foram aplicadas nos motores de busca ACM Digital Library e IEEE Xplore, apresentando um retorno de 1010 trabalhos. A base de dados ACM Digital Library teve 727 (72%), apresentando o maior número de artigos retornados pela busca, enquanto a base IEEE Xplore retornou 283 (28%) dos artigos. A Figura 2 apresenta os valores absolutos retornados.

Já a Tabela 1 exibe o total de artigos classificados dentro dos critérios de pré-seleção, CE e CI, em que cada linha resulta no total de estudos retornados de cada indexador. A pré-seleção abrange a maior quantidade de artigos (865) dentre as pesquisas eliminadas, seguido pelos CE.9 e CE.5 com respectivos 29 e 7 exclusões. Os dados que constam na coluna de CI.1 e CI.2 indicam os trabalhos que passaram nos dois Critérios de Inclusão estabelecidos.

Ainda na Tabela 1, o motor de busca ACM Digital Library apresentou a maior pré-seleção de artigos retirados (662), seguido do CE.9 como principal critério de eliminação de artigos (15), enquanto o IEEE Xplore excluiu 203 artigos em sua pré-seleção, tendo o CE.9 como o critério com maior taxa de eliminação de trabalhos (14).

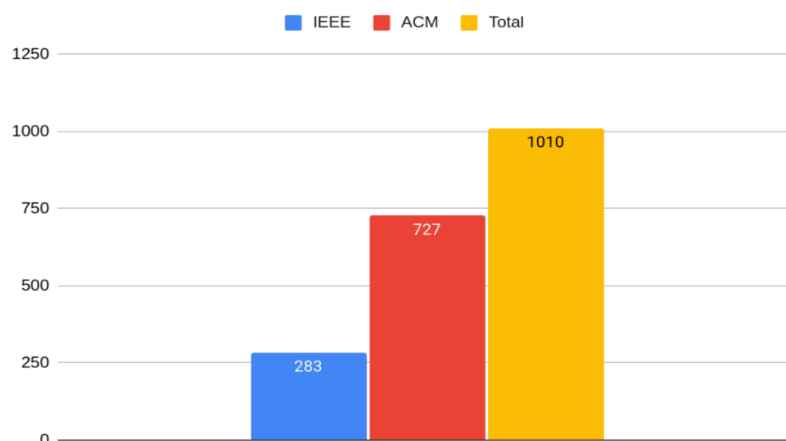


Figura 2 - Número de estudos retornados pelos motores de busca
Fonte: Elaboração própria (2021).

Tabela 1 - Estudos retornados com base nos critérios de inclusão e exclusão

Fonte	Estudos Primários								
	Trabalhos retornados	Pré-seleção	CE.5	CE.6	CE.7	CE.8	CE.9	CE.10	CI.1 CI.2
IEEE Xplore	283	203	0	2	4	3	14	2	41
ACM Digital Library	727	662	7	2	0	3	15	0	25
Total	1010	865	7	4	4	6	29	2	66

Fonte: Elaboração própria (2021).

O resultado da seleção dos estudos foi de 66 artigos publicados entre os períodos de 2007 até 2021. A Figura 3 mostra a quantidade de artigos encontrados por ano, sendo que os anos com maior número de publicações foram 2017, com 15 artigos, e 2019, 13 artigos. Os anos de 2007, 2011, 2012 e 2021 tiveram a menor taxa de publicação de artigos, sendo de 1 para cada ano.

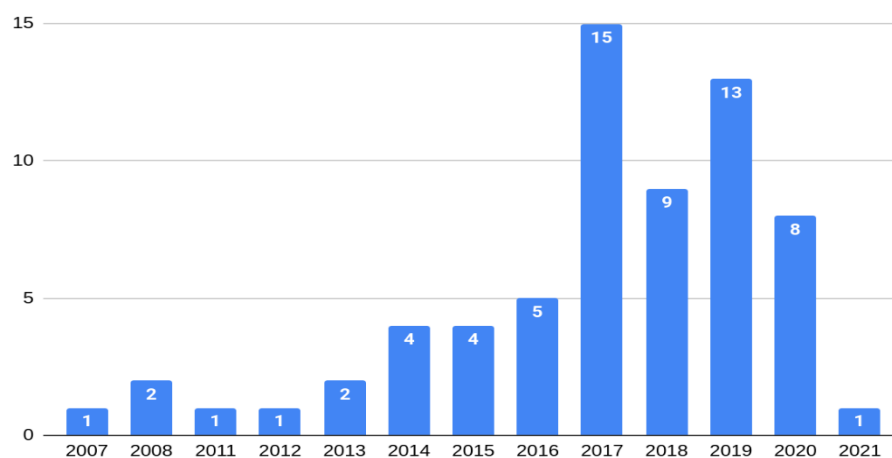


Figura 3 - Publicações por ano
Fonte: Elaboração própria (2021).

Em se tratando da distribuição dos tipos de publicação, o número de maior valor está relacionado a conferências, com 65,2%, seguido pelos simpósios (22,7%), periódicos (7,6%) e workshop (4,5%), como apresentado pela Figura 4.

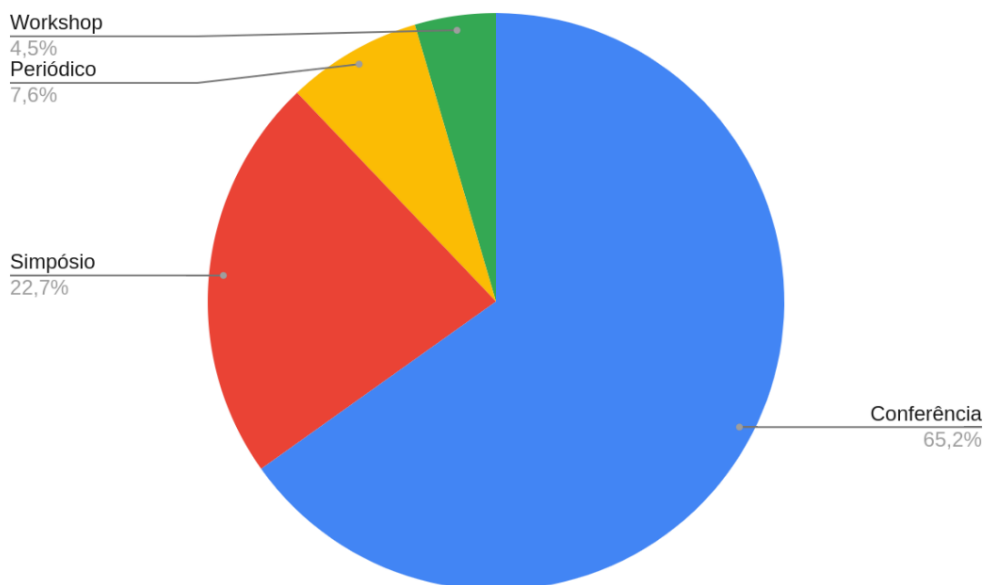


Figura 4 - Tipo de eventos
Fonte: Elaboração própria (2021).

Apesar do maior número de trabalhos publicados estar presente em conferências, não é possível destacar que isto possua relação com os objetivos deste trabalho. No entanto, pode estar mais relacionado à velocidade de publicação que cada veículo possui, sendo o período de publicação de periódicos mais prolongado que conferências e simpósios. Os workshops por sua vez trabalham mais com treinamentos, artigos curtos e tutoriais, e este trabalho busca mais os resultados do uso de jogos e não tanto o tutorial de como aplicar.

A Tabela 2 classifica os eventos de participação de acordo com o tipo de publicação mencionados anteriormente. No caso da conferência, a maior quantidade de trabalhos (6) esteve presente na “*Conference on Software Engineering Education and Training*” (CSEE&T), diferenciando-se do simpósio, em que houve igualdade amostral numérica entre “*Brazilian Symposium on Software Engineering*” (SBES) e “*Brazilian Symposium on Software Quality*” (SBQS) de 6 artigos, e dos periódicos, com os grupos ACM *Transactions on Computing Education* (TOCE) e IEEE *Transactions on Education* (ToE) apresentando 2 artigos para cada um. Por fim, todos os eventos dos workshops tiveram a mesma quantidade de artigos (1).

Também faz-se necessário destacar os principais locais relacionados ao tema de utilização de jogos na engenharia de software. Não menos importante, a maior quantidade dos locais de publicação indica a maturidade do tema abordado.

Tabela 2 - Locais de publicação.

Conferência	Número de Artigos
Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering	2
International Conference on Software Engineering Companion (ICSE-C)	2
International Requirements Engineering Conference (RE)	2
Frontiers in Education Conference (FIE)	3
Global Engineering Education Conference (EDUCON)	4
Conference on Software Engineering Education and Training (CSEE&T)	6
Simpósio	Número de Artigos
Technical Symposium on Computer Science Education (SIGCSE)	1
Symposium on Applied Computing (SAC)	2
Brazilian Symposium on Software Engineering (SBES)	6
Brazilian Symposium on Software Quality (SBQS)	6
Periódico	Número de Artigos
IEEE Latin America Transactions (IEEE LATAM)	1
ACM Transactions on Computing Education (TOCE)	2
IEEE Transactions on Education (ToE)	2
Workshop	Número de Artigos
International Workshop on Software Engineering Curricula for Millennials (SECM)	1
International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)	1
International Workshop on Games and Software Engineering: Realizing User Engagement with Game Engineering Techniques (GAS)	1

Fonte: Elaboração própria (2021).

Ademais, com relação às universidades, aos autores e aos países com maior número de artigos publicados sobre o tema, todos podem ser conferidos na Tabela 3. No critério de contagem foram catalogadas todas as aparições dos elementos, independentemente de ser autor ou coautor do trabalho. No que tange às universidades, a Universidade Federal do Ceará foi a que apresentou mais artigos com relação ao tema da pesquisa (4). Em relação aos países que mais publicaram, o Brasil foi o que mais publicou pesquisas (23). Já se tratando de autores que mais apareceram, o que possui maior frequência nos trabalhos foi Javier Gonzalez-Huerta, com 3 trabalhos.

Os dados da Tabela 3 são relevantes na apresentação das principais universidades, autores e países que mais contribuíram com o tema abordado neste trabalho, pois servem como fonte de orientação ou cooperação para desenvolvimento de pesquisas relacionadas.

Tabela 3 - Universidades, Autores e Países que possuem a maior quantidade de trabalhos.

Universidades	Artigos por Universidade
Rio de Janeiro State University	2
Federal University of Pará	2
Federal University of São Carlos	2
Blekinge Institute of Technology	3
Federal University of Ceará	4
Autores	Artigos por Autor
Sandro Ronaldo Bezerra Oliveira	2
Vera Maria Benjamin Werneck	2
Anna B. S. Marques	2
Fabiano Cutigi Ferrari	2
Javier Gonzalez-Huerta	3
Países	Artigos por Países
Finland	3
United Kingdom	3
Germany	6
USA	7
Brazil	23

Fonte: Elaboração própria (2021).

5.1.1. QS1: Quais são as áreas da Engenharia de Software contempladas?

A área de conhecimento mais abordada nos artigos que usam jogos para ensino é o Processo de Software, com o total de 45 aparições. Destes, a maior parte focava em apresentar o conteúdo inicial base e mais generalista para os alunos. Dentre os assuntos principais há a preocupação de transmitir conceitos de gerenciamento; práticas e ferramentas de métodos ágeis; habilidades de *soft skill*; e conceitos gerais de engenharia de software. A Figura 5 mostra as principais áreas contempladas pelas pesquisas encontradas.

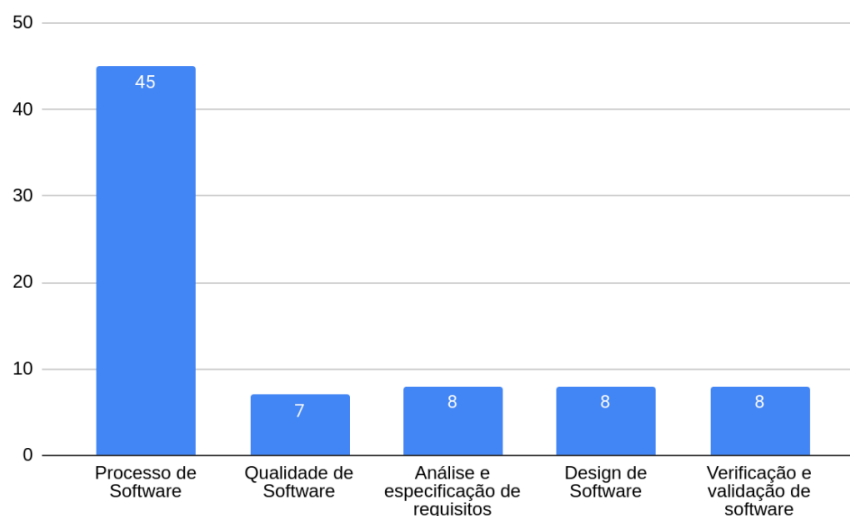


Figura 5 - Áreas mais abordadas para uso de jogos na engenharia de software.

Fonte: Elaboração própria (2021).

5.1.2. QS2: Quais são os tipos de jogos comumente usados?

O número de artigos citado por tipo de jogos pode ser visto na Figura 6, onde é possível identificar que *board games* foi o tipo de jogo com maior predominância, sendo utilizado em 17 trabalhos, seguido de *serious game* e RPG, com 9 e 4 trabalhos, respectivamente.

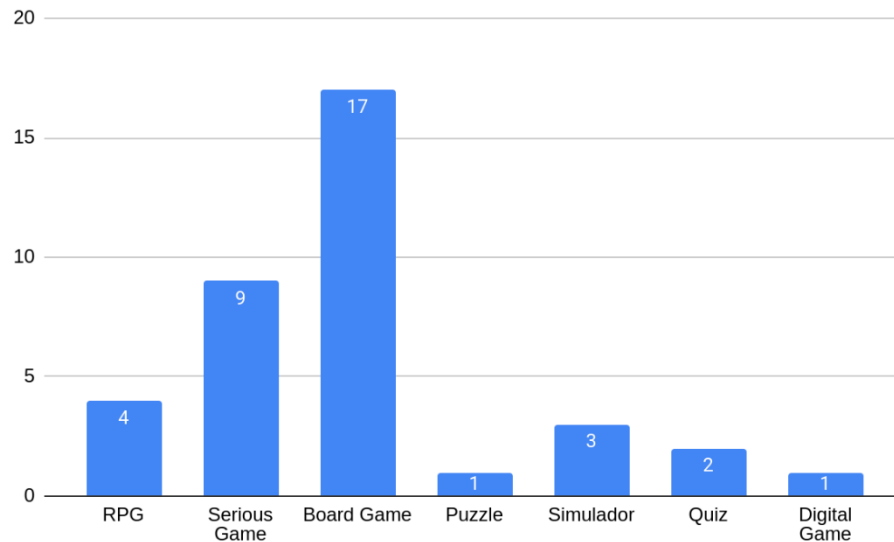


Figura 6 - Tipos de jogos
Fonte: Elaboração própria (2021).

O número de artigos citados por elementos gamificados pode ser visto na Figura 7, onde ao todo foram identificados 68 elementos que foram utilizados nos trabalhos retornados. O elemento de pontuação foi o mais trabalhado pelos autores, com 36 aparições, seguido de elemento de times, com 29 aparições, e uma distribuição igualitária de colaboração, elemento de progresso e elemento de *ranking*, com 25 aparições cada um.

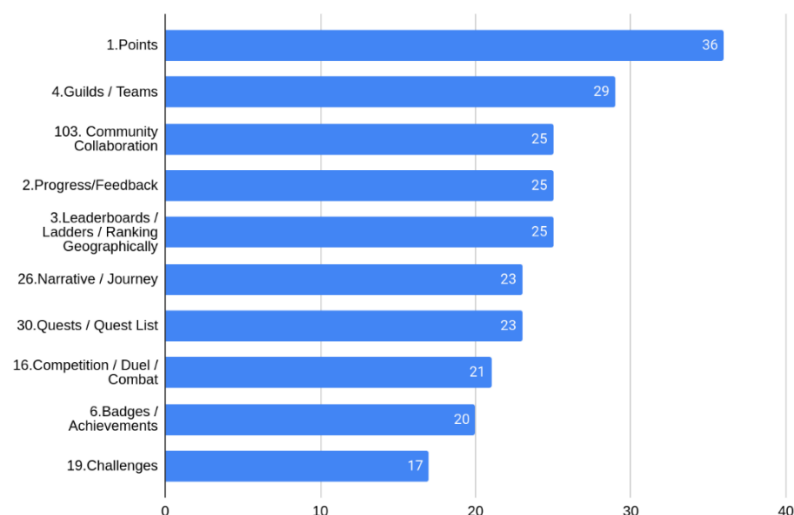


Figura 7 - Elementos gamificados
Fonte: Elaboração própria (2021).

5.1.3. QS3: Quais as formas de aplicação dos jogos?

A forma de aplicação destes jogos de ensino e atividades gamificadas foi distribuída entre experimentação e estudo de controle. Devido ao caráter criativo que envolve o desenvolvimento dos *serious games* ou das atividades gamificadas, vários trabalhos apresentam a necessidade de testar a abordagem com vários grupos de usuários para receberem *feedback* sobre o uso do jogo ou da atividade. A experimentação dá mais liberdade para os autores colherem estes dados e aperfeiçoarem quais aspectos e elementos gamificados foram mais eficazes segundo os alunos e os testes pós experimento.

Os estudos de controle, por outro lado, exigem maior planejamento e tempo para a execução, sem desconsiderar o maior número de usuários necessários para a realização do experimento. Distintamente, há o cenário utilizado para o estudo de controle, ao qual implementa o comparativo entre um grupo de estudo que utiliza métodos de gamificação e outro grupo que utiliza apenas o método tradicional de ensino. Com base nestes fatores, finda por ser mais comum os trabalhos de experimentação, que testam a viabilidade dos protótipos das atividades gamificadas, pois caso haja algum desvio, este é logo tratado. Os estudos de controle são reservados para os projetos que passaram pela fase de experimentação e prototipagem, onde já possuem maturidade para serem expostos em comparação com uma metodologia tradicional de ensino.

Como é possível visualizar na Figura 8, a forma de aplicação experimental é a que apareceu em mais trabalhos, sendo descrita em 53 dos artigos retornados pela revisão. No caso do estudo de controle, este teve sua aparição em 13 pesquisas.

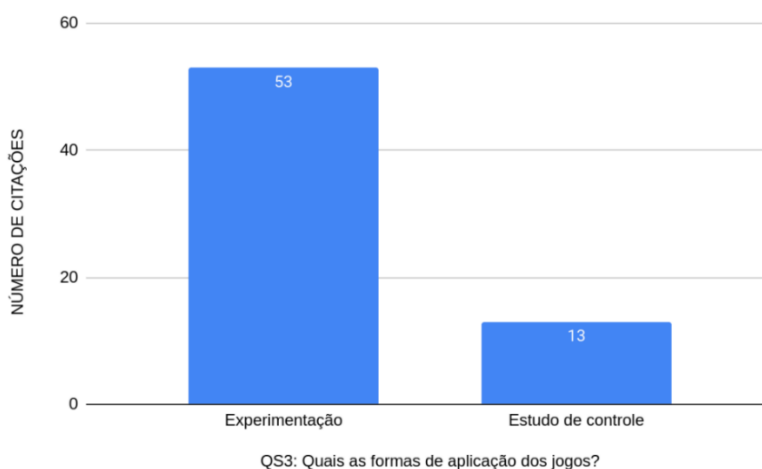


Figura 8 - Forma de aplicação
Fonte: Elaboração própria (2021).

5.1.4. QS4: Qual tipo de conhecimento o jogo visa treinar?

No fim da RSL foram constatados 24 tipos de conhecimentos que os trabalhos visavam treinar dentro do contexto da Engenharia de Software. Os conhecimentos mais ensinados com a implementação de jogos são *soft skills*, gerenciamento de projeto, métodos ágeis, conceitos gerais de engenharia de software e processo de software. O número de artigos citados por conhecimento pode ser visto na Figura 9.

Entre os conhecimentos apresentados em ordem de maior número de citações, é possível identificar (e visualizar seu título completo no apêndice): *Soft Skill* (A1, A4, A6,

A7, A13, A25, A26, A28, A29, A32, A44, A51, A52, A53); Gerenciamento de Projeto (A1, A8, A12, A16, A20, A21, A22, A25, A32, A35, A46, A49, A60); Métodos Ágeis (A5, A6, A11, A12, A13, A29, A32, A50, A52, A54, A55, A59, A61); Principais conceitos de Engenharia Software (A4, A5, A7, A15, A18, A20, A45, A48, A56, A57); Processo de Software (A1, A5, A9, A24, A25, A28, A49, A51); Principais conceitos de Engenharia de Requisitos (A5, A10, A17, A36); Teste de Software (A40, A58, A63, A65); Inspeção de Software (A2, A23, A64); Conceitos de *Global Software Engineering* (A14, A44, A53); Gerenciamento de Risco (A16, A39, A47).

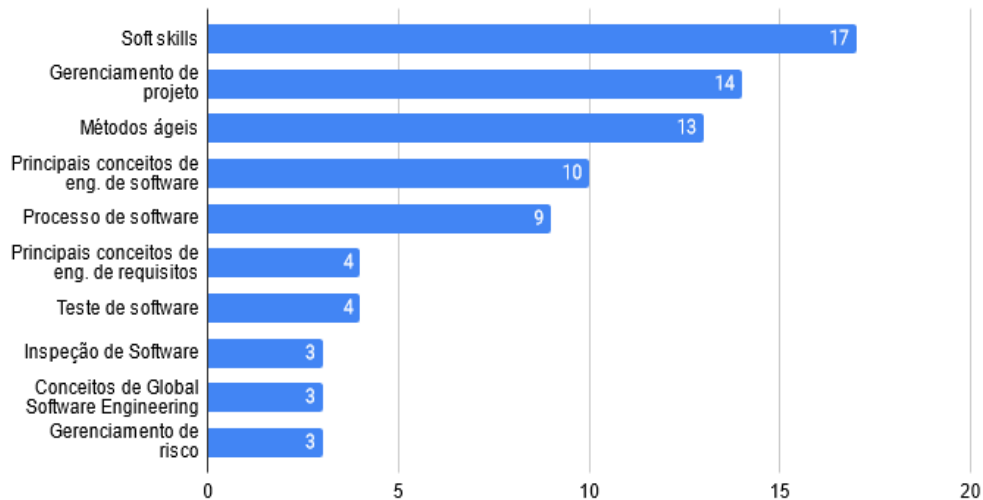


Figura 9 - Competências desenvolvidas

Fonte: Elaboração própria (2021).

5.1.5. QS5: O jogo foi criado com o intuito de ser comercial ou acadêmico?

Nesta questão o ambiente acadêmico tem como foco a educação dos alunos de algum assunto ou matéria, enquanto no ambiente comercial, pelo fato dos funcionários possuírem experiência na área, é trabalhado mais o aperfeiçoamento de conhecimentos e as habilidades usados na sua área de atuação na empresa.

A seguir, a Figura 10 apresenta o percentual de trabalhos de intuito acadêmico ou comercial, dos quais do total de 66 artigos apenas 6 são de caráter comercial.

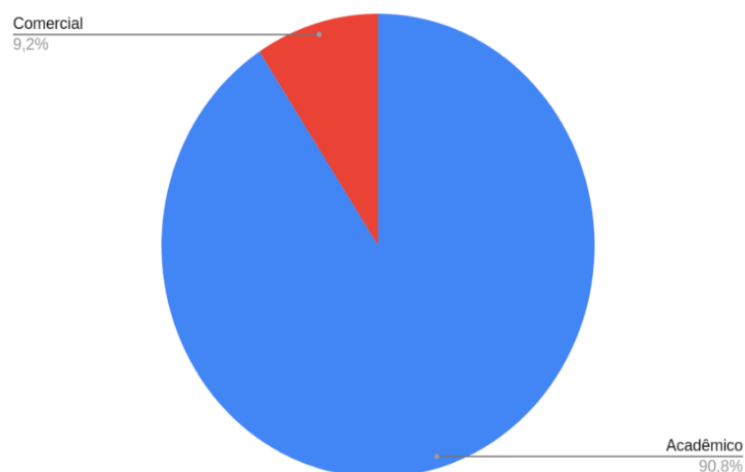


Figura 10 - Contexto de aplicação

Fonte: Elaboração própria (2021).

6. CONCLUSÕES

O presente estudo apresenta a condução de uma revisão da literatura com o intuito de identificar jogos e abordagens gamificadas que são utilizadas no processo de ensino-aprendizagem de Engenharia de Software, buscando identificar desde quais áreas da ES são abordadas até competências que cada jogo ou abordagem visou desenvolver nos participantes.

Por meio da RSL foi possível reconhecer que existe uma variedade de jogos que são utilizados para o ensino de ES, variando desde jogos de tabuleiro até jogos de interpretação de personagens. Dentro das abordagens que fizeram uso de gamificação também é possível observar que um conjunto de elementos são utilizados e dificilmente um elemento é utilizado sozinho, uma vez que um conjunto de elementos gamificados consegue levar a um maior engajamento por parte dos participantes.

Foi possível observar que, dentre os países reconhecidos nesta revisão, o Brasil conseguiu sobressair-se na quantidade de trabalhos produzidos relacionados ao tema desta pesquisa e, dentre as instituições de ensino do Brasil, a Universidade Federal do Ceará foi a que mais realizou pesquisas, com um total de 4 trabalhos identificados.

Os resultados da RSL que foram apresentados neste trabalho servem como alicerce para trabalhos vindouros que visam realizar a criação de jogos ou abordagens que auxiliem no ensino de Engenharia de Software, visto que foram exibidas as principais tendências e formas utilizadas para lidar com o processo de ensino-aprendizagem de ES com o uso de jogos e abordagens gamificadas.

Esta pesquisa tem como intuito continuar evoluindo, de forma que seja feita uma análise mais aprofundada dos resultados obtidos e, posteriormente, seja criada uma plataforma de criação de jogos que possa ser utilizada para ajudar no ensino de Engenharia de Software, aplicando a plataforma em universidades do Brasil e identificando o que pode ser melhorado.

REFERÊNCIAS BIBLIOGRÁFICAS

ALHAMMAD, M. M.; MORENO, A. M. (2018) “What is going on in agile gamification?”. In Proceedings of the 19th International Conference on Agile Software Development: Companion (XP '18). Association for Computing Machinery, New York, NY, USA, Article 36, 1–4. DOI:<https://doi.org/10.1145/3234152.3234161>

CARVER, J.; JACCHERI, L.; MORASCA, S.; SHULL, F. (2003) “Issues in Using Students in Empirical Studies in Software Engineering Education”. Proceedings of the 9th International Software Metrics Symposium, IEEE.

DETERDING S.; KHALED R.; NACKE L.; DIXON D. (2011a) “Gamification: toward a definition”. in CHI 2011, Vancouver, BC, Canada.

DETERDING S.; KHALED R.; NACKE L.; DIXON D. (2011b) “From game design elements to gamefulness: defining gamification”. in Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments. ACM, pp. 9–15.

GHEZZI, C.; MANDRIOLI, D. (2005) "The Challenges of Software Engineering Education". Proceedings of the 27th international conference on Software engineering, p. 637-638, St. Louis, MO, USA, May.

IEEE & ACM JTFCC. (2015) "Software Engineering 2014: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering". In IEEE & ACM; The Joint Task Force on Computing Curricula.

KITCHENHAM, B.; CHARTERS, S. (2007) "Guidelines for performing Systematic Literature Reviews". In *Software Engineering*. (3^o ed., Vol. ||). -2007-01,: EBSE Technical Report.

MALCHER, P. R. C.; FERREIRA, D. A. L.; OLIVEIRA, S. R. B.; VASCONCELOS, A. M. L. (2015) "Um Mapeamento Sistemático sobre Abordagens de Apoio à Rastreabilidade de Requisitos no Contexto de Projetos de Software". *Revista de Sistemas de Informação da FSMA* n. 16, pp. 3-15.

METTLER, T; PINTO, R. (2015) "Serious Games as a Means for Scientific Knowledge Transfer—A Case from Engineering Management Education". *IEEE Transactions on Engineering Management*, vol. 62, no. 2, pp. 256-265, doi: 10.1109/TEM.2015.2413494.

PORTELA, C. S.; VASCONCELOS, A. M. L.; OLIVEIRA S. R. B. (2015) "Análise da Relevância dos Tópicos e da Efetividade das Abordagens para o Ensino de Engenharia de Software". VIII Fórum de Educação em Engenharia de Software, <https://www.researchgate.net/publication/292720684>.

PRIKLADNICKI, R.; ALBUQUERQUE, A. B.; WANGENHEIM, C. G. v.; CABRAL, R. (2009) "Ensino de Engenharia de Software: Desafios, Estratégias de Ensino e Lições Aprendidas". In Proceedings of FEES09 Fórum de Educação em Engenharia de Software. Fortaleza, CE.

SOMMERVILLE, I. (2011) "Engenharia de Software". 9^a edição. Pearson Education do Brasil. ISBN 978-85-7936-108-1.

SOUZA, M. R. D. A.; VEADO, L. F.; MOREIRA, R. T.; FIGUEIREDO, E. M. L.; COSTA, H. A. X. (2017) "Games for learning: bridging game-related education methods to software engineering knowledge areas". 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET), pp. 170-179, doi: 10.1109/ICSE-SEET.2017.17.

ZAMBON, C.; THIRY, M. (2018) "Ludic Practices to Support the Development of Software Engineering Educational Games: A Systematic Review". 2018 XLIV Latin American Computer Conference (CLEI), pp. 794-802, doi: 10.1109/CLEI.2018.00100.

WANGENHEIM, C. G.; SILVA, D. A. (2009) "Qual conhecimento de engenharia de software é importante para um profissional de software?" In: FÓRUM DE EDUCAÇÃO EM ENGENHARIA DE SOFTWARE, 2. Fortaleza. Anais. Fortaleza: UFC.

ZUCARELLI, I; COUTO, L. (2013) “Jogo de tabuleiro em incentivo à alimentação infantil”. Universidade São Judas Tadeu, São Paulo.

ZYDA, M. (2005) “From visual simulation to virtual reality to games”. Computer, 38(9):25– 32.

APÊNDICE - ESTUDOS SELECIONADOS

ID	TÍTULO
A1	Work in progress - a game-based learning system for software engineering education
A2	InspectorX 2.0: Developing a Multi-Device Game for Software Inspection Education
A3	Board Game as a Tool to Teach Software Engineering Concept -- Technical Debt
A4	A case study of software engineering methods education supported by digital game-based learning: Applying the SEMAT Essence kernel in games and course projects
A5	Gamification in Software Engineering Education: An Empirical Study
A6	Enhancing software engineering student team engagement in a high-intensity extreme programming course using gamification
A7	A Board Game to Simulate the Software Development Process Based on the SEMAT Essence Standard
A8	Game Design Techniques for Software Engineering Management Education
A9	Experimental Evaluation of a Serious Game for Teaching Software Process Modeling
A10	Analysis of the effects of the use of Gamification as a teaching strategy in disciplines related to the area of Software Engineering
A11	Relax, It's a Game: Utilising Gamification in Learning Agile Scrum Software Development
A12	SCRUMI: A Board Serious Virtual Game for Teaching the SCRUM Framework
A13	Training scrum with gamification: Lessons learned after two teaching periods
A14	Evaluating the Effectiveness of Board Game Usage to Teach GSE Dynamics
A15	Game Elements in a Software Engineering Study Group: A Case Study
A16	Using Games in Software Engineering Education to Teach Risk Management
A17	Design and Development of a Serious Game for the Teaching of Requirements Elicitation and Analysis
A18	Integrating Serious Games as Learning Resources in a Software Project Management Course: The Case of ProDec
A19	Gamifying the learning of design patterns in software engineering education
A20	Gamification of a Software Engineering course and a detailed analysis of the factors that lead to it's failure
A21	Teaching University Students Kanban with a Collaborative Board Game
A22	Project management game 2D (PMG-2D): A serious game to assist software project managers training
A23	InspectorX: A game for software inspection training and learning
A24	Implementing gamification techniques into university study path - A case study
A25	Gamifying the Escape from the Engineering Method Prison

A26	Can a Team Coordination Game Help Student Software Project Teams?
A27	It was a bit of a race: Gamification of version control
A28	Use of role-play and gamification in a software project course
A29	Engaging and Motivating Developers by Adopting Scrum Utilizing Gamification
A30	Gamifying Software Engineering Tools to Motivate Computer Science Students to Start and Finish Programming Assignments Earlier
A31	On the gamification of human-centric traceability tasks in software testing and coding
A32	SoftBook: Software Development as an Adventure
A33	CityVR: Gameful Software Visualization
A34	Can Learning Formal Specification Be Fun? —Experience and Perspective
A35	What Can Go Wrong in a Software Project? Have Fun Solving it
A36	Gameplay to Introduce and Reinforce Requirements Engineering Practices
A37	Gamifying Collaborative Prioritization: Does Pointsification Work?
A38	Towards a Gamified Support Tool for Requirements Gathering in Bahasa Indonesia
A39	Implementing a Gamified application for a Risk Management course
A40	EMVille: A gamification-based approach to address the equivalent mutant problem
A41	Tool-Supported Collaborative Requirements Prioritisation
A42	CRSG: a serious game for teaching code review
A43	LEARN Board Game: A game for teaching Software Architecture created through Design Science Research
A44	Evaluating GSD-Aware: A Serious Game for Discovering Global Software Development Challenges
A45	Gamification applied for Software Engineering teaching-learning process
A46	A Legacy Game for Project Management in Software Engineering Courses
A47	IsITethical? Board game: playing with speculative ethics of IT innovation in disaster and risk management
A48	Non-conventional dynamics in a software engineering course: practical and ludic activities
A49	Masters of the Process: A Board Game Proposal for Teaching Software Management and Software Development Process
A50	2TScrum': A Board Game to Teach Scrum
A51	ProcSoft: A Board Game to Teach Software Processes Based on ISO/IEC 29110 Standard
A52	Use of Gamification to Teach Agile Values and Collaboration: A multi-week Scrum simulation project in an undergraduate software engineering course
A53	Building LEGO Towers: An Exercise for Teaching the Challenges of Global Work
A54	Evaluating the students' experience with the Scrum Card Game: an experience report in a Software Engineering course
A55	Approaching the Relative Estimation Concept with Planning Poker
A56	Gamification in Remote Teaching of SE Courses: Experience Report
A57	Understanding gamification mechanisms for software development
A58	Is It Worth Using Gamification on Software Testing Education?: An Experience Report
A59	Teaching students scrum using LEGO blocks
A60	Improving Project Manager Decision with Gamification: An Experience Report

A61	SimScrumF: a game for supporting the process of teaching Scrum
A62	Gamification and Evaluation the Use of the Function Points Analysis Technique in Software Quality Subjects: The Experimental Studies
A63	Gamifying a Software Testing Course with Code Defenders
A64	A gamified requirements inspection process for goal models
A65	Does Gamification Improve the Training of Software Testers? A Preliminary Study from the Industry Perspective
A66	The effect of gamification on software architecture knowledge management: a student experiment and focus group study