



UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS  
FACULDADE DE COMPUTAÇÃO  
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

João Luís dos Reis Lima  
Rafael Ferreira de Souza

UM MAPEAMENTO SISTEMÁTICO DA LITERATURA EM MELHORIA DO  
PROCESSO DE SOFTWARE E MEDIÇÃO DE SOFTWARE POR  
PROPOSTAS DE GAMIFICAÇÃO E JOGOS SÉRIOS

Belém  
2018

João Luís dos Reis Lima  
Rafael Ferreira de Souza

UM MAPEAMENTO SISTEMÁTICO DA LITERATURA EM MELHORIA DO  
PROCESSO DE SOFTWARE E MEDIÇÃO DE SOFTWARE POR  
PROPOSTAS DE GAMIFICAÇÃO E JOGOS SÉRIOS

Trabalho de Conclusão de Curso  
apresentado como requisito parcial  
para a obtenção do título de  
Bacharel em Sistemas de  
Informação pela Universidade  
Federal do Pará.

Orientador: Prof. Dr. Sandro Ronaldo  
Bezerra Oliveira.

Coorientador: MSc. Lennon Sales  
Furtado.

Belém  
2018

João Luís dos Reis Lima  
Rafael Ferreira de Souza

UM MAPEAMENTO SISTEMÁTICO DA LITERATURA EM MELHORIA DO  
PROCESSO DE SOFTWARE E MEDIÇÃO DE SOFTWARE POR  
PROPOSTAS DE GAMIFICAÇÃO E JOGOS SÉRIOS

Trabalho de Conclusão de Curso  
apresentado como requisito parcial  
para a obtenção do título de  
Bacharel em Sistemas de  
Informação pela Universidade  
Federal do Pará.

Orientador: Prof. Dr. Sandro Ronaldo  
Bezerra Oliveira.

Coorientador: MSc. Lennon Sales  
Furtado.

Conceito:

BANCA EXAMINADORA:

---

Prof. Dr. Sandro Ronaldo Bezerra Oliveira  
Faculdade de Computação/UFPA - Orientador

---

Prof. Dr. Josivaldo de Souza Araújo  
Faculdade de Computação/UFPA - Membro

---

Prof. Dr. Raimundo Viégas Junior  
Faculdade de Computação/UFPA - Membro

*Dedicamos este trabalho às nossas famílias por estarem sempre presentes e nos ajudando nos momentos mais difíceis e por todo apoio que nos deram para chegarmos até aqui.*

## **AGRADECIMENTO - JOÃO LUÍS LIMA**

Agradeço a Deus pelo dom da vida, por me dar força principalmente nos momentos mais difíceis da minha caminhada, jamais poderia deixar de agradecer à Nossa Senhora de Nazaré, a qual dedico minha devoção, por sempre me ajudar nos maiores momentos de fraqueza. Agradeço à minha mãe, Nazaré Lima, meu maior exemplo de pessoa forte e guerreira, sei que ela nunca mediu esforços para que eu pudesse realizar este sonho, a minha irmã Thelma por estar ao meu lado sempre. À minha parceira de todas as horas, Ghislaine, pelo incentivo e até mesmo pelas cobranças. Elas confiaram em mim e me ajudaram para que pudesse ter a oportunidade de concretizar e encerrar este ciclo importante da minha vida, a compreensão, ajuda e confiança delas fez com que isso fosse possível hoje.

Ao meu amigo e dupla de TCC, Rafael, pelo esforço conjunto e pelo apoio principalmente nos momentos mais complicados vividos durante todas as etapas deste trabalho, para que pudéssemos concluir com êxito mais uma etapa em nossas vidas.

Ao meu coorientador, MSc. Lennon Sales Furtado e ao aluno de Mestrado Roberto Pinheiro, pela ajuda na realização, pois através de esforço conjunto pudemos finalizar este projeto. Ao meu orientador, Prof. Dr. Sandro Ronaldo Oliveira Bezerra, que, com muita paciência e atenção, dedicou o seu valioso tempo para me orientar em cada passo deste trabalho. E aos demais professores que ao longo da graduação contribuíram na minha formação acadêmica e pela influência mesmo que indiretamente na minha futura vida profissional.

Aos meus colegas de turma, e amigos que direta ou indiretamente me apoiaram e viveram comigo vários momentos de parceria, momentos de descontração e ajuda mútua ao longo da graduação, o meu muito obrigado.

## **AGRADECIMENTO – RAFAEL FERREIRA DE SOUZA**

Agradeço a Deus pela vida, saúde e resiliência. Não menos importante, agradeço aos meus pais, Joselina Ferreira de Souza e Raimundo Soares de Souza, os pilares da minha vida. Também ao meu irmão Rodrigo e meu filho Marlon Cauê. Minha família esteve comigo em todos os momentos. Eles confiaram em mim e me ajudaram a concretizar mais este projeto. Sei que eles não mediram esforços para que este sonho se realizasse, sem a compreensão, ajuda e confiança deles nada disso seria possível.

Em especial, dedico este trabalho a Maria Simoni, uma mulher fantástica! Os últimos anos têm sido um misto felicidades e imensas dificuldades, estas jamais teriam sido sequer lutadas sem ela a meu lado. Sua força, confiança e ajuda incondicional me fizeram chegar ao final dessa empreitada. Espero um dia poder retribuí pelo menos um terço do apoio recebido, muito obrigado.

Ao meu amigo e dupla de TCC, João, pelo esforço conjunto e pelo apoio durante todas as etapas deste trabalho fica aqui meu sincero agradecimento.

Agradeço também aos meus colegas de turma que me deram suporte quando precisei. Pessoas que eu conheci, e que jamais esquecerei do que já fizeram por mim. Cito em particular o Robson dos Santos, Samuel Santana, Jones Perote e Luiz Wagner entre outros que não cabe escrever, mas que tenho grande e sincero apreço. Todos me ajudaram de forma indispensável em algum momento do curso, por isso sou grato.

E por último, e claro, não menos importante, agradeço ao meu coorientador, MSc. Lennon Sales Furtado e ao aluno de Mestrado Roberto Pinheiro, pela ajuda e parceria nesta pesquisa. Ao meu orientador, Prof. Dr. Sandro Ronaldo Oliveira Bezerra pela atenção e preocupação. Sempre solicito e extremamente educado em todos os momentos. E aos demais professores que ao longo da graduação contribuíram na minha formação acadêmica e pela influência mesmo que indiretamente na minha futura vida profissional.

## RESUMO

Contexto: o uso de jogos sérios e gamificação em engenharia de software não é tão novo. Contudo, as tecnologias recentes proporcionaram novas oportunidades para sua aplicação tanto no ambiente acadêmico quanto industrial, podendo aumentar o envolvimento e engajamento de alunos e colaboradores. Objetivo: o objetivo deste trabalho é identificar diferentes abordagens no ensino de Medição de Software e Melhoria do Processo de Software por propostas de gamificação e jogos sérios a partir de um estudo de mapeamento sistemático, buscando caracterizar o estado da arte sobre o uso de métodos relacionados à gamificação e a jogos sérios nos temas citados. Método: realizou-se um estudo de mapeamento sistemático para identificar estudos primários e secundários que abordem o uso, a proposta ou a avaliação de gamificação, jogos sérios, seus elementos e mecânicas de jogos em engenharia de software. Resultados: foram identificados 167 estudos primários e secundários, publicados entre 2001 e 2017. Os estudos preliminares mostram uma proximidade entre os métodos utilizados nos trabalhos pesquisados, pois em 92 é usado jogos sérios, enquanto em 71, gamificação é o método explorado. Conclusões: apesar do uso de jogos sérios e gamificação em engenharia de software não ser tão novo, existe um grande espaço a ser explorado, principalmente em Melhoria do Processo de Software e Medição de Software.

Palavras-chave: Engenharia de Software, Melhoria do processo de software, Medição de Software, Gamificação, Jogos sérios, Mapeamento Sistemático da Literatura.

## ABSTRACT

Context: the use of serious games and gamification in software engineering is not so recent. However, recent technologies have provided new opportunities for their application in both the academic and industrial settings, and may increase the engagement of students and staff. Objective: the objective of this work is to identify different approaches in the teaching of Software Measurement and Software Process Improvement by gamification proposals and serious games through a systematic mapping study. Seeking to characterize the state of the art on the use of methods related to gamification and serious games in the mentioned subjects. Method: we performed a systematic mapping study to identify primary and secondary studies that address the use, proposal or evaluation of gamification, serious games, their elements and game mechanics in software engineering. Results: we identified 167 primary and secondary studies, published between 2001 and 2017. Preliminary studies show a close relationship between the methods used in the researched jobs, since in 92 serious games are used, whereas in 71, gamification is the method explored. Conclusions: although the use of serious games and gamification in software engineering is not recent, there is a large space to be explored, especially in Software Process Improvement and Software Measurement.

**Keywords:** Software Engineering, Software Process Improvement, Software Measurement, Gamification, Serious Games, Systematic Literature Mapping.

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>15</b>
1.1	VISÃO GERAL.....	15
1.2	JUSTIFICATIVA.....	16
1.3	MOTIVAÇÃO.....	18
1.4	OBJETIVO GERAL.....	18
1.4.1	Objetivos Específicos.....	18
1.5	METODOLOGIA.....	19
1.6	ESTRUTURA DO TRABALHO.....	20
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>21</b>
2.1	MELHORIA DO PROCESSO DE SOFTWARE.....	22
2.2	MEDIÇÃO DE SOFTWARE.....	23
2.3	ENGENHARIA DE SOFTWARE BASEADA EM EVIDÊNCIAS.....	24
2.4	REVISÃO SISTEMÁTICA DA LITERATURA.....	25
2.5	REVISÃO INFORMAL X REVISÃO SISTEMÁTICA.....	26
2.6	METODOLOGIA DA REVISÃO SISTEMÁTICA.....	27
2.6.1	Planejamento da Revisão.....	28
2.6.2	Condução da Revisão.....	29
2.6.3	Apresentação da Revisão.....	31
2.7	CONSIDERAÇÕES DO CAPÍTULO.....	31

<b>3</b>	<b>PLANEJAMENTO DO MAPEAMENTO.....</b>	<b>31</b>
<b>3.1</b>	<b>PROTOCOLO DE REVISÃO.....</b>	<b>33</b>
3.1.1	Objetivos do Mapeamento.....	33
3.1.2	Questões de Pesquisa Principais.....	34
3.1.3	Questões de Pesquisa Secundárias.....	36
3.1.4	Escopo da Pesquisa.....	39
3.1.5	Seleção de Fontes.....	38
3.1.6	Métodos de Busca Primária.....	39
3.1.6.1	Identificação de Palavras-Chaves e Sinônimos.....	39
3.1.6.2	Geração de Strings de Busca.....	41
3.1.7	Seleção dos Estudos Primários.....	44
3.1.7.1	Critérios de Inclusão e Exclusão dos Estudos Primários.....	44
3.1.7.2	Processo de Seleção dos Estudos Primários.....	45
3.1.8.	Avaliação dos Estudos Primários.....	47
3.1.8.1	Critérios de Qualidade dos Estudos Primários.....	48
3.1.8.2	Processo de Avaliação dos Estudos Primários.....	49
<b>3.2</b>	<b>CONSIDERAÇÕES DO CAPÍTULO.....</b>	<b>54</b>
<b>4</b>	<b>RESULTADOS E ANÁLISE DO MAPEAMENTO</b>	
	<b>SISTEMÁTICO.....</b>	<b>56</b>
<b>4.1</b>	<b>CONDUÇÃO DO MAPEAMENTO.....</b>	<b>55</b>
4.1.1	Busca Primária.....	55

4.1.2 Seleção de Estudos Primários.....	56
4.1.3 Avaliação dos Estudos Primários.....	66
<b>4.2 CONSIDERAÇÕES DO CAPÍTULO.....</b>	<b>67</b>
<b>5 CONCLUSÕES.....</b>	<b>68</b>
<b>5.1 VISÃO GERAL.....</b>	<b>68</b>
<b>5.2 RESULTADOS OBTIDOS.....</b>	<b>69</b>
<b>5.3 PONTOS FORTES E OPORTUNIDADES DE MELHORIA..</b>	<b>69</b>
<b>5.4 TRABALHOS FUTUROS.....</b>	<b>71</b>
<b>REFERÊNCIAS.....</b>	<b>72</b>
<b>APÊNDICE A – PROTOCOLO DE REVISÃO.....</b>	<b>74</b>
<b>APÊNDICE B – ESTUDOS PRIMÁRIOS E SECUNDÁRIOS.....</b>	<b>90</b>
<b>APÊNDICE C – ESTUDOS EXCLUÍDOS.....</b>	<b>98</b>

## LISTA DE GRÁFICOS

Gráfico 1 – Número de Estudos retornados a partir da Busca Primária.....	56
Gráfico 2 – Número de Estudos ao Longo dos Anos.....	59
Gráfico 3 – Número de Estudos por Autor.....	59
Gráfico 4 – Número de Estudos por Instituição.....	60
Gráfico 5 – Número de Estudos por País.....	61
Gráfico 6 – Distribuição de Estudos por Tipo de Publicação.....	62
Gráfico 7 – Principais Workshops.....	62
Gráfico 8 – Principais Conferências.....	63
Gráfico 9 – Principais Periódicos.....	63
Gráfico 10 – Distribuição dos Trabalhos por Tipo de Estudo.....	64
Gráfico 11 – Métodos Empregados nos Estudos do Tipo Experimental.....	65
Gráfico 12 – Gamificação vs Jogo Séri.....	66

## LISTA DE QUADROS

Quadro 1 – Seleção dos Estudos Primários.....	58
---	----

## LISTA DE TABELAS

Tabela 1 – Escala por Critério.....	50
Tabela 2 – Níveis de Qualidade.....	53
Tabela 3 – Qualidade dos Estudos Primários.....	66

## LISTA DE ABREVIATURAS E SIGLAS

SPI	<i>Software Process Improvement</i>
CMMI	<i>Capability Maturity Model Integration</i>
ESBE	Engenharia de Software Baseada em Evidências
MR-MPS-BR	Modelo de Referência para Melhoria de Processo de Software
MSL	Mapeamento Sistemático da Literatura
RSL	Revisão Sistemática da Literatura
SPIDER	<i>Software Process Improvement: Development and Research</i>

**[1] Comentário:** Adicionei o espaço entre a sigla e o texto.

## 1. INTRODUÇÃO

Esse capítulo apresenta a introdução deste trabalho. Está dividido nas seguintes seções: visão geral, justificativa, motivação, metodologia, objetivos do trabalho e sua estrutura.

### 1.1 VISÃO GERAL

A engenharia de software não se relaciona com a criação de documentos. Refere-se à geração de qualidade. Melhor qualidade leva à redução de retrabalho, e menor retrabalho resulta em tempo de entrega mais rápido (PRESSMAN, 2006), ou seja, a engenharia de software tem por objetivo garantir a qualidade dos processos que envolvem o desenvolvimento de software. Assim, entende-se que qualidade dos processos está intimamente ligada à melhoria do processo de software (SPI - *Software Process Improvement*), e sabe-se que um elemento-chave de qualquer processo de engenharia é a medição.

Neste contexto, o termo melhoria do processo de software implica em muitos cenários. Primeiro que os elementos de um processo de software eficaz podem ser definidos de maneira eficaz. Segundo, que uma abordagem organizacional existente para o desenvolvimento de software pode ser avaliada em relação aos elementos. E terceiro, que uma estratégia significativa pode ser definida para a melhoria. A abordagem SPI transforma a abordagem existente para o desenvolvimento de software em algo mais focado, com melhor repetibilidade e mais confiável, em termos de qualidade de produto e prazo de entrega (PRESSMAN, 2016).

De acordo com PRESSMAN (2016), a medição permite entender melhor o processo e o projeto, pois é um mecanismo para a avaliação objetiva. Medições podem ser aplicadas ao processo de software com a intenção de melhorá-lo continuamente. Elas podem ser usadas durante um projeto de software para ajudar nas estimativas, no controle de qualidade, na produtividade e no controle do projeto. Também, podendo ser usadas para avaliar a qualidade dos artefatos e auxiliar na tomada de decisões táticas à medida que o projeto avança.

A ênfase para buscar melhorias da qualidade deve ser concentrada em melhoramentos contínuos, atitudes que, promovidas continuamente, permitam reconhecer os problemas, priorizar ações corretivas e implantá-las (SILVA, 1999).

Adicionalmente, a gamificação vem se tornando um elemento cada vez mais popular em diversos ambientes. A utilização da mesma pode ser vista em empresas (WITT *et al.*, 2011), na educação (DOMÍNGUEZ *et al.*, 2013; GIBSON *et al.*, 2013), em carros ecológicos (MUNTEAN, 2011), entre outros (CLEMENTI, 2014). Podem ser usadas para impulsionar o trabalho em equipe, aumentando o senso de colaboração e produtividade. Assim, a gamificação apresenta-se como um fenômeno emergente com muitas potencialidades de aplicação em diversos campos da atividade humana.

Segundo KAPP (2012), gamificação é uma aplicação apropriada de três elementos dos jogos: mecânica, estética e conceito; com o objetivo de proporcionar o engajamento entre as pessoas, motivar ações, encorajar a aprendizagem e promover a resolução de problemas. O principal objetivo é aumentar o engajamento e despertar a curiosidade dos usuários e, além dos desafios propostos nos jogos, na gamificação as recompensas também são itens cruciais para o sucesso. A gamificação é, basicamente, usar ideias e mecanismos de jogos para incentivar alguém a fazer algo.

De acordo com ZYDA (2005), jogo sério de computador é um desafio mental com regras específicas para capacitar e/ou educar de forma lúdica. Para DEMPSEY *et al.* (1996), jogos sérios são aqueles projetados para ensinar sobre determinados assuntos, expandir conceitos, ou ajudar a exercitar ou aprender uma habilidade ou mudar atitudes.

## 1.2 JUSTIFICATIVA

Com o objetivo de aplicar técnicas de um Mapeamento Sistemático da Literatura e, assim, fazer uma análise crítica da literatura, no que diz respeito à Gamificação e Jogos Sérios em SPI e Medição de Software, **teve-se a necessidade em se** realizar uma RSL - Revisão Sistemática da Literatura para descobrir abordagens (técnicas, modelos de processos, *frameworks* de processo, ferramenta, metodologias e afins) que pudessem auxiliar esta área.

[2] Comentário: modificação feita pelo prof. Josivaldo

Vale ressaltar que este trabalho foi desenvolvido como uma das etapas de uma tese de doutorado, ainda em desenvolvimento, vinculado ao projeto SPIDER – *Software Process Improvement: Development and Research*, com o objetivo de reunir conhecimentos a respeito da implementação e avaliação de propostas de jogos sérios e de gamificação para o ensino do processo de medição e processos relacionados com programas de melhoria de processos de software. A partir disso, criar uma ferramenta educacional para o ensino do processo de medição que faça uso de boas práticas presentes nos trabalhos pesquisados. O escopo deste trabalho não contempla todas as etapas de uma Revisão Sistemática da Literatura, já que neste trabalho será apenas apresentada a Revisão Sistemática da Literatura até a etapa de avaliação de qualidade dos Estudos Primários, tendo como principal resultado a caracterização dos estudos relacionados à Gamificação e Jogos Sérios em SPI e Medição de Software disponíveis na literatura.

Este trabalho é justificado pelo fato de que programas de melhoria de processo de software, assim como, processo de medição, são de grande importância para organizações desenvolvedoras de software, e suas gestões, ocupam papel relevante em modelos de melhoria de processo de software, como o MR-MPS-SW (Modelo de Melhoria do Processo de Software Brasileiro) (SOFTEX, 2016) e o CMMI (*Capability Maturity Model Integration*) (SEI, 2010). Além disso, os resultados de uma Revisão Sistemática da Literatura para Gamificação e Jogos Sérios em SPI e Medição no contexto de projetos de software poderá contribuir tanto para o meio acadêmico quanto para o industrial, apresentando um conjunto de estudos primários já avaliados quanto à relevância de seu conteúdo e também quanto à sua qualidade.

KITCHENHAM (2007) afirma, também, que os resultados de uma revisão sistemática ajudam a identificar alguma lacuna na pesquisa atual para sugerir áreas para investigações futuras, examinar o quanto evidências experimentais apoiam ou contradizem hipóteses teóricas e apoiar a geração de novas hipóteses. Com este intuito, essa pesquisa visa, por meio de um método confiável, rigoroso e auditável, encontrar na literatura abordagens que apoiem o processo de Gamificação e Jogos Sérios em SPI e Medição de Software,

além de descobrir como este assunto tem sido abordado nos últimos dezessete anos na literatura especializada.

### **1.3 MOTIVAÇÃO**

Daqui a alguns anos, uma boa parcela das nossas atividades diárias terão elementos de games. Isto significa, na prática, que quase tudo o que seja feito terá algum elemento que está presente nos games: definição de objetivos (*goals*), desafios (*challenges*) e recompensas (*rewards*), que funcionam a partir de mecânicas como personalização (avatares), pontuações (scores), classificações (*leaderboards*) e *rankings*. Entretanto, algumas informações também são preocupantes, o que faz necessário um trabalho na área. Neste contexto, a Engenharia de Software Baseada em Evidências busca estudos na literatura para apoiar uma hipótese ou responder a questões de pesquisa, o que pode ser realizado por meio, principalmente, de Revisões Sistemáticas da Literatura.

Além disso, os resultados de uma Revisão Sistemática da Literatura para a Gamificação e Jogos Sérios em SPI e Medição de Software poderão contribuir tanto para o meio acadêmico quanto para o industrial. Esta Revisão poderá ser usada em diferentes ambientes e para diferentes fins, seja para apresentação e aprendizado sobre a Gamificação e Jogos Sérios no meio acadêmico ou para auxiliar uma empresa que queira realizar este processo em seus softwares em casos de desenvolvimento prático.

### **1.4 OBJETIVO GERAL**

O objetivo geral deste trabalho é realizar um Mapeamento Sistemático da Literatura para a área de Gamificação e Jogos Sérios em Melhoria do Processo de Software e Medição de Software.

#### **1.4.1 Objetivos Específicos**

Definir previamente especificações, tais como, restrições de pesquisas, critérios de inclusão e exclusão de Estudos Primários, critérios de qualidade para esses estudos, dentre outras, que nortearão a realização do Mapeamento Sistemático da Literatura.

Conduzir o Mapeamento Sistemático da Literatura seguindo as especificações estabelecidas previamente.

Analisar os resultados da revisão por meio da caracterização dos estudos selecionados.

[3] Comentário: Acrescentei os objetivos específicos

## 1.5 METODOLOGIA

O método a ser utilizado na pesquisa foi o método dedutivo, pois partiu de princípios gerais sobre Gamificação e Jogos Sérios em SPI e Medição para serem aplicados a casos específicos.

A pesquisa realizada neste trabalho pode ser classificada quanto a seus objetivos como descritiva e quanto a seus procedimentos como bibliográfica e experimental. Isto porque a pesquisa, em mãos, deve servir como procedimento básico para o estudo, pelos quais se busca o domínio do “estado da arte” sobre Gamificação e Jogos Sérios em SPI e Medição de Software por meio de um Mapeamento Sistemático da Literatura, confirmando sua classificação em relação a seus procedimentos como bibliográfica e experimental.

É considerada quanto a seus objetivos como descritiva por procurar descobrir, com a precisão possível, a frequência com que as abordagens (técnicas, modelos de processos, *frameworks* de processo, ferramenta, metodologias e afins) ocorrem, suas relações e conexões, suas naturezas e características.

Quanto à técnica, este trabalho fez a opção pelo Mapeamento Sistemático da Literatura. Esta opção justifica-se porque o método escolhido permite verificar na literatura, de uma forma confiável, criteriosa e rigorosa, a produção científica sobre Gamificação e Jogos Sérios em SPI e Medição de Software.

A pesquisa utilizou-se de 6 fontes de dados internacionais previamente escolhidas e expostas no protocolo de revisão, que será apresentado no Capítulo 2. A pesquisa nas fontes escolhidas foi por meio de *strings* de busca, que foram utilizadas em cada fonte. Os estudos primários

foram selecionados e guardados na ferramenta JabRef, pois esta ferramenta permite o gerenciamento de referência o que será de grande ajuda devido à quantidade de estudos manipulados.

Os estudos retornados passaram por uma seleção, por meio de critérios de inclusão e exclusão. Após a seleção, os estudos foram novamente avaliados, utilizando os critérios de qualidade que também foram estabelecidos no protocolo de revisão sistemática. Após essa avaliação, foram extraídos resultados dos estudos aprovados.

## **1.6 ESTRUTURA DO TRABALHO**

Este trabalho está organizado em cinco capítulos, da seguinte forma:

- **CAPÍTULO 1** – descreve a motivação, problema de pesquisa, objetivos, metodologia da pesquisa, bem como a justificativa e organização do trabalho;
- **CAPÍTULO 2** – apresenta a fundamentação teórica do trabalho, no que diz respeito à Gamificação e Jogos Sérios em SPI, Medição de Software e à Revisão Sistemática da Literatura;
- **CAPÍTULO 3** – apresenta o planejamento da revisão;
- **CAPÍTULO 4** – apresenta a principal contribuição desse trabalho, a execução da revisão e seus resultados;
- **CAPÍTULO 5** – apresenta as conclusões do trabalho e trabalhos futuros.

## 2. FUNDAMENTAÇÃO TEÓRICA

Segundo DAVIS *apud* (GLASS, 2002), a indústria de software encontra-se no mesmo estado da arte que a indústria farmacêutica esteve no século XIX. Para MAFRA e TRAVASSOS (2006) a ideia de DAVIS, ampara-se no fato de que frequentemente a indústria de software é acometida por diversos anúncios de “cura” para os mais variados problemas, assim como foi à indústria farmacêutica de dois séculos atrás. Com isto, surgem dúvidas de qual solução deve-se escolher, e onde buscar fundamentação para auxiliar esta escolha, bem como, saber a precedência desta solução ou “cura” anunciada.

MAFRA e TRAVASSOS (2006) afirmam que as respostas para estas questões poderiam ser obtidas de forma razoável caso a Engenharia de Software fizesse um uso intenso e sistemático de uma abordagem baseada em evidências. Neste cenário, verifica-se a importância de se trabalhar com a Engenharia de Software Baseada em Evidências (ESBE).

A ciência é uma atividade cooperativa e social, e o conhecimento científico é resultado do processo cumulativo desta cooperação (BIOLCHINI, 2005). Com isso em mente, MAFRA e TRAVASSOS (2006) afirmam que a revisão na literatura é o meio pelo qual o pesquisador pode identificar o conhecimento específico, em uma determinada área, de forma a planejar sua pesquisa, evitando duplicação de esforços e a repetição de erros passados.

Porém, uma revisão na literatura deve ser conduzida de forma confiável e abrangente, através de um protocolo pré-estabelecido para garantir que esta revisão tenha de fato valor científico e com possibilidade de repetição, pois caso isto não aconteça, as revisões tornam-se informais e dependentes dos revisores que a conduziram, diminuindo o grau de confiabilidade das mesmas.

Neste contexto MAFRA e TRAVASSOS (2006) afirmam que o desenvolvimento de uma abordagem sistemática de revisão visa estabelecer um processo formal para conduzir este tipo de investigação, evitando a inclusão de eventuais vieses da revisão de literatura informal. Com isso obtendo uma revisão confiável, rigorosa e auditável.

## **2.1 MELHORIA DO PROCESSO DE SOFTWARE**

Processos devem ser tecnicamente corretos e devem ser capazes de atender às necessidades do negócio. Entretanto, processos podem estar corretos do ponto de vista da Engenharia de Software e não serem competitivos. Podem consumir demasiado tempo e esforço ou não produzirem produtos com a qualidade necessária para satisfazer as necessidades de seus usuários. Processos podem apresentar problemas e devem ser objeto, continuamente, de melhorias. É importante, neste contexto, dispor-se de mecanismos capazes de evidenciar problemas nos processos e apoiar na identificação de objetivos de melhoria (CAVALCANTI *et al.*, 2012).

Um objetivo de melhoria de processo é um conjunto de características desejadas, definidas para orientar o esforço de melhoria de processos de modo específico e mensurável (SEI, 2010). Estes objetivos devem agregar valor ao negócio da organização e melhorar a qualidade dos produtos desenvolvidos (BARRETO, 2011).

A melhoria de processos de software pode estar relacionada a: (i) galgar níveis mais altos de maturidade (melhoria vertical); e/ou (ii) realizar mudanças visando a uma maior adequação às necessidades da organização ou melhorias no desempenho dos processos (melhoria horizontal). Nos dois casos, a melhoria dos processos deve estar associada a objetivos de melhoria que visam principalmente: (i) entender as características dos processos existentes e os fatores que afetam o seu desempenho; (ii) planejar e implementar ações que modifiquem o processo para atender melhor às necessidades de negócio; e (iii) avaliar os impactos e os benefícios obtidos com as mudanças realizadas nos processos (ALBUQUERQUE, 2008; FLORAC e CARLETON, 1999).

## **2.2 MEDIÇÃO DE SOFTWARE**

Medição de software é uma avaliação quantitativa de qualquer aspecto dos processos e produtos de software, que permite seu melhor entendimento e, com isso, auxilia o planejamento, controle e melhoria do que se produz e de como é produzido (BASS *et al.*, 1999).

Medir é essencial na Engenharia de Software, pois é através da medição que se pode entender e controlar processos, produtos e projetos. Medidas fornecem informações sobre objetos (processos, produtos e projetos) e eventos, por exemplo, a fase de testes de um projeto, tornando-se compreensíveis e controláveis (FENTON e PFLEEGER, 1997). Com estas medidas é possível conhecer a qualidade de um produto, assim como, a estabilidade e capacidade de um processo ou estágio atual de um projeto, e com estes dados em mãos, controlar, tomar decisões e melhorar a qualidade de processos e produtos.

Medir software, entretanto, custa caro e envolve um esforço considerável em termos de recursos humanos. E, mais grave, as medidas nem sempre são úteis para o conhecimento, o controle e a tomada de decisão. Muitas vezes, mede-se simplesmente por medir ou para atender aos requisitos de um modelo de maturidade de processos. Nestes casos, não se atingem os objetivos de medição de software, nem se obtêm os benefícios que as medidas podem oferecer (CAVALCANTI *et al.*, 2012).

Medições de software, para serem efetivas, precisam estar alinhadas às necessidades de negócio da organização, ou seja, aos seus objetivos estratégicos, e estarem direcionadas às necessidades de informação de gerentes de projetos e engenheiros de software. Estas necessidades devem ser explicitadas e devem orientar a definição do que medir e de como analisar e comunicar o resultado das medidas (MCGARRY *et al.*, 2002; ISO/IEC, 2002; BASILI *et al.*, 1994; BARRETO, 2011).

### **2.3 ENGENHARIA DE SOFTWARE BASEADA EM EVIDÊNCIAS**

Segundo KITCHENHAM (2004), a Engenharia de Software Baseada em Evidências é aquela que busca prover meios pelos quais melhores evidências provenientes da pesquisa possam ser integradas com experiência prática e valores humanos no processo de tomada de decisão, considerando o desenvolvimento e a manutenção do software. Com esta afirmação, é definido um caráter experimental para a Engenharia de Software que seria o uso da

abordagem científica para desenvolvimento, evolução e manutenção de software.

KITCHENHAM *et al.* (2004) afirmam, também, que outras áreas que adotaram a abordagem baseada em evidências avançaram consideravelmente, a exemplo da medicina, que nas duas últimas décadas aumentou o número de publicações utilizando este tipo de abordagem, além do surgimento de revistas da área especializadas no assunto em questão.

COSTA (2010), sobre abordagem baseada em evidências, afirma que, com o aumento da importância do software em diversas áreas nos dias atuais, como aparelhos celulares e sistemas de controle de voo, é preciso tomar mais cuidado com os métodos de pesquisa, também na engenharia de software.

A essência do paradigma baseado em evidência é coletar e analisar sistematicamente todos os dados disponíveis sobre determinado fenômeno para obter uma perspectiva mais completa e mais ampla do que se pode captar por meio de um estudo individual (COSTA, 2010).

Segundo MAFRA e TRAVASSOS (2006), para atingir um nível adequado de evidência a respeito da caracterização de uma determinada tecnologia em uso, a Engenharia de Software Baseada em Evidência deve fazer uso basicamente de dois tipos de estudos:

- Estudos Primários, que são os estudos que visam caracterizar uma determinada tecnologia em uso dentro de um contexto específico. Onde se encontram os *surveys* e os estudos de caso;
- Estudos Secundários, que são os estudos que visam identificar, avaliar e interpretar todos os resultados relevantes a um determinado tópico de pesquisa, fenômeno de interesse ou questão de pesquisa. Onde se encontram as Revisões Sistemáticas.

Um dos principais métodos da Engenharia de Software Baseada em Evidências são as Revisões Sistemáticas da Literatura (RSL), classificadas como Estudos Secundários, já que, dependem dos Estudos Primários

utilizados para revelar evidências e construir conhecimento (DYBÅ *et al.*, 2007; OATES e CAPPER, 2009; TRAVASSOS, 2007 *apud* COSTA, 2010).

## 2.4 REVISÃO SISTEMÁTICA DA LITERATURA

A Revisão Sistemática da Literatura (RSL) consiste em uma metodologia de pesquisa a qual objetiva identificar, avaliar e interpretar a maior quantidade possível de estudos relevantes e disponíveis para uma particular questão de pesquisa, tópico de pesquisa, ou fenômeno de interesse (KITCHENHAM, 2007). Revisões sistemáticas avaliam evidências de forma sistemática/metódica e transparente, a partir da aplicação de métodos explícitos de busca, apreciação crítica e síntese da informação selecionada (KITCHENHAM, 2007 *apud* COSTA, 2010). Para TRAVASSOS (2007) *apud* COSTA (2010), RSL “proveem meios para executar revisões da literatura abrangentes e não tendenciosas, fazendo com que seus resultados tenham valor científico”.

A principal meta de uma RSL consiste em realizar pesquisa exaustiva na literatura, em busca de evidências que possam apoiar uma determinada hipótese, ou simplesmente a busca por conhecimento aprofundado acerca de certo fenômeno de interesse. Para tal, a revisão sistemática faz uso de estudos previamente publicados e validados pertinentes ao tópico de interesse: os estudos primários, estudos de natureza experimental que envolvem hipóteses e resultados obtidos com pesquisas e experimentação, a partir de diferentes métodos, como *surveys*, estudo de caso e experimentos. Na Engenharia de Software, os estudos experimentais consistem em uma importante maneira de se obter maiores informações a respeito de novas tecnologias, metodologias e boas práticas quanto ao desenvolvimento de software (MAFRA e TRAVASSOS, 2006).

A revisão sistemática consiste em um estudo secundário, por utilizar como base, estudos primários previamente publicados. Assim, pode ser feita a integração de diversos estudos experimentais, de forma a comparar seus resultados, visto que, nenhum estudo individualmente pode ser considerado

definitivo (MAFRA e TRAVASSOS, 2006), sendo necessária a confirmação de resultados obtidos a partir da análise de um número maior de estudos.

Importante ressaltar que, para este trabalho, não foi realizada todas as etapas da Revisão Sistemática da Literatura. Foi elaborado um Mapeamento Sistemático da Literatura, que abrange a revisão dos estudos em busca de objetivos definidos no Protocolo de Revisão. Após a busca, gera-se os Estudos Primários, como dito anteriormente, os quais são analisados superficialmente e alguns são selecionados como estudos secundários, para uma análise mais aprofundada. Com o término da análise, são gerados os resultados requeridos pelo Protocolo de Revisão. Após isso é feita a Avaliação dos Estudos Primários, finalizando assim, o processo de Mapeamento Sistemático da Literatura (foco deste trabalho). Os passos subsequentes da Revisão Sistemática da Literatura dependem dos resultados obtidos do MSL, fazendo deste um passo fundamental para o produto final da RSL.

## **2.5 REVISÃO INFORMAL X REVISÃO SISTEMÁTICA**

Uma revisão sistemática difere-se de uma revisão informal nos seguintes aspectos:

- Inconsistência dos resultados. Para BIOLCHINI *et al.* (2005), nas revisões informais, a presença de inconsistência é um fator negativo, já que dificulta o entendimento e julgamento do pesquisador. Já nas revisões sistemáticas, esta inconsistência é um fator de incentivo para o pesquisador, já que por meio desta inconsistência pode ser percebida a falta de robustez para o fenômeno em estudo, o que pode auxiliar no apoio ou não de suas hipóteses;
- Propósito. MAFRA e TRAVASSOS (2006) afirmam que uma revisão sistemática não é simplesmente uma revisão da literatura conduzida conforme um planejamento. A revisão de literatura é parte integrante de uma revisão sistemática, ou seja, o objetivo de uma revisão sistemática é maior do que o de uma revisão de literatura informal, tendo em vista que o propósito de uma revisão sistemática é a análise dos dados coletados com o objetivo da geração de evidências para o fenômeno em questão.

No contexto da Engenharia de Software, MAFRA e TRAVASSOS (2006) afirmam que frequentemente as revisões de literatura são realizadas de forma informal, sem planejamento e critérios estabelecidos, o que geralmente ocasiona em uma revisão pouco abrangente, não passível de repetição, pouco confiável e dependente de revisores. Aspectos estes que não são encontrados em uma revisão sistemática, já que a mesma estabelece um processo formal.

Vale ressaltar, ainda, neste aspecto, que uma revisão sistemática não é apenas um agrupamento de estudos e sim uma abordagem metodológica para se realizar pesquisa com finalidades experimentais.

## **2.6 METODOLOGIA DA REVISÃO SISTEMÁTICA**

Uma RSL deve ser conduzida por meio de um protocolo, criado anteriormente, assim, garantindo de fato, o valor científico e possibilidade de repetição para a revisão. Sem estes processos as revisões tornam-se informais e dependentes dos revisores que as conduziram e com isso diminui o grau de confiabilidade das mesmas.

A revisão sistemática segue uma metodologia específica, apresentada por KITCHENHAM (2007), em que define: revisões sistemáticas começam pela definição de um Protocolo de Revisão, que especifica a questão de pesquisa a ser tratada e os métodos que serão usados para sua realização; que são baseadas em uma estratégia de busca definida pelo pesquisador, a qual objetiva identificar o máximo possível de estudos relevantes à questão de pesquisa; onde a estratégia de busca é documentada nos mínimos detalhes, para que o leitor possa avaliar seu rigor, completude e replicabilidade; exigem critérios explícitos de inclusão e exclusão de Estudos Primários, de forma a avaliar a necessidade destes estudos para a pesquisa; especificam a informação a ser obtida a partir de cada Estudo Primário, incluindo critérios de qualidade para avaliação de cada estudo; além de ser um pré-requisito para uma meta-análise quantitativa de dados.

Desta forma, KITCHENHAM (2007) resume as etapas de uma revisão sistemática em três fases principais: planejamento, condução e apresentação. Estas etapas são explicadas nas próximas subseções.

### 2.6.1 Planejamento da Revisão

Durante o Planejamento da revisão sistemática, o passo mais importante a ser tomado consiste na elaboração de um Protocolo de Revisão objetivo, metódico e replicável. O Protocolo de Revisão é o documento mais valioso de uma RSL, por conter toda a metodologia que será seguida pelo pesquisador durante a Condução de sua pesquisa. É a partir deste documento que a revisão sistemática torna-se replicável, reproduzível e passível de novas mudanças; este documento também define o rigor da pesquisa em si, incluindo critérios de inclusão, exclusão e qualidade dos estudos primários a serem considerados na pesquisa.

O Protocolo de Revisão define toda a fase referente à Condução da Pesquisa e Apresentação de Resultados, identificando de que maneira os estudos serão obtidos, quais serão as fontes de busca, quais serão os critérios de inclusão de estudos, quais motivos levarão à exclusão de potenciais estudos da pesquisa, o que será procurado nos estudos selecionados, como estes dados serão analisados e sintetizados, como a qualidade dos estudos será avaliada. Enfim, toda a pesquisa será baseada nas informações existentes no Protocolo de Revisão, o que torna este o momento crucial da pesquisa, o qual irá definir seu rigor, valor científico e aderência às características mais nobres de precisão e confiabilidade da RSL.

Para KITCHENHAM (2007), antes de realizar uma revisão sistemática, os pesquisadores devem assegurar que a mesma é necessária e o protocolo deve ser capaz de responder a algumas questões:

- Quais são os objetivos desta revisão?
- Que fontes foram pesquisadas para identificar os Estudos Primários? Houve alguma restrição?
- Quais foram os critérios de inclusão / exclusão e como foram aplicados?
- Que critérios foram utilizados para avaliar a qualidade dos Estudos Primários?
- Como foram aplicados os critérios de qualidade?
- Como os dados foram extraídos dos Estudos Primários?

- Como os dados foram sintetizados?
- Quais foram as diferenças entre os estudos pesquisados?
- Como os dados foram combinados?

TRAVASSOS (2007), por meio de alguns passos, confirma e acrescenta novas informações sobre o que a fase de planejamento deve contemplar:

- Objetivos da pesquisa devem ser listados;
- Questões de pesquisa formuladas (*Strings* de busca preparadas);
- Métodos que serão utilizados para executar a revisão e analisar os dados obtidos devem ser definidos;
- As fontes e seleção de estudos devem ser planejadas;
- Um Protocolo de Revisão deve ser definido, documentado e disponibilizado.

Fator crucial a ser levado em consideração, é que depois de definido, o protocolo deve passar por uma avaliação para garantir a viabilidade da pesquisa. É sugerido que essa avaliação seja realizada por um especialista.

#### 2.6.2 Condução da Revisão

Uma vez que o Protocolo de Revisão tenha sido elaborado e aprovado, a condução da pesquisa pode ter início. É importante, como notado por KITCHENHAM (2007), que os pesquisadores, ao construírem o Protocolo, levem em consideração as etapas existentes na condução da revisão, de modo que o Protocolo possa estar adequado à realidade da pesquisa. Deste modo, não devem ser planejadas atividades que não possam ser exatamente cumpridas durante o andamento da pesquisa. Podem ser realizados testes-piloto, para assegurar o rigor e completude do Protocolo de Revisão.

Outro ponto importante é a documentação de todo o processo de busca de estudos e a extração de dados. Uma revisão sistemática deve ser transparente e replicável. Portanto, todos os passos seguidos, todas as mudanças realizadas no protocolo de revisão ou mudanças na metodologia de busca/extração/síntese de dados, devem ser devidamente documentadas, de

modo que o leitor possa acompanhar todo o processo de realização da revisão, comprovando a sistematização da pesquisa.

Para TRAVASSOS (2007), o processo para realizar a etapa de Condução de uma Revisão Sistemática é definido pelos seguintes passos:

- Realização das Buscas nas fontes definidas: o processo deve ser transparente, repetível e documentado, assim como as mudanças que ocorrem no processo;
- Seleção dos Estudos Primários com os critérios de inclusão e exclusão definidos;
- Extração dos dados, desde informações gerais dos estudos às respostas para as questões de pesquisa. Formulários são um bom meio para registrar todos os dados necessários e o uso de uma ferramenta computacional pode apoiar a extração e registro dos dados e posterior análise;
- Avaliação da qualidade dos estudos é importante para balancear a importância de diferentes estudos, reduzir o viés (tendência a produzir “resultados tendenciosos” que se separam sistematicamente dos resultados verdadeiros), maximizar a validade interna e externa e guiar recomendações para pesquisas futuras;
- Síntese dos dados é realizada de acordo com as questões de pesquisa, por meio de quadros para realçar as similaridades e diferenças entre estudos. Se dados quantitativos estão disponíveis, pode-se considerar fazer uma meta-análise.

### 2.6.3 Apresentação da Revisão

A última etapa de uma revisão sistemática consiste, de acordo com a análise e síntese dos dados, na escrita do relatório da revisão sistemática e avaliação do mesmo. Posteriormente, os resultados são apresentados, com informações tabuladas de forma consistente com as questões de pesquisa, utilizando recursos como tabelas, gráficos, e destacando similaridades e

diferenças entre os resultados, isto é, ressaltando as possíveis análises e combinação de dados (COSTA, 2010).

Para a escrita do relatório, deve-se levar em consideração o formato no qual o mesmo irá ser divulgado. Geralmente uma revisão sistemática será relatada em pelo menos dois formatos, como afirma SANTOS (2010):

- Relatório técnico ou seção de uma tese/dissertação;
- Artigo em *journal* ou conferência: há certa restrição de espaço para este formato, portanto é recomendado o uso de um *template* de resumo estruturado.

Importante acrescentar que as etapas apresentadas, não necessariamente, precisam ser realizadas de forma sequencial, algumas delas podem ser realizadas de forma concorrente.

## 2.7 CONSIDERAÇÕES DO CAPÍTULO

A utilização de Mapeamento Sistemático na Engenharia de Software pode trazer diversos benefícios, entre eles podemos destacar o foco que este tipo de pesquisa proporciona ao cientista durante a sua execução. Isso se dá devido ao caráter metodológico desse tipo de investigação, que impede, por exemplo, a distração dos pesquisadores em outros artigos interessantes, mas não relevantes a sua pesquisa.

Outro benefício que pode ser destacado é a maior quantidade de informações obtidas sobre o tópico pesquisado, já que, por meio dos resultados de uma MSL, têm-se a possibilidade de verificar como o tópico de interesse vem sendo tratado na literatura e se existem lacunas em sua pesquisa, o que permite identificar novas oportunidades.

Ainda neste sentido, os resultados de um mapeamento sistemático podem: apontar se há trabalhos que tratem da área de interesse dos pesquisadores; identificar na literatura o quanto evidências empíricas suportam ou contradizem hipóteses teóricas; apresentar quais são seus pontos fortes e

fracos; e apoiar a geração de novas hipóteses. Todos estes fatores podem auxiliar, no contexto organizacional, na tomada de decisão.

É interessante pontuar que a utilização da experimentação no contexto da Engenharia de Software pode auxiliar na caracterização de tecnologias em uso, que podem passar por um processo evolutivo, por meio de novas experimentações em diferentes contextos, ou, então, definir a adoção de uma nova tecnologia.

Para a comunidade acadêmica, o que inclui pesquisadores, estudantes e orientadores, o mapeamento sistemático é um importante meio para realização de novas pesquisas com alta confiabilidade e valor científico, contribuindo, dessa forma, enormemente com o conhecimento científico voltado à Engenharia de Software (MAFRA e TRAVASSOS, 2006).

Neste capítulo foram apresentados os conceitos e definições sobre Revisão Sistemática da Literatura, bem como sua metodologia. O foco deste trabalho está na Condução de uma Revisão Sistemática, a partir da definição de um Protocolo de Revisão, apresentado no Capítulo 3.

### 3 PLANEJAMENTO DO MAPEAMENTO

Como parte do nosso Mapeamento Sistemático da Literatura (MSL), ao iniciar a etapa de planejamento, realizado neste trabalho, foi criado um protocolo de uma Revisão Sistemática, e este capítulo apresenta alguns tópicos definidos para a condução do mapeamento citado com base no protocolo da revisão.

#### 3.1 PROTOCOLO DE REVISÃO

O Protocolo de Revisão é o documento em que se encontra o planejamento de uma revisão sistemática. Nele são definidos os objetivos, o escopo, as restrições, os critérios, entre outras especificações para que uma revisão sistemática seja conduzida com sucesso. Nas subseções seguintes são apresentadas estas especificações no contexto deste trabalho, ou seja, apenas as seções do protocolo que se referem à etapa de planejamento da revisão e a etapa de Condução da Revisão sobre as fases de seleção dos Estudos Primários e avaliação da qualidade desses estudos. O protocolo completo pode ser visualizado no Apêndice A.

##### 3.1.1 Objetivos do Mapeamento

Este Mapeamento sistemático teve por objetivo identificar diferentes abordagens no ensino do processo de medição e do ensino de processos relacionados a Programas de Melhorias de Processos de Software (PMPS), por propostas de gamificação e jogos sérios. Ao identificar estas abordagens, este mapeamento evidenciará as dinâmicas, mecânicas e componentes de jogos presentes nestas abordagens com intuito de direcionar o desenvolvimento da ferramenta educacional proposta por uma tese de doutorado, vinculada a este trabalho. Desta forma, têm-se a seguinte estrutura para o objetivo, conforme proposto em (SANTOS, 2010):

- **Analisar:** relatos de experiência e publicações científicas através de um estudo baseado em mapeamento sistemático;
- **Com o propósito de:** identificar as abordagens na utilização de dinâmicas, mecânicas e componentes de jogos no ensino do processo de

medição e no ensino de PMPS, visando identificar as necessidades de informação, indicadores e medidas mais utilizadas em PMPS com ênfase no processo de medição de software;

- **Com relação ao:** emprego de proposta de gamificação e jogo sérios aplicada ao processo de medição dentro de Programas de MPS, visando o ensino ou a execução das atividades de Medição em organizações de desenvolvimento de software ou cursos que ministram engenharia de software;
- **Do ponto de vista:** dos pesquisadores;
- **No contexto:** acadêmico e industrial.

### 3.1.2 Questões de Pesquisa Principais

As questões levantadas foram organizadas conforme a estrutura *Population, Intervention, Context, Outcomes, Comparison* (PICOC), recomendada por Kitchenham (2007):

**Qual o estado da arte da pesquisa sobre aplicação ou ensino de programas de melhorias de processos de software (PMPS) utilizando jogos sérios ou gamificação?**

**População (P):** Organizações de Software e Instituições de ensino;

**Intervenção (I):** Abordagem utilizada para aplicar ou ensinar o processo de melhoria de software;

**Contexto (C):** Este trabalho visa explorar a comparação dos artigos voltados tanto para o ambiente industrial quanto para o acadêmico. E terá quatro pesquisadores executando as atividades deste mapeamento sistemático que é considerada de grande escala, pois sua duração foi de aproximadamente três trimestres;

**Resultados (O):** Objetivos das abordagens por gamificação ou jogo sério, dinâmicas, mecânicas e componentes de jogos presentes nas propostas

discutidas e a eficiência no ensino ou na aplicação prática do processo de melhoria de software ao utilizar estas propostas.

**Comparação (C):** Não se aplica a este trabalho.

**Qual o estado da arte da pesquisa sobre aplicação ou ensino de medição de software utilizando jogos sérios ou gamificação?**

**População (P):** Organizações de Software e Instituições de ensino;

**Intervenção (I):** Abordagem utilizada para aplicar ou ensinar o processo de medição de software;

**Contexto (C):** Este trabalho visa explorar a comparação dos artigos voltados tanto para o ambiente industrial quanto para o acadêmico. Terá quatro pesquisadores executando as atividades deste mapeamento sistemático que é considerada de grande escala, pois sua duração foi de aproximadamente três trimestres;

**Resultados (O):** Objetivos das abordagens por gamificação ou jogo sério, dinâmicas, mecânicas e componentes de jogos presentes nas propostas discutidas e a eficiência no ensino ou na aplicação prática do processo de medição ao utilizar estas propostas.

**Comparação (C):** Não se aplica a este trabalho.

### 3.1.3 Questões de Pesquisa Secundárias

Um conjunto de questões secundárias referentes à primeira e à segunda questão principal foram estabelecidas, questões estas para serem respondidas durante a fase de extração de informações. Tais questões têm objetivo de esclarecer detalhes importantes que este mapeamento procura identificar, para colaborar com o projeto onde esta se insere.

### **Questões Secundárias Gerais(QSG)**

**QSG1.** Em quais contextos foi aplicada a proposta de gamificação ou Jogo sério? (Ensino, Trabalho, ...)

**QSG2.** Quais as limitações relatadas no uso da gamificação ou Jogo Sérios para o ensino?

**QSG3.** Quais métodos de pesquisa foram empregados na validação da proposta de gamificação ou jogo sério?

**QSG4.** Quais os elementos de jogos foram utilizados na proposta de gamificação ou jogo?

**QSG5.** Quais as mecânicas de jogos foram utilizadas na proposta de gamificação ou jogo?

**QSG6.** Quais as dinâmicas de jogos foram utilizadas na proposta de gamificação ou jogo sério?

**QSG7.** O jogo sério ou a proposta de gamificação são considerados apropriados em termos e relevância de conteúdo, corretude, grau de dificuldade, método de ensino, duração no contexto o qual é pretendido e a repetibilidade do jogo? O jogo é considerado engajador? Quais seus pontos fracos e pontos fortes?

**QSG8.** Como foi a eficiência no aprendizado ou na aplicação do programa de melhorias de processos de software com a utilização da proposta de gamificação ou jogo sério ao comparar com o meio tradicional de aprendizado ou aplicação?

### **Questões Secundárias Específicas de Melhoria de Processos de Software(QSMPS)**

**QSMPS1.** Em quais processos foi aplicada a proposta de gamificação ou Jogo sério na área de MPS? (Medição, Coleta de Requisitos, ...)

**QSMPS2.** Como educadores ou a indústria podem se beneficiar na aplicação ou ensino de programas de melhorias de processos de software (MPS) através de propostas de gamificação ou jogos sérios?

### **Questões Secundárias Específicas de Medição de Software (QSMS)**

A seguir será apresentado um conjunto de questões secundárias referentes à segunda questão principal:

**QSMS1.** A proposta apresentada foi baseada em algum modelo ou norma ou paradigma? Se sim, qual?

**QSMS2.** Quais métricas foram cobertas pela proposta de gamificação e jogo sérios?

**QSMS3.** Quais tópicos de medição foram cobertos pelas propostas? (Coletar, Armazenar, Analisar e Relatar)

**QSMS4.** Como educadores ou a indústria podem se beneficiar no ensino ou aplicação de programas de medição de software através de propostas de gamificação ou jogos sérios?

#### 3.1.4 Escopo da Pesquisa

Foram definidos critérios para a seleção das fontes de pesquisa e restrições para garantir a viabilidade da pesquisa e com isso limitar um escopo de pesquisa.

Para a seleção das fontes de pesquisa, foram definidos os seguintes critérios:

- Disponibilidade para consultas web;
- Disponibilidade para busca de artigos através do domínio da UFPA;
- Disponibilidade de artigos na íntegra através do domínio da UFPA ou a partir da utilização da *engine* de busca Google ou Google Scholar ou Portal Capes;

- Disponibilidade de artigos em inglês ou português;
- Que possuam máquinas de busca;
- A pesquisa não pode incorrer em ônus financeiro aos pesquisadores. Portanto, apenas foram selecionadas as fontes que possibilitam consultas de forma gratuita (também serão consideradas fontes que oferecem consultas de forma gratuita a partir do acesso pelo domínio da UFPA);
- A pesquisa esteve restrita aos resultados publicados entre 01 de janeiro de 2000 até 31 de janeiro de 2017.

#### 3.1.5 Seleção de Fontes

Com base nos critérios de seleção e nas restrições da pesquisa, foram selecionadas as seguintes fontes de pesquisa onde foram realizadas as buscas dos estudos primários, de acordo com disponibilidade de pesquisa e relevância para área de Qualidade de Software:

- IEEEExplore Digital Library;
- EI Compendex;
- ISI Web of Knowledge (Web of Science);
- ACM;
- Scopus;
- Science Direct.

#### 3.1.6 Métodos de Busca Primária

Os métodos que foram utilizados para as buscas seguem a estratégia de:

1. Identificação de Palavras-chave e sinônimos;
2. Geração de *Strings* de Busca;

3. Realização das buscas nas fontes de pesquisas selecionadas (aplicação de Strings de busca);

Nas subseções seguintes são detalhadas as estratégias que foram utilizadas neste trabalho.

#### 3.1.6.1 Identificação de Palavras-Chaves e Sinônimos

A partir das questões de pesquisa, palavras-chave foram identificadas em acordo com a estrutura População, Intervenção e Resultados para a posterior formulação da *string* de busca. Segue a listagem de palavras-chave definidas:

Para a primeira questão principal de pesquisa:

- POPULAÇÃO
  - Inglês: *Project, Development, Organization, Enterprise, Company, Industry, Institute, Research Group, Technology Center;*
- INTERVENÇÃO
  - Inglês: *Process, Improvement, SPI;*
- CONTEXTO
  - *Learning, Teaching, Education, Training, Practice, Application;*
- RESULTADOS
  - Inglês: *Gamification, Game, Serious Game, Funware, Game Elements, Game Mechanics, Game Component, Game factor, Game aspect;*

Para a segunda questão principal de pesquisa:

- POPULAÇÃO
  - Inglês: *Project, Development, Organization, Enterprise, Company, Industry, Institute, Research Group, Technology Center;*
- INTERVENÇÃO

- Inglês: *Process, Measuring, Software, Measurement, Metrics, Metrology*;

- CONTEXTO

- *Learning, Teaching, Education, Training, Practice, Application*;

- RESULTADOS

- Inglês: *Gamification, Game, Serious Game, Funware, Game Elements, Game Mechanics, Game Component, Game factor, Game aspect*;

### 3.1.6.2 Geração de *Strings* de Busca

A *string* de busca é o agrupamento das palavras-chave, por meio dos operadores <OR> e <AND>. O operador <OR> é utilizado para o agrupamento das palavras-chave e sinônimos, por elemento (População, Intervenção e Resultados). O operador <AND> é utilizado para agrupar o conjunto de palavras-chave definidos para todos os elementos, de acordo com a estrutura PICO, conforme segue (SANTOS, 2010):

**P <and> I <and> C <and> O**

Importante observar que o elemento Comparison (C) não está no contexto desse trabalho, logo, o conjunto de palavras-chave para este elemento é vazio. Para as questões de pesquisa será considerada a seguinte *string* de busca em inglês, e com base nesta foi gerada uma *string* de busca para cada fonte de pesquisa automática devido a diferença entre as máquinas de busca de cada fonte ser diferente.

**Para Science Direct Q1:**

**Inglês:** *title-abstr-key(Software AND (Project OR Development OR Organization OR Enterprise OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification))*;

**Para Science Direct Q2:**

**Inglês:** *title-abstr-key(Software AND (Project OR Development OR Organization OR Enterprise OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*) AND (Gam\* OR Funware OR Ludification));*

**Para IEEE Q1:**

**Inglês:** *Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification);*

**Para IEEE Q2:**

**Inglês:** *(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*) AND (Gam\* OR Funware OR Ludification));*

**Para ACM (Association for Computing Machinery) Q1:**

**Inglês:** *recordAbstract(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification));*

**Para ACM (Association for Computing Machinery) Q2:**

**Inglês:** *recordAbstract(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*) AND (Gam\* OR Funware OR Ludification));*

**Para EI compendex Q1:**

**Inglês:** *(((((Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training*

*OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification)) WN KY AND ({english} WN LA));*

**Para EI Compendex Q2:**

**Inglês:** *(((Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*) AND (Gam\* OR Funware OR Ludification)) WN KY AND ({english} WN LA)));*

**Para Web of Science Q1:**

**Inglês:** *(TS=(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification)));*

**Para Web of Science Q2:**

**Inglês:** *(TS=(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*) AND (Gam\* OR Funware OR Ludification)));*

**Para Scopus Q1:**

**Inglês:** *((TITLE-ABS-KEY(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification))) AND (PUBYEAR > 1999)) AND ( LIMIT-TO ( LANGUAGE,"English " ) );*

**Para Scopus Q2:**

**Inglês:** *(TITLE-ABS-KEY(Software AND (Project OR Development OR Organization OR Enterprise OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*))*

AND (Gam\* OR Funware OR Ludification))) AND ( ( PUBYEAR > 1999 ) ) AND ( LIMIT-TO ( LANGUAGE, "English" ) );

### 3.1.7 Seleção dos Estudos Primários

A etapa de seleção de estudos primários pode ser dividida em: Definição de Critérios de Inclusão e Exclusão de Estudos Primários e Definição do Processo de Seleção dos mesmos.

#### 3.1.7.1 Critérios de Inclusão e Exclusão dos Estudos Primários

Os critérios de Inclusão e Exclusão dos Estudos Primários são os que vão nortear os pesquisadores na seleção dos estudos que foram coletados das fontes de pesquisas, além do que determina o rigor da pesquisa e impossibilita os vieses dos pesquisadores no momento da seleção. Os critérios de exclusão dos artigos foram:

CE.1) Artigos que não estejam disponíveis livremente para consulta ou download (em versão completa) aos alunos da universidade federal do Pará;

CE.2) Artigos que não foram encontrados ao utilizar as *Strings* de busca nos indexadores de conteúdo;

CE.3) Artigos não presentes nas bases listadas;

CE.4) Artigos que violem os princípios de alguma das bases listadas. Como artigos que plagiam outros artigos e afins;

CE.5) Artigos repetidos (em mais de uma fonte de busca) terão apenas sua primeira ocorrência considerada;

CE.6) Artigos duplicados terão apenas sua versão mais recente ou a mais completa considerada, salvo casos em que haja informações complementares;

CE.7) Estudos enquadrados como resumos, keynote speeches, cursos, tutoriais, workshops, pôsteres e afins;

CE.8) Artigos que não estiverem inseridos no contexto de ensino ou aplicação de programas de melhoria de software ou programas de medição através do uso de gamificação ou jogo sério;

CE.9) Artigos que não estiverem em inglês ou em português;

CE.10) Artigos irrelevantes para a pesquisa de acordo com as questões levantadas;

Os critérios de inclusão dos artigos baseiam-se em:

CI.1) Artigos que apresentem relatos de experiência na indústria, ou na academia, ou pesquisas de caráter experimental ou teórico, que apresentem exemplos de aplicação, descrição de experimentos ou casos reais de uso de abordagens de ensino ou aplicação de programas de melhoria de software ou programas de medição através do uso de gamificação ou jogo sério;

#### 3.1.7.2 Processo de Seleção dos Estudos Primários

Para a execução deste mapeamento sistemático foram seguidos os seguintes passos:

- Quatro pesquisadores (um doutorando, um mestrando e dois graduandos) foram os participantes na condução das atividades deste mapeamento sistemático. E como primeiro passo, estes realizaram o teste (verificação e validação) das *strings* de busca com o intuito de averiguar sua acurácia no retorno dos artigos primários e também, assim, poder criar múltiplas instâncias destas *strings* adaptadas para cada base de dados;
- Após os testes das *strings* de busca, os quatro pesquisadores realizaram a busca nos indexadores de conteúdo científico por meio do domínio da Universidade Federal do Pará para encontrar os possíveis artigos primários;
- Os alunos leram os títulos e resumos dos artigos retornados pela *string* de busca. Para assim, criarem uma lista com os possíveis artigos primários;

- Os artigos presentes na lista de possíveis artigos primários, tiveram seus títulos, resumos, introduções e conclusões lidos. E neste momento os alunos aplicaram os critérios de inclusão e exclusão para descartar os falsos positivos. E assim, criar uma lista dos artigos primários e uma lista dos artigos excluídos;

- As listas contendo os artigos primários foram comparadas e unificadas. Quando houve discordância entre os alunos na inclusão ou exclusão de um artigo, o mesmo, foi incluído;

- Os estudos presentes na lista foram lidos em sua totalidade e os critérios de qualidade foram aplicados. Assim, os artigos que obtiveram uma determinada pontuação de qualidade foram compilados na lista de artigos primários com qualidade verificada;

- Após, todos os artigos presentes na lista gerada anteriormente passaram pela estratégia de extração de dados;

- Ademais, todos os documentos e procedimentos foram validados através de reuniões com o coordenador do Projeto SPIDER, que possui uma gama de artigos publicados em periódicos e conferências sobre o tema de medição e também tem orientado várias dissertações de mestrado e trabalhos de conclusão de curso que versam sobre o tema de medição. Além disto, possui experiência prática na implementação do processo de medição ao ter prestado consultoria a diversas empresas brasileiras sobre este tema, sendo certificado como avaliador, consultor-implementador e instrutor oficial dos modelos de melhoria de processo e produto de software, CMMI, MPS.BR, Certics, Medepros, QPS. Para mais, o professor também é o orientador do aluno que conduz a tese de doutorado;

### 3.1.8 Avaliação dos Estudos Primários

A qualidade de um artigo pode ser mensurada pela relevância e valor científico de seu conteúdo. A avaliação da qualidade dos Estudos Primários é considerada, também, um critério de exclusão, a ser aplicado durante a condução da pesquisa. Apesar de não existir uma definição universal do que

seja qualidade de estudo, a maioria dos *checklists* incluem questões que objetivam avaliar a extensão em que o viés é minimizado e a validação interna e externa são maximizadas (KHAN *et al.*, 2001; KITCHENHAM, 2007).

Desta forma, durante a análise dos Estudos Primários e a coleta dos resultados, foram aplicados os critérios de qualidade, permitindo um processo adicional de validação dos estudos, de forma a identificar possíveis trabalhos que ainda devem ser desconsiderados da pesquisa e observar o grau de importância dos estudos individualmente para quaisquer comparações durante a síntese dos dados (KITCHENHAM, 2007).

Adicionalmente, a avaliação da qualidade pode servir como recomendação de estudos para futuras pesquisas, fornecendo informações a respeito da qualidade das informações de cada estudo avaliado (KITCHENHAM, 2007).

Artigos que, porventura, foram excluídos por não se adequarem aos critérios de qualidade foram citados, juntamente com as razões para sua exclusão. Após este passo, os artigos incluídos na pesquisa passaram pela fase de extração de dados.

#### 3.1.8.1 Critérios de Qualidade dos Estudos Primários

Os critérios de qualidade que foram aplicados aos Estudos Primários foram adaptados de (COSTA, 2010), uma vez que descreviam critérios abrangentes o suficiente para cobrir o escopo dos estudos a serem considerados, com alterações para se adequarem aos objetivos e questões de pesquisa desta revisão sistemática. Os critérios de qualidade foram:

##### **Introdução**

Os objetivos ou questões do estudo são claramente definidos? E a problemática abordada na pesquisa está descrita claramente (incluindo justificativas para a realização do estudo)?

O tipo de estudo está definido claramente?

##### **Desenvolvimento**

Existe uma clara descrição do contexto no qual a pesquisa foi realizada?

O trabalho é adequadamente referenciado (apresenta trabalhos relacionados ou semelhantes e baseia-se em modelos e teorias da literatura)?

### **Conclusão**

O estudo relata de forma clara e não ambígua os resultados?

Os objetivos ou questões do estudo são alcançados?

### **Crítérios para a Questão de Investigação**

O estudo apresenta primária ou secundariamente uma abordagem ou ferramenta de ensino ou de aplicação de programas de melhoria de software ou programas de medição através do uso de gamificação ou jogo sério?

### **Critério Específico para estudos Experimentais**

Existe um método ou um conjunto de métodos descrito para a realização do estudo?

### **Critério Específico para estudos Teóricos**

Existe um processo não tendencioso na escolha dos estudos?

### **Critério Específico para Revisões Sistemáticas**

Existe um protocolo rigoroso, descrito e seguido?

### **Critério Específico para Relato de Experiência Industrial**

Existe uma descrição sobre a(s) organização(ões) onde foi conduzido o estudo?

#### **3.1.8.2 Processo de Avaliação dos Estudos Primários**

A execução do processo da etapa de avaliação dos Estudos Primários ocorreu a partir da adoção dos seguintes critérios, usando a tabela abaixo como referência:

**Concordo totalmente (4):** deve ser concedido no caso em que o trabalho apresente no texto os critérios que atendam totalmente a questão;

**Concordo parcialmente (3):** deve ser concedido no caso em que o trabalho atenda parcialmente aos critérios da questão;

**Neutro (2):** deve ser concedido no caso em que o trabalho não deixa claro se atende ou não a questão;

**Discordo parcialmente (1):** deve ser concedido no caso em que os critérios contidos na questão não são atendidos pelo trabalho avaliado;

**Discordo totalmente (0):** deve ser concedido no caso em que não existe nada no trabalho que atenda aos critérios da questão.

Tabela 1 – Escala por critério

Escala por Critério	
Critério	Escala
1a.	4 – Define claramente os objetivos e justifica o estudo. 3 - Define claramente o estudo, porém a justificativa não é clara ou não justifica o estudo. 2 – Não Define os objetivos, mas justifica o estudo. 1 - A definição e a justificativa do estudo não são claras. 0 - Não define os objetivos e nem justifica o estudo.
1b.	4 – Define o tipo de estudo, referenciando na literatura a metodologia. 3 – Define o tipo de estudo, porém sem referenciar a metodologia.

	<p>2 – Não define o tipo de estudo. É possível inferir facilmente.</p> <p>1 – Não define o tipo de estudo. É possível inferir com dificuldade.</p> <p>0 – Não é possível inferir o tipo de estudo.</p>
2a.	<p>4 – Define claramente uma seção com o contexto da pesquisa.</p> <p>3 – O contexto da pesquisa está incluído em uma seção não exclusiva.</p> <p>2 – O contexto da pesquisa está disperso ao longo do texto.</p> <p>1 – O contexto da pesquisa está disperso e é insubstancial.</p> <p>0 – O contexto da pesquisa não é abordado.</p>
2b.	<p>4 – O texto apresenta uma seção de trabalhos relacionados.</p> <p>3 – O texto apresenta trabalhos relacionados em uma seção não exclusiva.</p> <p>2 – O texto apresenta trabalhos relacionados dispersos ao longo do texto.</p> <p>1 – O texto não apresenta trabalhos relacionados, mas se apoia na literatura.</p> <p>0 – O texto não apresenta trabalhos relacionados nem se apoia na literatura.</p>
3a.	<p>4 – Resultados são claramente apresentados na seção de conclusão.</p> <p>3 – Resultados são claramente referenciados na seção de conclusão.</p> <p>2 – Resultados apresentados na conclusão não são claros.</p> <p>1 – Resultados referenciados na conclusão não são claros.</p> <p>0 – Não são apresentados resultados.</p>

3b.	<p>4 – Os resultados estão totalmente aderentes ao objetivo do estudo.</p> <p>3 – Os resultados estão aderentes ao objetivo do estudo, no entanto o autor faz ressalvas.</p> <p>2 – Os resultados são parcialmente aderentes ao objetivo do estudo.</p> <p>1 – Os resultados não estão aderentes ao objetivo do estudo.</p> <p>0 – Não é alcançado nenhum resultado.</p>
4a.	<p>4 – Apresenta uma ferramenta ou abordagem de maneira detalhada e validada com usuários.</p> <p>3 – Apresenta uma ferramenta ou abordagem de maneira detalhada, porém não validada.</p> <p>2 – Apresenta uma ferramenta ou abordagem, porém sem detalhar a mesma. Contudo validando a mesma com os usuários.</p> <p>1 – Apresenta uma ferramenta ou abordagem, porém sem detalhar e validar a mesma com os usuários.</p> <p>0 – Apenas referência uma ferramenta ou não apresenta uma ferramenta ou abordagem.</p>
5a.	<p>4 – O método de experimento é definido e referenciado claramente.</p> <p>3 – O método de experimento é definido claramente.</p> <p>2 – O método de experimento é citado.</p> <p>1 – O método de experimento não é citado, porém é possível inferir.</p> <p>0 – Não é possível inferir o método de experimento.</p>

6a.	<p>4 – O texto descreve critérios para a escolha dos estudos.</p> <p>3 – O texto não descreve critérios para a escolha dos estudos, porém apresenta estudos que discordam do estudo apresentado.</p> <p>2 – O texto descreve apenas estudos aderentes ao estudo apresentado.</p> <p>1 – O texto descreve estudos insuficientes.</p> <p>0 – O texto não descreve estudos base.</p>
7a.	<p>4 – O protocolo de revisão é apresentado, descrito e seguido.</p> <p>3 – O protocolo de revisão é apresentado e descrito, porém há evidências de que não foi seguido adequadamente.</p> <p>2 – O protocolo de revisão não foi suficientemente descrito.</p> <p>1 – O protocolo de revisão apenas foi citado ao longo do texto.</p> <p>0 – Não há um protocolo de revisão.</p>
8a.	<p>4 – A área de atuação, tamanho e origem da organização são informados.</p> <p>3 – Apenas duas das características do item 4 são informadas.</p> <p>2 – Apenas uma das características do item 4 é informada.</p> <p>1 – Nenhuma das características do item 4 é informada.</p> <p>0 – O estudo não foi conduzido em uma ou mais organizações.</p>

**Fonte: Elaborado pelo autor.**

Deve-se então, dada à soma das pontuações, enquadrá-los em um dos cinco níveis de qualidade classificados por Beecham (2007), tais como, apresentados na Tabela a seguir.

**Tabela 2 - Níveis de Qualidade**

Faixa de Notas	Avaliação
----------------	-----------

Excelente	>86%
Muito Boa	66%-85%
Boa	46%-65%
Média	26%-45%
Baixa	< 26%

Fonte: Beecham (2007).

### 3.2 CONSIDERAÇÕES DO CAPÍTULO

O planejamento de uma revisão e mapeamento sistemático é uma das etapas mais importantes desse processo, pois é durante essa etapa que são definidos aspectos que norteiam o pesquisador durante a condução do mapeamento, fazendo com que o mesmo não perca o foco, além de eliminar prováveis vieses.

O protocolo de revisão é o artefato que garante o caráter repetível de uma revisão sistemática, além de comprovar que a revisão realizada foi planejada e que as decisões tomadas durante essa etapa foram registradas. O processo para a definição do protocolo de revisão não é sequencial, o que garante a revisão, modificação e/ou refinamento do planejamento de forma iterativa.

Este capítulo apresentou a fase de planejamento do mapeamento sistemático que foi realizada no contexto deste trabalho, com a definição do protocolo de revisão.

## 4 RESULTADOS E ANÁLISE DO MAPEAMENTO SISTEMÁTICO

O presente capítulo apresenta a principal contribuição deste trabalho, a aplicação das técnicas de revisão sistemática da literatura a partir de um mapeamento sistemático durante a fase de condução que é apresentada conforme definida no protocolo de revisão, discutida no Capítulo 3.

### 4.1 CONDUÇÃO DO MAPEAMENTO

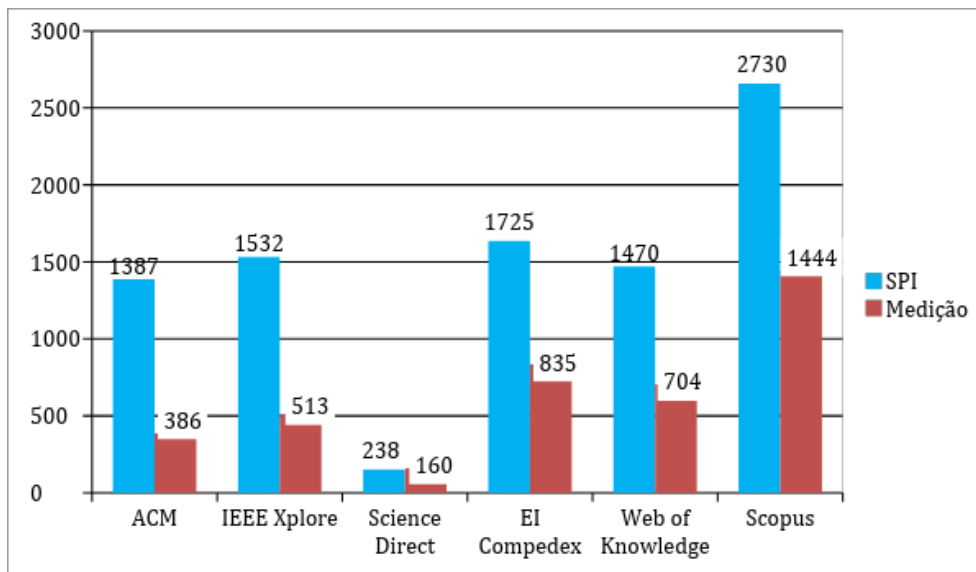
Neste trabalho, a fase de condução foi dividida em: Busca Primária, Seleção de Estudos Primários e Avaliação de Qualidade dos Estudos. A execução, resultados e análise dos mesmos, são apresentados nas subseções seguintes.

#### 4.1.1 Busca Primária

Conforme definido no Protocolo de Revisão, no que tange o método de busca a ser realizado nas fontes selecionadas, aplicamos em cada base a busca primária, com uma *string* de busca específica, assim como foi mostrado no item 3.1.6.2 deste trabalho. A partir disso, obteve-se um total de 13124 trabalhos, sendo 9082 em SPI e 4042 em Medição de Software, no qual foram identificados 1773 na ACM, 2045 no IEEE Xplore, 398 no Science Direct, 2560 no EI compendex, 2174 na Web of Knowledge e 4174 no SCOPUS. Como pode-se visualizar no Gráfico 4.1.

É importante frisar que a quantidade de estudos retornada por cada fonte pode ser justificada por diversos fatores, tais como, a quantidade total de estudos na fonte e a relevância da fonte para a questão de pesquisa.

Gráfico 1 – Número de Estudos retornados a partir da Busca Primária



#### 4.1.2 Seleção de Estudos Primários

Tendo em vista que o número de estudos retornados na busca primária foi satisfatório para que continuasse a condução do mapeamento, deu-se continuidade à pesquisa inserindo todos os artigos retornados na ferramenta JabRef, e depois foi realizada a verificação com relação às restrições da pesquisa. Os estudos que foram aprovados foram inseridos em uma Tabela de Seleção criada para cada fonte de pesquisa.

A partir do processo de seleção definido no Capítulo 3, cada um dos pesquisadores selecionou, inicialmente, os possíveis estudos primários por meio da leitura do título e resumo de cada estudo retornado, para, assim, criarem uma lista com os possíveis artigos primários. Os artigos, presentes na nova lista criada após o passo anterior, tiveram seus títulos, resumos, introduções e conclusões lidos. E neste momento, os critérios de inclusão e exclusão foram aplicados para descartar os falsos positivos, e assim criar uma lista de artigos primários e uma lista de artigos excluídos.

Após este processo, as Tabelas de Seleção dos pesquisadores para cada fonte foram comparadas, e, por fim, durante reuniões entre estes, foi

realizado o consenso entre as divergências para a inclusão de estudos, seguindo o que previamente foi definido no protocolo de revisão.

O número de estudos retornados na busca primária foi relativamente alto, porém a partir do processo de seleção definido no Protocolo de Revisão Sistemática, anexo a este documento, esse número foi bastante reduzido. O Quadro 5 apresenta a evolução em números do processo de seleção de Estudos Primários. O quadro mostra os valores na busca primária para cada *string*, no total, 13124 estudos retornados, e que a partir da primeira seleção por título e resumo, foram identificados 711 estudos potencialmente relevantes para a pesquisa, sendo 594 em SPI e 117 em Medição de Software.

Com a leitura do título, resumo, introdução e conclusão dos estudos potencialmente relevantes, e utilizando-se os critérios de inclusão e exclusão, chegou-se a 100 Estudos Primários e 68 Estudos Secundários, disponíveis no Apêndice B deste trabalho. Assim, 543 trabalhos considerados potencialmente relevantes na primeira seleção foram excluídos e os principais motivos para exclusão foram: não respondiam a nenhuma questão de pesquisa; identificados duas vezes por fontes diferentes, isto é, repetido ou duplicado; e, não estavam em português ou inglês. Os estudos excluídos e as razões que levaram a sua exclusão estão disponíveis no Apêndice C deste trabalho.

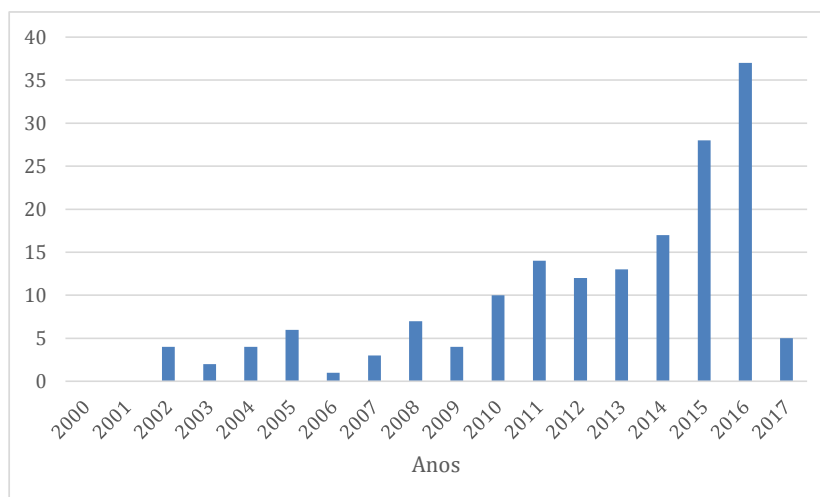
Quadro 1 – Seleção dos Estudos Primários

Seleção de Estudos Primários																
Fontes	Estudos Retornados		1ª Seleção (Título e Resumo)		2ª Seleção (Título, Resumo, Introdução e Conclusão)											
	SPI	Medição	Estudos Potencialmente Relevantes		Excluídos					Incluídos						
			SPI	Med.	Não Relevantes	Repetidos / Duplicados	Não estão em Inglês / Português	Inacessíveis	Estudos Primários	Estudos Secundários						
			SPI	Med.	SPI	Med.	SPI	Med.	SPI	Med.	SPI	Med.	SPI	Med.		
ACM	1387	386	54	10	17	1	9	7	0	0	1	0	13	2	14	0
IEEE Xplore	1532	513	108	27	45	10	0	12	4	0	0	0	21	4	36	2
Science Direct	238	160	14	3	7	2	0	1	0	0	0	0	6	0	1	0
EI																
Compendex	1725	835	120	25	32	8	50	12	0	0	0	0	24	4	11	4
Web of Knowledge	1470	704	121	18	42	3	60	15	0	0	0	0	8	0	11	0
Scopus	2730	1444	179	34	34	18	139	16	0	0	0	0	4	0	2	0
<b>Total</b>	<b>9082</b>	<b>4042</b>	<b>595</b>	<b>117</b>	<b>177</b>	<b>42</b>	<b>258</b>	<b>63</b>	<b>4</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>76</b>	<b>10</b>	<b>75</b>	<b>6</b>

[JL4] Comentário: Mudança de posição do quadro 1: colocado na posição horizontal a pedido do prof. Josivaldo.

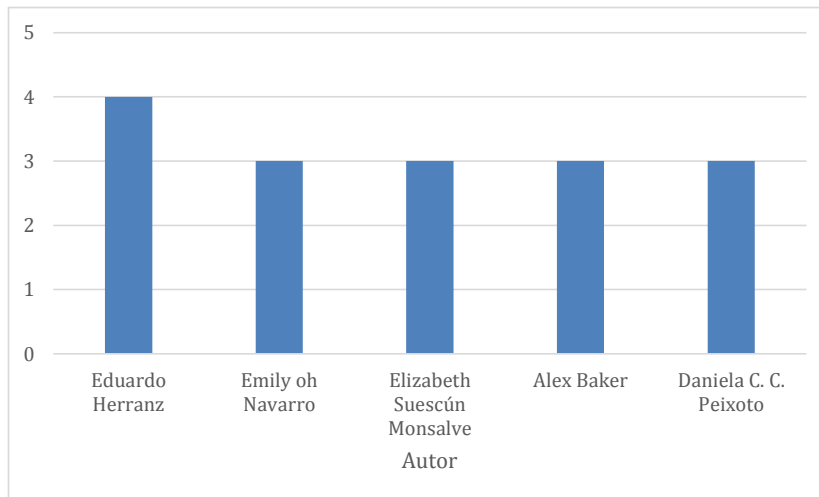
Como critério, os artigos foram limitados aos anos entre 2000 e 2017, devido aos estudos em Gamificação serem relativamente recentes. Após análise, pôde-se perceber no Gráfico 2, que houve um número maior de estudos a partir do ano de 2010, podendo concluir que isso se deve ao aumento dos estudos na área devido ao reconhecimento de sua importância nos últimos anos e ao aumento nas pesquisas.

**Gráfico 2 – Número de Estudos ao Longo dos Anos**



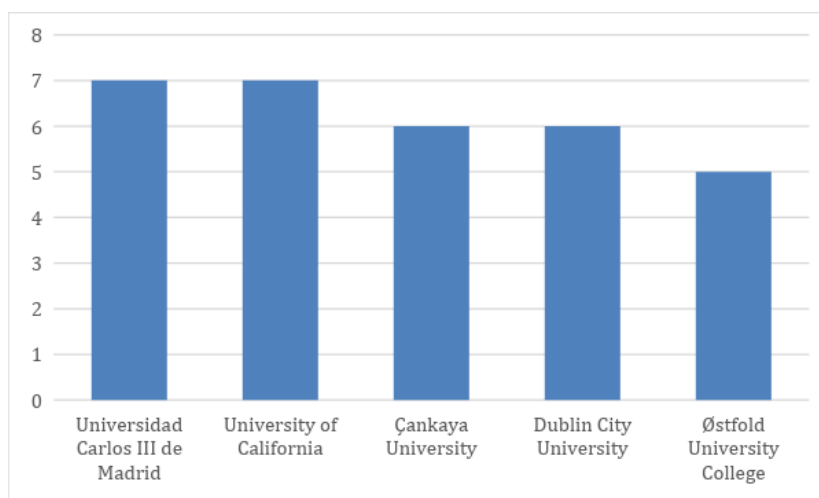
No Gráfico 3 pode-se observar o número de publicações por autores que entraram nesta pesquisa através de suas publicações dentro do espaço de tempo determinado. É importante informar que outros autores foram omitidos, devido ao número reduzido de artigos publicados com relação aos demais que aparecem no gráfico. Por questão de organização, foram incluídos os cinco autores que mais publicaram.

**Gráfico 3 – Número de Estudos por Autor**



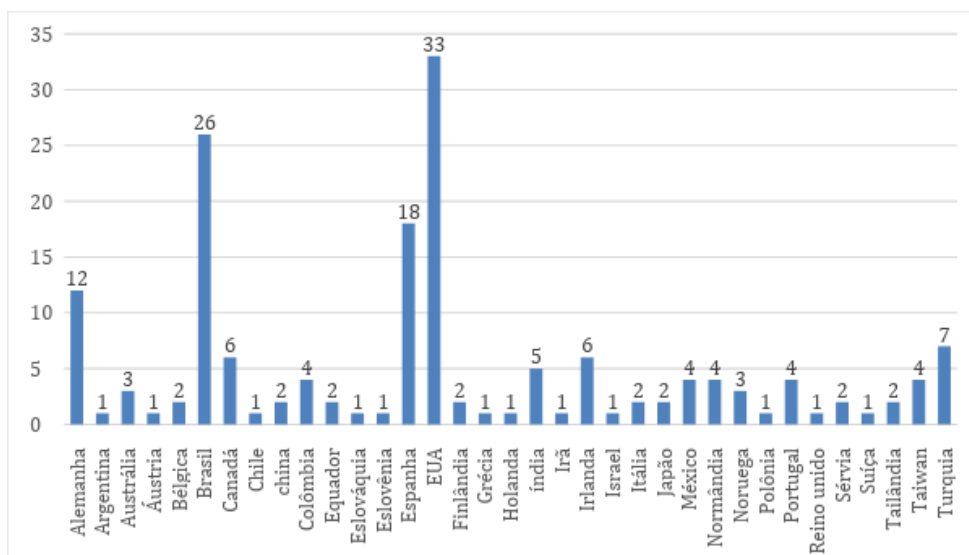
A distribuição de estudos por instituição de filiação pode ser visualizada no Gráfico 4, onde o número de instituições com publicações é maior do que se visualiza no gráfico, porém algumas foram omitidas. O gráfico apresenta as 5 principais instituições que publicaram sobre Gamificação e Jogos Sérios em Melhoria do Processo de Software, onde pode-se perceber que instituições de países europeus vem liderando as pesquisas na área, seguindo pela região da Califórnia (EUA) o que pode ser um fator de explicação para o grande aumento tecnológico que vem ocorrendo na região daquele país.

**Gráfico 4 – Número de Estudos por Instituição**



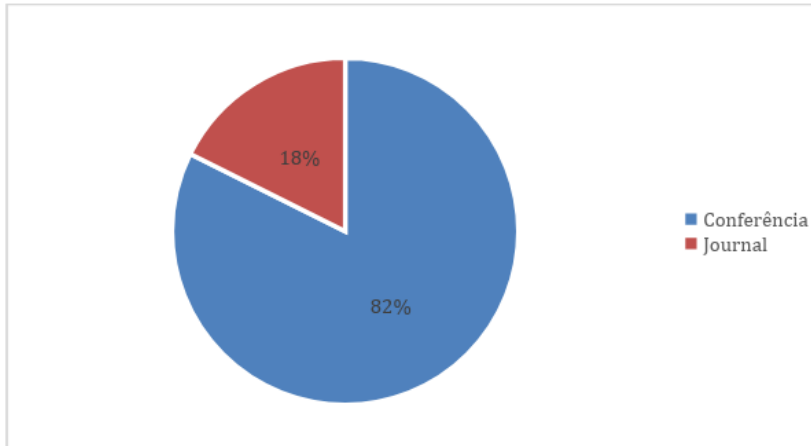
No gráfico 5 pode-se observar quais os países que mais publicam sobre Gamificação e Jogos Sérios em Melhoria do Processo de Software. Pode-se observar também a importância do Brasil, pois é o segundo país que mais publica sobre o assunto, ficando atrás apenas dos EUA. É importante ressaltar que as fontes de pesquisa não são nacionais, o que dá mais credibilidade. Espanha e Alemanha vêm logo em seguida, apresentando um número de publicações superior aos demais. Seguindo de Canadá e demais países da Europa. Podemos observar que os estudos não se limitam a um determinado continente, por ser uma área bastante abrangente e com potencial de crescimento em várias áreas de estudo.

**Gráfico 5 – Número de Estudos por País**



O Gráfico 6 apresenta a distribuição dos trabalhos por tipo de publicação, mostrando que a maioria dos Estudos Primários, 82%, foi publicada em anais de eventos (Conferências, *Workshops* e Simpósios). Já os outros 18% foram publicados em periódicos. Tal diferença entre os tipos de publicação pode ser explicada pelo fato da maior facilidade de publicação por anais. Periódicos tendem a ser mais criteriosos quanto a escolha de publicações e tendem a levar mais tempo, por isso, muitos pesquisadores tendem procurar para publicar em anais de eventos.

**Gráfico 6 – Distribuição de Estudos por Tipo de Publicação**



Os gráficos 7, 8 e 9 apresentam os 5 principais fóruns em que os estudos foram publicados, de acordo com seu tipo de publicação.

**Gráfico 7 – Principais Workshops**

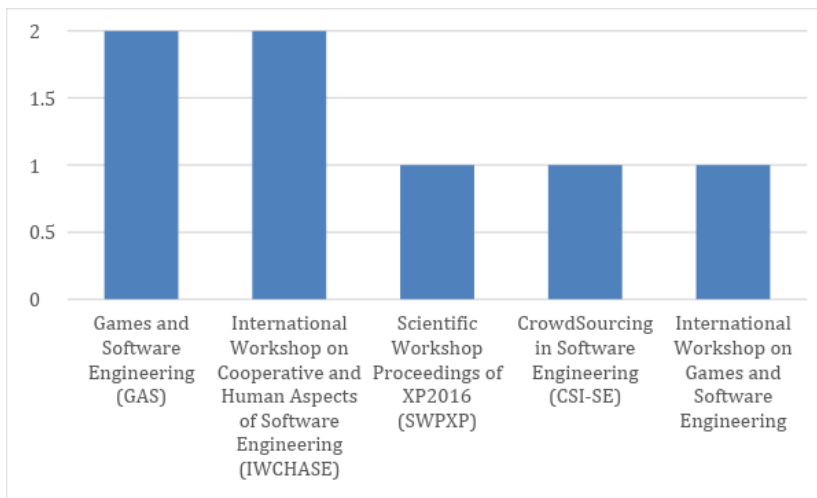


Gráfico 8 – Principais Conferências

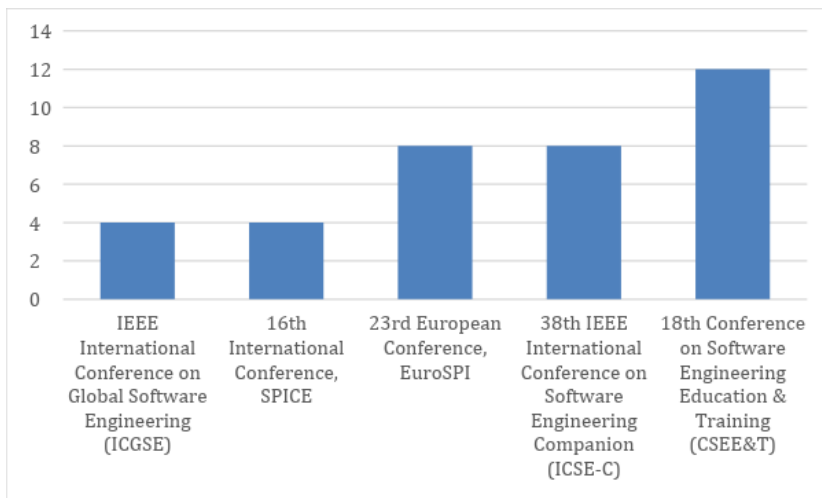
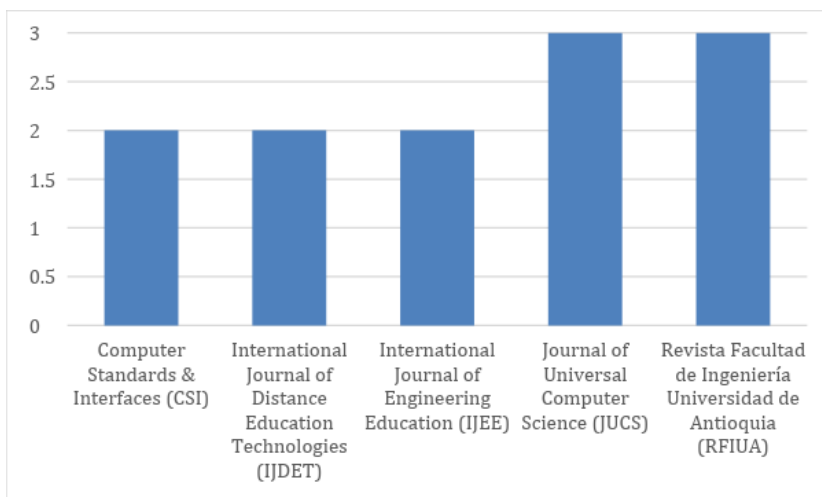


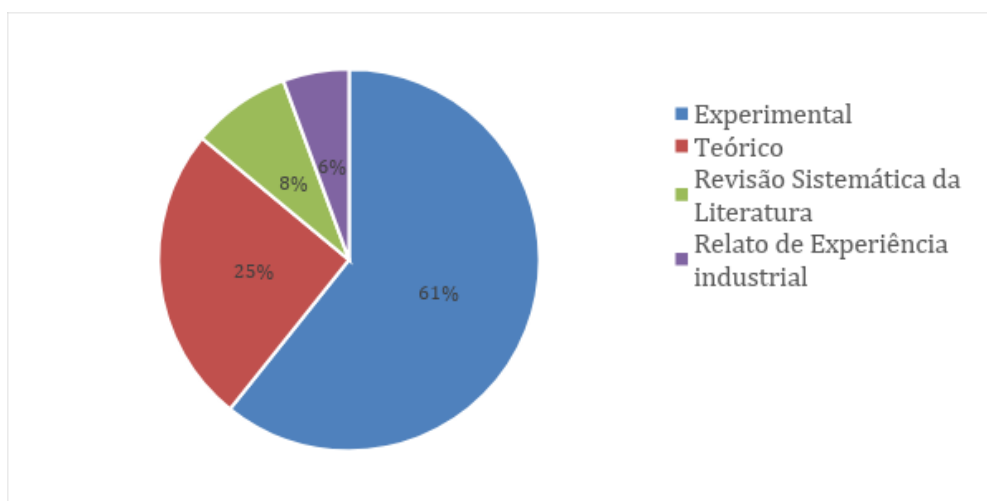
Gráfico 9 – Principais Periódicos



Dentre os 168 Estudos Primários e Secundários da pesquisa, 61% apresentam estudos experimentais (estudos baseados em evidências ou experimentos, *Empirical Studies*, em inglês), 25% teóricos (estudos conceituais baseados em um entendimento de uma área, referenciando outros trabalhos relacionados), 8% eram Revisão Sistemática da Literatura e apenas 6% caracterizam-se como relato de experiência industrial. Com estes dados, podemos inferir que os estudos experimentais sobre Gamificação e Jogos Sérios em Melhoria do Processo de Software estão crescendo, apesar de que a maioria das abordagens experimentais serem *ad-hoc*. Como é uma área em

expansão, a tendência é que estes números aumentem gradativamente. Estudos teóricos também aparecem em uma percentagem significativa, isso deve principalmente ao fato de ser uma área nova no contexto de projetos de software. Podemos verificar que ainda não há muitas RSL's, devido aos números de estudos ainda não serem tão abrangentes. Com número menor ainda que os anteriores, relatos de experiência industrial, que poderiam servir de base prática para o estudo teórico e a experiência na indústria ainda estão sendo iniciados e, a base para esses estudos, como uma RSL ainda está básico e com possibilidades de aumento no número de estudos com o passar dos anos. Analisando estes dados, pode-se ter uma noção do cenário atual sobre Gamificação e Jogos Sérios em Melhoria do Processo de Software, pois, quando se tem uma área nova em determinado contexto, tende-se a ter mais quantidade de estudos experimentais e estudos teóricos do que aplicações no contexto industrial. O Gráfico 10 ilustra a divisão dos tipos de estudos da pesquisa.

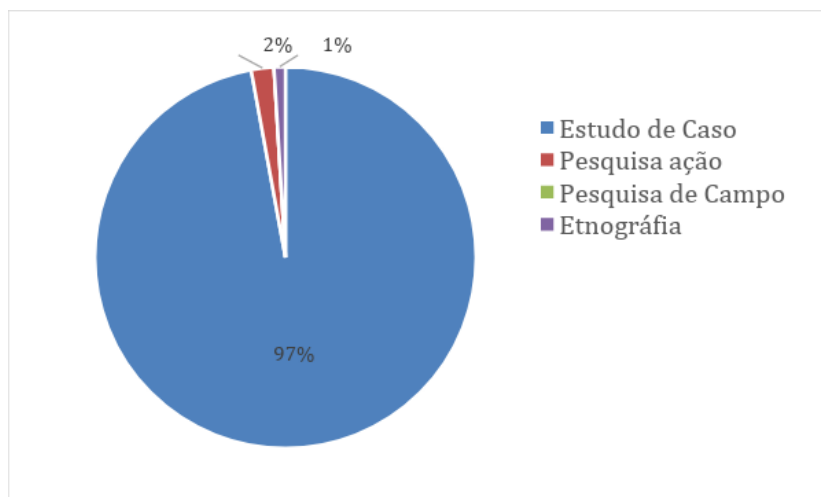
**Gráfico 10 – Distribuição dos Trabalhos por Tipo de Estudo**



Como visto, 61% dos Estudos Primários foram do tipo experimental, e os principais métodos identificados na pesquisa para a realização do estudo, baseado na classificação de Easterbrook *et al.* (2007), foram: Estudo de Caso, Pesquisa de Campo, Pesquisa-Ação e Etnografia. O Gráfico 11 ilustra a distribuição dos métodos utilizados, no qual, a maioria, 97%, realizaram Estudo

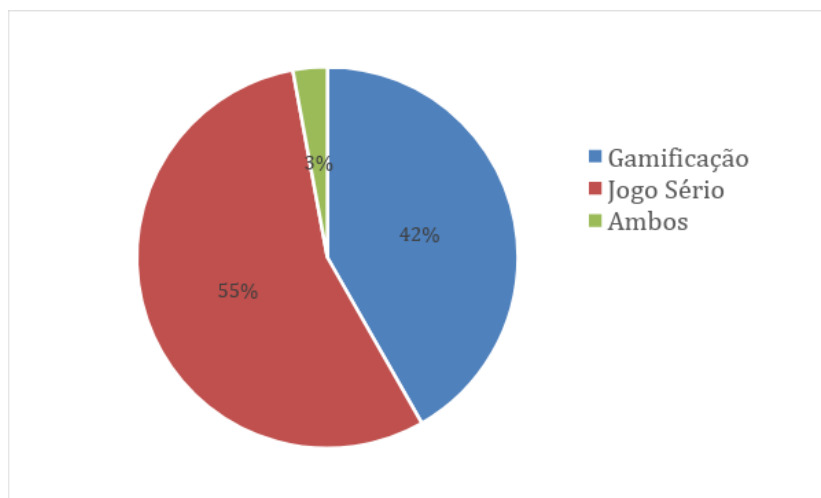
de Caso, enquanto apenas 2% realizaram Pesquisa ação e 1% Etnografia, e nenhum dos trabalhos utilizou Pesquisa de Campo.

**Gráfico 11 – Métodos Empregados nos Estudos do Tipo Experimental**



Quanto ao assunto principal dos trabalhos, foi possível classificá-los como: Gamificação ou Jogos Sérios. O Gráfico 12 ilustra esta divisão, demonstrando que 55% dos trabalhos, apresentou um estudo focado em Jogos Sérios, 42% abordavam Gamificação, além de 3% tratavam de ambos os assuntos. Isso demonstra que Jogos Sérios ainda possuem uma abordagem maior apesar de que a tendência é que os estudos sobre Gamificação aumente gradativamente com o passar dos anos.

Gráfico 12 – Gamificação vs Jogo Séri



#### 4.1.3 Avaliação dos Estudos Primários

Por fim, os resultados da avaliação da qualidade são apresentados pela Tabela 1. De acordo com as definições sobre o processo de avaliação de qualidade, um estudo poderia alcançar no máximo 36 pontos, devido a escala atribuída para a avaliação com base nos critérios de qualidade. Sendo assim, a nota final de um estudo seria definida segundo a proposta de Beecham *et al.* (2007), que divide a porcentagem da nota em 5 faixas, onde a cada uma é associada uma classificação, que pode ser: Excelente, Muito Boa, Boa, Média e Baixa.

Tabela 3 - Qualidade dos Estudos Primários

	Baixa < 26%	Média 26% - 45%	Boa 46% - 65%	Muito Boa 66% - 85%	Excelente > 86%	Total
Número de Estudos Primários	6	25	28	56	53	168
%	3,6%	14,9%	16,7%	33,3%	31,5%	100%

Como pode ser observado, 6 trabalhos (3,6%) estão na faixa baixa e 25 (14,9%) estão na faixa média, enquanto 28 estudos (16,7%) estão na faixa Boa, 56 estudos (33,3%) estão na muito boa e 53 estudos (31,5%) na faixa Excelente. Portanto, é possível observar que os trabalhos analisados apresentam qualidade acima da média de acordo com os critérios utilizados. Estes números trazem grande credibilidade aos resultados deste mapeamento,

uma vez que estudos acima da média, subentende-se que teremos um resultado final de qualidade.

Avaliar um estudo quanto a sua qualidade é importante devido às futuras evidências que podem ser encontradas, já que, quanto melhor for a avaliação de qualidade de um estudo, maior deve ser a importância que se deve dar às evidências fornecidas pelo mesmo.

## **4.2 CONSIDERAÇÕES DO CAPÍTULO**

Este capítulo apresentou parte da etapa de Condução de uma Revisão Sistemática, em que se pode verificar a aplicação das técnicas definidas no Protocolo de Revisão, tendo como resultado a sumarização de alguns números, que mostram características gerais desse tipo de pesquisa.

Os resultados obtidos podem auxiliar no planejamento de outros trabalhos em ESBE, assim como outras revisões, citadas no planejamento desta pesquisa, serviram para guiar o desenvolvimento desta. Esse auxílio pode se dar, por meio das definições gerais de uma RSL.

## **5 CONCLUSÕES**

Este capítulo apresenta as principais conclusões do trabalho realizado e as contribuições do mesmo para a área da Engenharia de Software no que tange a realização de Revisões Sistemáticas da Literatura, SPI e Medição em projetos de software. Desta forma, são apresentados a seguir uma visão geral sobre o trabalho realizado, os resultados obtidos e os trabalhos futuros.

### **5.1 VISÃO GERAL**

Com o objetivo geral de realizar uma Revisão Sistemática da Literatura para identificar diferentes abordagens no ensino do processo de medição e do ensino de processos relacionados a Programas de Melhorias de Processos de Software (PMPS), por propostas de gamificação e jogos sérios no contexto de projetos de software, no capítulo introdutório, foi apresentada a contextualização deste trabalho, por meio de uma visão geral sobre o assunto principal da pesquisa, sua justificativa e motivação, além de seus objetivos e a metodologia aplicada para a realização deste.

No segundo capítulo foi introduzido os conceitos de Melhoria do Processo de Software, Medição de Software, Engenharia de Software Baseada em Evidências e a definição de Estudos Primários e Secundários. Em seguida foi apresentado o conceito de Revisão Sistemática da Literatura, seu diferencial em relação a revisões informais e sua metodologia.

O terceiro capítulo apresentou os resultados da fase de planejamento da revisão, mais especificamente o Protocolo de Revisão, onde estão inseridas as definições que devem nortear toda a Condução de uma Revisão, desde a definição das questões de pesquisa até a forma de extração dos dados.

O quarto capítulo mostrou os resultados da fase de Condução da Revisão, onde foram apresentados: a quantidade de estudos retornados pelas buscas primárias realizadas nas fontes escolhidas; os resultados do processo de seleção que definiu os Estudos Primários a serem analisados; a análise realizada nos Estudos Primários selecionados e a sumarização de alguns números; e, por fim, os resultados da etapa de avaliação da qualidade desses estudos.

## 5.2 RESULTADOS OBTIDOS

Por meio deste trabalho, obteve-se o Protocolo de Revisão Sistemática da Literatura para SPI e Medição aplicando gamificação e jogos sérios no contexto de projetos de software. Este protocolo garante a repetição de pesquisas como essa, além de sua possível utilização para auxílio na realização de novas revisões sistemáticas.

Como resultado pode-se, também, apresentar o conjunto de Estudos Primários selecionados nesta pesquisa, bem como seus níveis de qualidade, o que possibilita a descoberta de abordagens (técnicas, modelos de processos, *frameworks* de processo, ferramenta, metodologias e afins) que apoiam a realização de boas práticas para tal processo. Este fato, pode auxiliar organizações para o desenvolvimento ou evolução de seu processo para SPI e Medição aplicando gamificação ou jogos Sérios, além de uma visão geral de como se encontra o fenômeno estudado na literatura, quais as principais pesquisas e quais as prováveis lacunas encontradas.

Os gráficos trazem uma visão, apesar de serem fontes superficiais, mais clara dos resultados obtidos. Com eles, pode-se obter de maneira rápida e prática padrões dos estudos de SPI e Medição de Software, através de cruzamentos entre eles.

## 5.3 PONTOS FORTES E OPORTUNIDADES DE MELHORIA

A realização da RSL apresentada forneceu aos autores desse trabalho um entendimento mais abrangente sobre Melhoria do Processo de Software, Medição de Software, Gamificação e Jogos Sérios no contexto de projetos de software e conhecimento sobre outros assuntos relacionados. Com a aplicação das técnicas de revisão sistemática foi possível observar a importância de um bom planejamento e comprovar que esse tipo de revisão não se restringe apenas a isso.

Muitas dificuldades são encontradas na realização de uma RSL, como por exemplo: o grande volume de artigos retornados no início da pesquisa; artigos que tratem do assunto, mas que não correspondam com os objetivos da

RSL; divergências entre os membros em relação a inclusão de um determinado estudo; entre outros. Para realizar uma RSL, deve-se ter preocupações com o tempo de duração, pois a quantidade de estudos retornados, normalmente é muito grande e pode-se demandar muito tempo para a leitura e análise dos estudos; e deve-se seguir o protocolo à risca, para que não haja inclusão de um estudo que não esteja de acordo com os objetivos.

Dentre as oportunidades de melhoria para a RSL realizada, pode-se destacar o aumento na quantidade de fontes de pesquisa, o que poderia reforçar as principais práticas e abordagens encontradas ou até mesmo ocasionar na descoberta de novas abordagens. Como melhoria, também pode-se destacar o tempo em que foi realizado todo o processo da revisão sistemática que teve a duração de onze meses.

Uma dificuldade encontrada ao realizar esta pesquisa foi a grande quantidade de estudos retornados inicialmente em algumas das fontes de pesquisa. Neste caso foi necessário verificar qual a relevância do assunto pesquisado para a fonte e a eficiência da *string* de busca utilizada na mesma. Após essa análise, foram necessários refinamentos em algumas *strings* de busca.

#### **5.4 TRABALHOS FUTUROS**

Esta revisão servirá de insumo para direcionar o desenvolvimento de uma ferramenta educacional para o ensino do processo de medição no contexto de projetos de software. Além de auxiliar no planejamento de outros trabalhos relacionados à Engenharia de Software Baseada em Evidências.

## REFERÊNCIAS

ALBUQUERQUE, A. **Avaliação e Melhoria de Ativos de Processos Organizacionais em Ambientes de Desenvolvimento de Software** - Tese de Doutorado, Universidade Federal do Rio de Janeiro, 2008.

BARRETO, A. S. **Definição e Gerência de Objetivos de Software Alinhados ao Planejamento Estratégico** - Tese de Doutorado, Universidade Federal do Rio de Janeiro, 2011.

BASILI et al. **Goal Question Metric Paradigm, In: Encyclopedia of Software Engineering**, v.2, pp: 527 – 532, 1994.

BASS, L. et al. **Constructing Superior Software, Software Quality Institute Series, Macmillan Technical Publishing**, 1999.

BEECHAN, S. et al. **Motivation in Software Engineering: A systematic literature review**. Information and Software Technology: Elsevier, v. 50, n. 860 -878, 2007.

BIOLCHINI, J., MIAN, P.G., NATALI, A.C., TRAVASSOS, G.H. **Systematic Review in Software Engineering: Relevance and Utility**. Relatório Técnico ES-679/05, PESC – COPPE/UFRJ, 2005.

CAVALCANTI, Ana Regina R.; SANTOS Gleison S.; PERINI Monalesa B. **MEDIÇÃO DE SOFTWARE e Controle Estatístico de Processos**, Brasília, 2012.

CLEMENTI, Juliana Augusto. **Diretrizes Motivacionais para Comunidade de Prática Baseadas na Gamificação. 2014**. 198 f. Dissertação (Mestrado) - Curso de Engenharia e Gestão do 77 Conhecimento, Universidade Federal de Santa Catarina, Florianópolis, 2014.

COSTA, C.S. **Uma abordagem baseada em evidências para o gerenciamento de projetos no desenvolvimento distribuído de software**. Dissertação de Mestrado – Programa de Pós-Graduação em Ciência da Computação - Universidade Federal de Pernambuco, Recife, 2010.

[5] Comentário: Referências acrescentadas

[6] Comentário: Referências acrescentadas

[7] Comentário: Referências acrescentadas

DEMPSEY, J. V., LUCASSEN, B. e RASMUSSEN, K. **The Instructional Gaming Literature: Implications and 99 Sources**. Tech. Report 96-1, College of Education, University of South Alabama, EUA, 1996.

DYBA, T.; DINGSOYR, T.; HANSEN, G.K. **Applying Systematic Reviews to Diverse Study Types: An Experience Report**. First International Symposium on Empirical Software Engineering and Measurement, IEEE, 2007.

DOMÍNGUEZ, A.; SAENZ-DE-NAVARRETE; J. de-MARCOS; L., FERNÁNDEZ-SANZ et al. **Gamifying learning experiences: Practical implications and outcomes**, Computers & Education. 63(April), pages 380-392, 2013.

FENTON, N. e PFLEEGER, S. L. **Software Metrics: A Rigorous and Practical Approach**. International Thomson Computer Press, London, UK, second edition, 1997.

[8] Comentário: Referências acrescentadas

FLORAC, W. A., CARLETON, A. D. **Measuring the Software Process: Statistical Process Control for Software Process Improvement**. Addison Wesley, 1999.

FORMANSKI, F. N. **Aplicabilidade de Gamificação no Contexto Empresarial**. Florianópolis, 2016.

GIBSON, David C. et al. **Digital badges in education. Education And Information Technologies**. p. 1-8. 2013.

GLASS, R. **Facts and Fallacies of Software Engineering**. Addison-Wesley, 2002.

KAPP, K. **The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education**. Pfeiffer, 2012.

[9] Comentário: Referências acrescentadas

KHAN, K.S. et al. **Undertaking Systematic Review of Research on Effectiveness**. CRD Report Number 4 (Second Edition), NHS Centre for Reviews and Dissemination, University of York, UK, 2001.

KITCHENHAM, B. **Procedures for Performing Systematic Reviews**. Joint Technical Report, Software Engineering Group, Keele University, and Empirical Software Eng., Nat'l ICT Australia, 2004.

KITCHENHAM, B. **Guidelines for performing Systematic Literature Reviews in Software Engineering**, Technical Report EBSE-2007-01, Department of Computer Science Keele University, Keele, 2007.

MAFRA, S. ; TRAVASSOS, G. **Estudos Primários e Secundários apoiando a busca por Evidências em Engenharia de Software** - Relatório Técnico: RT-ES-687/06 – Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ – Rio de Janeiro, 2006.

McGARRY et al. **Practical Software Measurement: Objective Information for Decision Makers** - Assison Wesley, 2002.

MUNTEAN, C. I. — **Raising engagement in e-learning through gamification** - The 6th International Conference on Virtual Learning, Cluj-Napoca, Romênia, 2011.

O'REGAN, Gerard. **Introduction to Software Process Improvement**. Undergraduate Topics in Computer Science, New York. p. 23.

OATES, J. B.; CAPPER G. **Using systematic review and evidence-based software engineering with masters students**. International Conference on Evaluation & Assessment in Software Engineering, EASE, 2009.

PRESSMAN, Roger S. **Engenharia de Software**, 6.ed. São Paulo: McGraw Hill, 2006.

PRESSMAN, Roger S.; MAXIM B. **Engenharia de Software: UMA ABORDAGEM PROFISSIONAL**. 8.ed. São Paulo: AMGH, 2016.

SANTOS, G. **Revisão Sistemática**, Mini-Curso. **Simpósio Brasileiro de Qualidade de Software – SBQS 2010**, Belém – PA, 2010.

SEI. **Capability Maturity Model Integration (CMMI) for Development, Version 1.3**. Carnegie Mellon, USA, 2010.

[10] Comentário: Referências acrescentadas

SILVA, L.S.C.V. **Aplicação do Controle Estatístico de Processos na Indústria de Laticínios Lactoplasa: Um estudo de caso** - Dissertação - Florianópolis: Universidade Federal de Santa Catarina, 1999.

[11] Comentário: Referências acrescentadas

SOFTEX. **Melhoria de Processo de Software Brasileiro (MPS.BR)** – Guia de implementação do nível G, 2016.

TRAVASSOS, G., BIOLCHINI J. **Revisões Sistemáticas Aplicadas a Engenharia de Software**. In: XXI SBES - Brazilian Symposium on Software Engineering, João Pessoa, PB, Brasil, 2007.

WITT M. et al. **Gamification of Online Ideas Competitions Insights from an Explorative Case**. INFORMATIK 2011, Berlin, Alemanha, 2011.

ZYDA, M. **From visual simulation to virtual reality to games**. Computer, 38(9), 25-32, 2005.

[12] Comentário: Referências acrescentadas

## APÊNDICE A – PROTOCOLO DE REVISÃO

### Protocolo da Revisão Sistemática

#### Histórico de Revisões

Data	Versão	Descrição	Autor
09/03/17	0.1	Criação do documento e preenchimento até a seção 5.4-Palavras Chaves e Sinônimos.	Lennon
16/03/17	0.2	Preenchimento até a seção 8-Critérios de Exclusão e Inclusão	Lennon
26/03/17	0.3	Preenchimento da seção 9 e 10. E alteração da seção 1 até a 8.	Lennon
28/04/17	0.4	Alteração das Questões secundárias, e critérios de qualidade.	Lennon
06/05/17	0.5	Alteração dos critérios de avaliação da qualidade e estratégia de extração dos dados.	Lennon
10/05/17	1.0	Revisão e Validação do Protocolo	Sandro
18/10/17	1.1	Inserção do critério de exclusão ce.11) e Removidas as bases Google Scholar e Springer Link.	Lennon
18/10/17	1.2	Atualização do método de pesquisa para quatro pesquisadores. E remoção do critério de qualidade 9 que tratava da organização e adesão do mesmo ao padrão formal da língua do artigo.	Lennon
18/10/17	1.3	Atualização da escala do critério 4.a) para contemplar o caso de ferramentas não detalhadas validadas ou não pelos usuário.	Lennon
29/10/17	1.4	Organização das questões secundárias de pesquisa por questões gerais, questões específicas de MPS e questões específicas de Medição de software.	Lennon
29/11/17	1.5	Substituição da ISSO 15939 por Modelo ou norma ou paradigma que a proposta de ensino abordou.	Lennon
30/11/17	1.6	Revisão e publicação da edição definitiva do protocolo	Lennon
01/12/17	1.7	Fator Comparação atualizado e Inserção das Strings de busca para cada Base	Lennon
01/12/17	1.8	Verificação do protocolo	Rafael e

			João
01/12/17	2.0	Validação da versão final do protocolo	Sandro

### Contexto

Esta revisão sistemática é parte de uma tese de doutorado vinculado ao projeto SPIDER com o objetivo de reunir conhecimentos a respeito da implementação e avaliação de propostas de jogos sérios e de gamificação para o ensino do Processo de medição e processos relacionados com programas de melhoria de processos de software. Para assim, criar uma ferramenta educacional para o ensino do processo de medição que faça uso de boas práticas presentes nos trabalhos pesquisados.

O processo de medição de software tem o objetivo de coletar, armazenar, analisar e relatar os dados relativos aos produtos desenvolvidos e aos processos implementados em uma organização, de forma a apoiar objetivos organizacionais. Mesmo o processo de medição tendo tamanha importância em um Programa de Melhoria de Processos de Software (PMPS), segundo a literatura, a indústria de software tem se mostrado tímida na aplicação de programas de medição eficientes. Isto se dá, pois, muitos gerentes de software e profissionais, incluindo acadêmicos em engenharia de software e ciência da computação, parecem ter pouco ou nenhum conhecimento prático sobre este assunto.

Neste contexto, esta revisão sistemática será um instrumento metodológico para apoiar a criação de uma ferramenta educacional que visa ensinar o processo de medição de maneira prática e inteligível.

### Objetivos

Esta revisão sistemática terá por objetivo de identificar diferentes abordagens no ensino do processo de medição e do ensino de processos relacionados a PMPS por propostas de gamificação e jogos sérios. Ao identificar estas abordagens, esta revisão evidenciará as dinâmicas, mecânicas e componentes de jogos presentes nestas abordagens com intuito de direcionar o desenvolvimento da ferramenta educacional proposta por esta tese de doutorado. Desta forma, têm-se a seguinte estrutura para o objetivo, conforme proposto em (SANTOS, 2010):

- **Analisar:** relatos de experiência e publicações científicas através de um estudo baseado em revisão sistemática;
- **Com o propósito de:** identificar as abordagens na utilização de dinâmicas, mecânicas e componentes de jogos no ensino do processo de medição e no ensino de PMPS visando identificar as necessidades de informação, indicadores e medidas mais utilizadas em PMPS com ênfase no processo de medição de software;
- **Com relação ao:** emprego de proposta de gamificação e jogo sérios aplicada ao processo de medição dentro de Programas de

MPS, visando o ensino ou execução das atividades de Medição em organizações de desenvolvimento de software ou cursos que ministrem engenharia de software;

- **Do ponto de vista:** dos pesquisadores;
- **No contexto:** acadêmico e industrial.

## Referências

Kitchenham, B. (2004) Procedures for performing systematic reviews. Keele, UK, Keele University, v. 33, n. 2004, p. 1–26.

KITCHENHAM, B. (2005) “Guidelines for performing Systematic Literature Reviews in Software Engineering”. Vol 2.3 EBSE Technical Report, EBSE-2007-01, 2007.

MAFRA, S., TRAVASSOS, G. (2005) “Técnicas de Leitura de Software: Uma Revisão Sistemática”. XIX Simpósio Brasileiro de Engenharia de Software (SBES 2005).

SANTOS, G. S. (2010) “Revisão Sistemática, Mini-Curso”. Simpósio Brasileiro de Qualidade de Software – SBQS 2010, Belém – PA.

## Formulação da Pergunta

As questões levantadas foram organizadas conforme a estrutura *Population, Intervention, Context, Outcomes, Comparison* (PICOC), recomendada por Kitchenham (2007):

1. **Qual o estado da arte da pesquisa sobre aplicação ou ensino de programas de melhorias de processos de software (PMPS) utilizando jogos sérios ou gamificação?**
  - a. **População (P):** Organizações de Software e Instituições de ensino;
  - b. **Intervenção (I):** Abordagem utilizada para aplicar ou ensinar o processo de melhoria de software;
  - c. **Contexto (C):** Este trabalho visa explorar a comparação dos artigos voltados tanto para o ambiente industrial quanto para o acadêmico. E terá quatro pesquisadores executando as atividades desta revisão sistemática que é considerada de grande escala. Pois, sua duração será de aproximadamente três trimestres;
  - d. **Resultados (O):** Objetivos das abordagens por gamificação ou jogo sério, dinâmicas, mecânicas e componentes de jogos presentes nas propostas discutidas e a eficiência no ensino ou na aplicação prática do processo de melhoria de software ao utilizar estas propostas.
  - e. **Comparação (C):** Não se aplica a este trabalho.

**1. Qual o estado da arte da pesquisa sobre aplicação ou ensino de medição de software utilizando jogos sérios ou gamificação?**

**a. População (P):** Organizações de Software e Instituições de ensino;

**b. Intervenção (I):** Abordagem utilizada para aplicar ou ensinar o processo de medição de software;

**c. Contexto (C):** Este trabalho visa explorar a comparação dos artigos voltados tanto para o ambiente industrial quanto para o acadêmico. E terá quatro pesquisadores executando as atividades desta revisão sistemática que é considerada de grande escala. Pois, sua duração será de aproximadamente três trimestres;

**d. Resultados (O):** Objetivos das abordagens por gamificação ou jogo sério, dinâmicas, mecânicas e componentes de jogos presentes nas propostas discutidas e a eficiência no ensino ou na aplicação prática do processo de medição ao utilizar estas propostas.

**e. Comparação (C):** Não se aplica a este trabalho.

**Questões Secundárias**

Um conjunto de questões secundárias referentes à primeira e a segunda questão principal foram estabelecidas, questões essas para serem respondidas durante a fase de extração de informações. Tais questões têm objetivo de esclarecer detalhes importantes que esta revisão procura identificar, para colaborar com o projeto onde esta se insere.

**Questões Secundárias Gerais(QSG)**

- **QSG1.** Em quais contextos foram aplicados a proposta de gamificação ou Jogo sério? (Ensino, Trabalho, ...)
- **QSG2.** Quais as limitações relatadas no uso da gamificação ou Jogo Sérios para o ensino?
- **QSG3.** Quais métodos de pesquisa foram empregados na validação da proposta de gamificação ou jogo sério?
- **QSG4.** Quais os elementos de jogos foram utilizados na proposta de gamificação ou jogo?
- **QSG5.** Quais as mecânicas de jogos foram utilizadas na proposta de gamificação ou jogo?
- **QSG6.** Quais as dinâmicas de jogos foram utilizadas na proposta de gamificação ou jogo sério?
- **QSG7.** O jogo sério ou a proposta de gamificação são considerados apropriados em termos e relevância de conteúdo, correteza, grau de dificuldade, método de ensino, duração no contexto o qual é intencido e a repetibilidade do jogo? O jogo é considerado engajador? Quais seus pontos fracos e pontos fortes?

- **QSG8.** Como foi a eficiência no aprendizado ou na aplicação do programa de melhorias de processos de software com a utilização da proposta de gamificação ou jogo sério ao comparar com o meio tradicional de aprendizado ou aplicação?

#### **Questões Secundárias Específicas de Melhoria de Processos de Software(QSMPS)**

- **QSMPS1.** Em quais processos foram aplicados a proposta de gamificação ou Jogo sério na área de MPS? (Medição, Coleta de Requisitos, ...)
- **QSMPS2.** Como educadores ou a indústria podem se beneficiar na aplicação ou ensino de programas de melhorias de processos de software (MPS) através de propostas de gamificação ou jogos sérios?

#### **Questões Secundárias Específicas de Medição de Software (QSMS)**

A seguir será apresentado um conjunto de questões secundárias referentes à segunda questão principal:

- **QSMS1.** A proposta apresentada foi baseada em algum modelo ou norma ou paradigma? Se sim, qual?
- **QSMS2.** Quais métricas foram cobertas pela proposta de gamificação e jogo sérios?
- **QSMS3.** Quais tópicos de medição foram cobertos pelas propostas? (Coletar, Armazenar, Analisar e Relatar)
- **QSMS4.** Como educadores ou a indústria podem se beneficiar no ensino ou aplicação de programas de medição de software através de propostas de gamificação ou jogos sérios?

#### **Escopo da Pesquisa**

Foram definidos critérios para seleção das fontes de pesquisa e restrições para garantir a viabilidade da pesquisa e com isso limitar um escopo de pesquisa.

#### **Métodos de Busca nas Fontes**

Para seleção das fontes de pesquisa, foram definidos os seguintes critérios:

- Disponibilidade para consultas web.
- Disponibilidade para busca de artigos através do domínio da UFPA.
- Disponibilidade de artigos na íntegra através do domínio da UFPA ou a partir da utilização da *engine* de busca Google ou Google Scholar ou Portal Capes.
- Disponibilidade de artigos em inglês ou português.
- Que possuam máquinas de busca

#### **Palavras-Chave e Sinônimos**

A partir das questões de pesquisa, palavras-chave foram identificadas em acordo com a estrutura População, Intervenção e Resultados para a

posterior formulação da *string* de busca. Segue a listagem de palavras-chave definidas:

Para a primeira questão principal de pesquisa:

- POPULAÇÃO
  - Inglês: *Project, Development, Organization, Enterprise, Company, Industry, Institute, Research Group, Technology Center;*
- INTERVENÇÃO
  - Inglês: *Process, Improvement, SPI;*
- CONTEXTO
  - *Learning, Teaching, Education, Training, Practice, Application;*
- RESULTADOS
  - Inglês: *Gamification, Game, Serious Game, Funware, Game Elements, Game Mechanics, Game Component, Game factor, Game aspect;*

Para a segunda questão principal de pesquisa:

- POPULAÇÃO
  - Inglês: *Project, Development, Organization, Enterprise, Company, Industry, Institute, Research Group, Technology Center;*
- INTERVENÇÃO
  - Inglês: *Process, Measuring, Software, Measurement, Metrics, Metrology;*
- CONTEXTO
  - *Learning, Teaching, Education, Training, Practice, Application;*
- RESULTADOS
  - Inglês: *Gamification, Game, Serious Game, Funware, Game Elements, Game Mechanics, Game Component, Game factor, Game aspect;*

### **Strings de Busca**

- **Para Science Direct Q1:**
  - **Inglês:** *title-abstr-key(Software AND (Project OR Development OR Organization OR Enterprise OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification));*
- **Para Science Direct Q2:**
  - **Inglês:** *title-abstr-key(Software AND (Project OR Development OR Organization OR Enterprise OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*) AND (Gam\* OR Funware OR Ludification));*
- **Para IEEE Q1:**
  - **Inglês:** *Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR*

*Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification);*

- **Para IEEE Q2:**

- **Inglês:** *(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*)) AND (Gam\* OR Funware OR Ludification));*

- **Para ACM (Association for Computing Machinery) Q1:**

- **Inglês:** *recordAbstract(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification));*

- **Para ACM (Association for Computing Machinery) Q2:**

- **Inglês:** *recordAbstract(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*)) AND (Gam\* OR Funware OR Ludification));*

- **Para EI compendex Q1:**

- **Inglês:** *(((((Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification)) WN KY) AND ({english} WN LA)));*

- **Para EI Compendex Q2:**

- **Inglês:** *(((((Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*)) AND (Gam\* OR Funware OR Ludification)) WN KY) AND ({english} WN LA)));*

- **Para Web of Science Q1:**

- **Inglês:** *(TS=(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification)));*

- **Para Web of Science Q2:**

- **Inglês:** *(TS=(Software AND (Project OR Development OR Organization OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*)) AND (Gam\* OR Funware OR Ludification));*

- **Para Scopus Q1:**

- **Inglês:** *((TITLE-ABS-KEY(Software AND (Project OR Development OR Organization OR Academy OR Industry OR*

*Learning OR Teaching OR Education OR Training OR Simulation) AND (Process OR Improvement) AND (Gam\* OR Funware OR Ludification))) AND (PUBYEAR > 1999)) AND ( LIMIT-TO ( LANGUAGE,"English" ) );*

- **Para Scopus Q2:**

- **Inglês:** *(TITLE-ABS-KEY(Software AND (Project OR Development OR Organization OR Enterprise OR Academy OR Industry OR Learning OR Teaching OR Education OR Training OR Simulation) AND (Measu\* OR Metr\*)) AND (Gam\* OR Funware OR Ludification))) AND ( ( PUBYEAR > 1999 ) ) AND ( LIMIT-TO ( LANGUAGE,"English" ) );*

### **Listagem de Fontes OK**

- IEEE Xplore
- ACM DLL
- Science Direct
- Scopus
- Isi of knowledge (Web of Science)
- El compendex

### **Tipos de Artigos**

Foram considerados os artigos, que tratem de: estudos experimentais, relatos de experiência, revisões sistemáticas, relatórios técnicos, levantamento bibliográfico, mapa sistemático de estudo e estudos de caso.

### **Idioma dos Artigos**

Foram considerados artigos escritos em português e inglês, a primeira por ser importante considerar pesquisas de âmbito nacional, dada a relevância do Programa MPS.BR para o estudo, e a segunda para expandir a abrangência da pesquisa, uma vez que é a língua definida como padrão na grande maioria dos periódicos e conferências internacionais.

### **Critérios de Inclusão e Exclusão de Estudos Primários**

Os critérios de exclusão dos artigos:

1. Artigos que não estejam disponíveis livremente para consulta ou download (em versão completa) aos alunos da universidade federal do Pará;
2. Artigos que não foram encontrados ao utilizar as strings de busca nos indexadores de conteúdo;
3. Artigos não presentes nas bases listadas;
4. Artigos que violem os princípios de alguma das bases listadas. Como artigos que plagiam outros artigos e afins.
5. Artigos repetidos (em mais de uma fonte de busca) terão apenas sua primeira ocorrência considerada;

6. Artigos duplicados terão apenas sua versão mais recente ou a mais completa considerada, salvo casos em que haja informações complementares;
7. Estudos enquadrados como resumos, keynote speeches, cursos, tutoriais, workshops, posters, e afins;
8. Artigos que não estiverem inseridos no contexto de ensino ou aplicação de programas de melhoria de software ou programas de medição através do uso de gamificação ou jogo sério;
9. Artigos que não estiverem em inglês ou em português;
10. Artigos irrelevantes para a pesquisa de acordo com as questões levantadas;

Os critérios de inclusão dos artigos se baseiam em:

1. Artigos que apresentem relatos de experiência na indústria, ou na academia, ou pesquisas de caráter experimental ou teórico, que apresentem exemplos de aplicação, descrição de experimentos ou casos reais de uso de abordagens de ensino ou aplicação de programas de melhoria de software ou programas de medição através do uso de gamificação ou jogo sério;

### **Critérios de Qualidade de Estudos Primários**

#### **1. Introdução**

- a. Os objetivos ou questões do estudo são claramente definidos? E a problemática abordada na pesquisa está descrita claramente (incluindo justificativas para a realização do estudo)?
- b. O tipo de estudo está definido claramente?

#### **2. Desenvolvimento**

- a. Existe uma clara descrição do contexto no qual a pesquisa foi realizada?
- b. O trabalho é adequadamente referenciado (apresenta trabalhos relacionados ou semelhantes e baseia-se em modelos e teorias da literatura)?

#### **2. Conclusão**

- a. O estudo relata de forma clara e não ambígua os resultados?
- b. Os objetivos ou questões do estudo são alcançados?

#### **2. Critérios para a Questão de Investigação**

- a. O estudo apresenta primária ou secundariamente uma abordagem ou ferramenta de ensino ou de aplicação de programas de melhoria de software ou programas de medição através do uso de gamificação ou jogo sério?

#### **2. Critério Específico para estudos Experimentais**

- a. Existe um método ou um conjunto de métodos descrito para a realização do estudo?

#### **2. Critério Específico para estudos Teóricos**

- a. Existe um processo não tendencioso na escolha dos estudos?

#### **2. Critério Específico para Revisões Sistemáticas**

- a. Existe um protocolo rigoroso, descrito e seguido?

#### **2. Critério Específico para Relato de Experiência Industrial**

- a. Existe uma descrição sobre a(s) organização(ões) onde foi conduzido o estudo?

## Processo de Seleção de Estudos Primários

Para a execução desta revisão sistemática foram seguidos os seguintes passos:

- Quatro pesquisadores (um doutorando, um mestrando e dois graduandos) foram os participantes na condução das atividades desta revisão sistemática. E como primeiro passo, estes irão realizar o teste (verificação e validação) das strings de busca com o intuito de averiguar sua acurácia no retorno dos artigos primários e também, assim, poder criar múltiplas instâncias destas strings adaptadas para cada base de dados;
- Após os testes das strings de busca, os quatro pesquisadores irão realizar a busca nos indexadores de conteúdo científico por meio do domínio da Universidade Federal do Pará para encontrar os possíveis artigos primários;
- Os alunos lerão os títulos e resumos dos artigos retornados pela string de busca. Para assim, criar uma lista com os possíveis artigos primários;
- Os artigos presentes na lista de possíveis artigos primários, terão seus títulos, resumos, introduções e conclusões lidos. E neste momento os alunos aplicarão os critérios de inclusão e exclusão para descartar os falsos positivos. E assim, criar uma lista dos artigos primários e uma lista dos artigos excluídos;
- As listas contendo os artigos primários foram comparadas e unificadas. Caso haja discordância entre os alunos na inclusão ou exclusão de um artigo, o mesmo, deve ser incluído;
- Os estudos presentes na lista devem ser lidos em sua totalidade e os critérios de qualidade devem ser aplicados. Assim, os artigos que obtiverem uma determinada pontuação de qualidade foram compilados na lista de artigos primários com qualidade verificada;
- Após, todos os artigos presentes na lista gerada anteriormente passarão pela estratégia de extração de dados;
- Ademais, todos os documentos e procedimentos foram validados através de reuniões com o coordenador do Projeto SPIDER, que possui uma gama de artigos publicados em periódicos e conferências sobre o tema de medição e também tem orientado várias dissertações de mestrado e trabalhos de conclusão de curso que versam sobre o tema de medição. Além disto, possui experiência prática na implementação do processo de medição ao ter prestado consultoria a diversas empresas brasileiras sobre este tema, sendo certificado como avaliador, consultor-implementador e instrutor oficial dos modelos de melhoria de processo e produto de software, CMMI, MPS.BR, Certics, Medepros, QPS. Para mais, o Prof. Dr. Sandro Ronaldo Bezerra Oliveira, também é o orientador do aluno que conduz esta tese de doutorado.

## Avaliação da Qualidade dos Estudos Primários

- **Concordo totalmente (4):** deve ser concedido no caso em que o trabalho apresente no texto os critérios que atendam totalmente a questão;
- **Concordo parcialmente (3):** deve ser concedido no caso em que o trabalho atenda parcialmente aos critérios da questão;
- **Neutro (2):** deve ser concedido no caso em que o trabalho não deixa claro se atende ou não a questão;
- **Discordo parcialmente (1):** deve ser concedido no caso em que os critérios contidos na questão não são atendidos pelo trabalho avaliado;
- **Discordo totalmente (0):** deve ser concedido no caso em que não existe nada no trabalho que atenda aos critérios da questão.

Escala por Critério	
Critério	Escala
1a.	4 – Define claramente os objetivos e justifica o estudo. 3 - Define claramente o estudo, porém a justificativa não é clara ou não justifica o estudo. 2 – Não Define os objetivos, mas justifica o estudo. 1 - A definição e a justificativa do estudo não são claras. 0 - Não define os objetivos e nem justifica o estudo.
1b.	4 – Define o tipo de estudo, referenciando na literatura a metodologia. 3 – Define o tipo de estudo, porém sem referenciar a metodologia. 2 – Não define o tipo de estudo. É possível inferir facilmente. 1 – Não define o tipo de estudo. É possível inferir com dificuldade. 0 – Não é possível inferir o tipo de estudo.
2a.	4 – Define claramente uma seção com o contexto da pesquisa. 3 – O contexto da pesquisa está incluído em uma seção não exclusiva. 2 – O contexto da pesquisa está disperso ao longo do texto. 1 – O contexto da pesquisa está disperso e é insubstancial. 0 – O contexto da pesquisa não é abordado.

2b.	<p>4 – O texto apresenta uma seção de trabalhos relacionados.</p> <p>3 – O texto apresenta trabalhos relacionados em uma seção não exclusiva.</p> <p>2 – O texto apresenta trabalhos relacionados dispersos ao longo do texto.</p> <p>1 – O texto não apresenta trabalhos relacionados, mas se apoia na literatura.</p> <p>0 – O texto não apresenta trabalhos relacionados nem se apoia na literatura.</p>
3a.	<p>4 – Resultados são claramente apresentados na seção de conclusão.</p> <p>3 – Resultados são claramente referenciados na seção de conclusão.</p> <p>2 – Resultados apresentados na conclusão não são claros.</p> <p>1 – Resultados referenciados na conclusão não são claros.</p> <p>0 – Não são apresentados resultados.</p>
3b.	<p>4 – Os resultados estão totalmente aderentes ao objetivo do estudo.</p> <p>3 – Os resultados estão aderentes ao objetivo do estudo, no entanto o autor faz ressalvas.</p> <p>2 – Os resultados são parcialmente aderentes ao objetivo do estudo.</p> <p>1 – Os resultados não estão aderentes ao objetivo do estudo.</p> <p>0 – Não é alcançado nenhum resultado.</p>
4a.	<p>4 – Apresenta uma ferramenta ou abordagem de maneira detalhada e validada com usuários.</p> <p>3 – Apresenta uma ferramenta ou abordagem de maneira detalhada, porém não validada.</p> <p>2 – Apresenta uma ferramenta ou abordagem, porém sem detalhar a mesma. Contudo validando a mesma com os usuários.</p> <p>1 – Apresenta uma ferramenta ou abordagem, porém sem detalhar e validar a mesma com os usuários.</p> <p>0 – Apenas referência uma ferramenta ou não apresenta uma ferramenta ou abordagem.</p>

5a.	<p>4 – O método de experimento é definido e referenciado claramente.</p> <p>3 – O método de experimento é definido claramente.</p> <p>2 – O método de experimento é citado.</p> <p>1 – O método de experimento não é citado, porém é possível inferir.</p> <p>0 – Não é possível inferir o método de experimento.</p>
6a.	<p>4 – O texto descreve critérios para a escolha dos estudos.</p> <p>3 – O texto não descreve critérios para a escolha dos estudos, porém apresenta estudos que discordam do estudo apresentado.</p> <p>2 – O texto descreve apenas estudos aderentes ao estudo apresentado.</p> <p>1 – O texto descreve estudos insuficientes.</p> <p>0 – O texto não descreve estudos base.</p>
7a.	<p>4 – O protocolo de revisão é apresentado, descrito e seguido.</p> <p>3 – O protocolo de revisão é apresentado e descrito, porém há evidências de que não foi seguido adequadamente.</p> <p>2 – O protocolo de revisão não foi suficientemente descrito.</p> <p>1 – O protocolo de revisão apenas foi citado ao longo do texto.</p> <p>0 – Não há um protocolo de revisão.</p>
8a.	<p>4 – A área de atuação, tamanho e origem da organização são informados.</p> <p>3 – Apenas duas das características do item 4 são informadas.</p> <p>2 – Apenas uma das características do item 4 é informada.</p> <p>1 – Nenhuma das características do item 4 é informada.</p> <p>0 – O estudo não foi conduzido em uma ou mais organizações.</p>

Fonte: Elaborado pelo autor.

Deve-se então, dada à soma das pontuações, enquadrá-los em um dos cinco níveis de qualidade classificados por Beecham (2007), tais como, apresentados na Tabela a seguir.

**Tabela 1. Níveis de Qualidade**

Faixa de Notas	Avaliação
Excelente	>86%
Muito Boa	66%-85%
Boa	46%-65%
Média	26%-45%
Baixa	< 26%

**Fonte: Beecham (2007).**

### **Estratégia de Extração de Informações**

Para os dados bibliométricos foram coletados os dados:

Título, autores, ano, país onde foi desenvolvida a pesquisa, indexador base (IEEE, ACM, etc), veículo de publicação (Conferência, Jornal, etc), tipo de estudo, tipo de estudo experimental, objetivos da pesquisa, resultados e ferramentas;

Para a primeira e segunda questões principais foram coletados os dados gerais a seguir:

### **Questões Secundárias Gerais (QSG)**

- **QSG1.** Em quais contextos foram aplicados a proposta de gamificação ou Jogo sério? (Ensino, Trabalho, ...)
- **QSG2.** Quais as limitações relatadas no uso da gamificação ou Jogo Sérios para o ensino?
- **QSG3.** Quais métodos de pesquisa foram empregados na validação da proposta de gamificação ou jogo sério?
- **QSG4.** Quais os elementos de jogos foram utilizados na proposta de gamificação ou jogo?
- **QSG5.** Quais as mecânicas de jogos foram utilizadas na proposta de gamificação ou jogo?
- **QSG6.** Quais as dinâmicas de jogos foram utilizadas na proposta de gamificação ou jogo sério?
- **QSG7.** Quais os pontos fracos e pontos fortes do trabalho?
- **QSG8.** Como foi a eficiência no aprendizado ou na aplicação do programa de melhorias de processos de software com a utilização da

proposta de gamificação ou jogo sério ao comparar com o meio tradicional de aprendizado ou aplicação?

Enquanto que para a primeira questão principal, foram coletados os seguintes dados específicos a seguir:

- **QSMPS1.** Em quais processos foram aplicados o programa de gamificação ou Jogo sério na área de MPS? (Medição, Coleta de Requisitos, ...)
- **QSMPS2.** Como educadores ou a indústria podem se beneficiar na aplicação ou ensino de programas de melhorias de processos de software (MPS) através de propostas de gamificação ou jogos sérios?

Já para a segunda questão principal, foram coletados os seguintes dados específicos a seguir:

- **QSMS1.** A proposta apresentada foi baseada em algum modelo ou norma ou paradigma? Se sim, qual?
- **QSMS2.** Quais métricas foram cobertas pela proposta de gamificação e jogo sérios?
- **QSMS3.** Quais tópicos de medição foram cobertos pelas propostas? (Coletar, Armazenar, Analisar e Relatar)
- **QSMS4.** Como educadores ou a indústria podem se beneficiar no ensino ou aplicação de programas de medição de software através de propostas de gamificação ou jogos sérios?

### **Sumarização dos Resultados.**

Essa etapa consiste em organizar os dados extraídos para apresentação dos gráficos que serviram como panorama geral e base para futuras análises. Entre os gráficos gerados teremos:

Para a bibliometria

- Gráficos com os autores, ano, país, indexador base, veículo de publicação, tipo de estudo e tipo de estudo experimental;

Para as questões principais

- Os contextos onde foram aplicados os programas de gamificação ou Jogo sério (Educação ou treinamento ou trabalho);
- Limitações encontradas durante a utilização das propostas de gamificação e jogo sério;
- Métodos de pesquisa empregados nos artigos
- Dinâmicas, mecânicas e componentes de jogos aplicados nas propostas de G/JS;
- Pontos Fracos e fortes dos trabalhos analisados;
- Sumarização dos resultados dos testes com os usuários das ferramentas apresentadas;

Para a primeira questão de pesquisa, especificamente foram utilizados os gráficos para representar:

- Áreas de aplicação das propostas de G/JS dentro de um PMPS (Medição, Coleta de requisitos, ...);

os  
Para a segunda questão de pesquisa, especificamente foram utilizados gráficos para representar:

- Métricas cobertas pelas propostas de G/JS;
- Tópicos de medição cobertos pelas propostas de G/JS;
- Normas, paradigmas e modelos cobertos pelas propostas de G/JS.

## APÊNDICE B – Estudos Primários e Secundários

ID	Título (Artigos Primários)	Ano de Publicação	Fonte
EP1	Teaching ISO/IEC 12207 software lifecycle processes: A serious game approach	2016	Science Direct
EP2	Software Engineering Management Education through Game Design Patterns	2016	Science Direct
EP3	A systematic literature review on serious games evaluation: An application to software project management	2015	Science Direct
EP4	Gamification in software engineering – A systematic mapping	2015	Science Direct
EP5	Towards a New Approach to Supporting Top Managers in SPI\ Organizational Change Management	2013	Science Direct
EP6	An experimental card game for teaching software engineering processes	2005	Science Direct
EP7	CrowdSummarizer: Automated Generation of Code Summaries for Java Programs through Crowdsourcing	2017	IEEE
EP8	Gamifying the learning of design patterns in software engineering education	2016	IEEE
EP9	Agile Workbench: Tying People, Process, and Tools in Distributed Agile Delivery	2016	IEEE
EP10	An agile software engineering process improvement game	2016	IEEE
EP11	A Gamification Approach for Distributed Agile Delivery	2016	IEEE
EP12	TestEG #x2014; A computational game for teaching of software testing	2016	IEEE
EP13	Experimental Evaluation of a Serious Game for Teaching Software Process Modeling	2015	IEEE
EP14	ARMI 2.0: An online risk management simulation	2015	IEEE
EP15	InspectorX: A game for software inspection training and learning	2014	IEEE
EP16	Crowdsourcing and Gamification of Enterprise Meeting Software Quality	2014	IEEE
EP17	An educational simulation model derived from academic and industrial experiences	2013	IEEE
EP18	It was a bit of a race: Gamification of version control	2012	IEEE
EP19	Challenges and issues in the development of a Software Engineering simulation game	2012	IEEE
EP20	Intelligent Behavior Simulation Module for Software Process Elements	2011	IEEE
EP21	Turning Real-World Software Development into a Game	2011	IEEE
EP22	hACMEgame: A Tool for Teaching Software	2009	IEEE

	Security		
EP23	Embracing Business Context in Pedagogical Simulation Games--A Case with Process Disciplined Project Management	2008	IEEE
EP24	A Constructivist Approach to Teaching Software Processes	2007	IEEE
EP25	It's All in the Game: Teaching Software Process Concepts	2005	IEEE
EP26	Design and Evaluation of an Educational Software Process Simulation Environment and Associated Model	2005	IEEE
EP27	An experimental card game for teaching software engineering	2003	IEEE
EP28	Problems and Programmers: an educational software engineering card game	2003	IEEE
EP29	Teaching University Students Kanban with a Collaborative Board Game	2016	IEEE
EP30	Protection Poker: The New Software Security "Game";	2010	IEEE
EP31	TREG Usability Tests: Evaluating a Training Game in Second Life	2010	IEEE
EP32	A Game of Refactoring: Studying the Impact of Gamification in Software Refactoring	2016	ACM
EP33	CLEVER: A Trivia and Strategy Game for Enterprise Knowledge Learning	2016	ACM
EP34	Animating Organizational Patterns	2015	ACM
EP35	Process Simulation for Software Engineering Education	2015	ACM
EP36	Applying Gamification in the Context of Knowledge Management	2015	ACM
EP37	Gamifying Software Engineering Tasks Based on Cognitive Principles: The Case of Code Review	2015	ACM
EP38	QuoDocs: Improving Developer Engagement in Software Documentation Through Gamification	2014	ACM
EP39	Time's Up: Studying Leaderboards for Engaging Punctual Behaviour	2013	ACM
EP40	Understanding Gamification Mechanisms for Software Development	2013	ACM
EP41	Towards Improving Bug Tracking Systems with Game Mechanisms	2012	ACM
EP42	HALO (Highly Addictive, Socially Optimized) Software Engineering	2011	ACM
EP43	CUTA4UML: Bridging the Gap Between Informal and Formal Requirements for Dynamic System Aspects	2010	ACM
EP44	Engendering an Empathy for Software Engineering	2005	ACM
EP45	SimSE: An Educational Simulation Game for Teaching the Software Engineering Process	2004	ACM

EP46	Crowd-centric Requirements Engineering	2014	ACM
EP47	Gamification at scraim	2017	EL
EP48	Designing game strategies: An analysis from knowledge management in software development contexts	2017	EL
EP49	A scrumban integrated gamification approach to guide software process improvement: A Turkish case study	2016	EL
EP50	Gamification behind the scenes: Designing a software engineering course	2016	EL
EP51	Bridging the gap between SPI and SMES in educational settings: A learning tool supporting ISO/IEC 29110	2016	EL
EP52	Process improving by playing: Implementing best practices through business games	2016	EL
EP53	Gamifying the onboarding process for novice software practitioners	2016	EL
EP54	A learning tool for the ISO/IEC 29110 standard: Understanding the project management of basic profile	2016	EL
EP55	Coverage of ISO/IEC 12207 software lifecycle process by a simulation-based serious game	2016	EL
EP56	A gamification approach to improve the software development process by exploring the personality of software practitioners	2016	EL
EP57	Gamification and functional prototyping to support motivation towards software process improvement	2016	EL
EP58	Gamification proposal for a software engineering risk management course	2015	EL
EP59	Gamifying software development environments using cognitive principles	2015	EL
EP60	Taking seriously software projects inception through games	2015	EL
EP61	Gamiware: A gamification platform for software process improvement	2015	EL
EP62	Process improvement with retrospective gaming in agile software development	2015	EL
EP63	Designing games for improving the software development process	2015	EL
EP64	Assessment proposal of teaching and learning strategies in software process improvement	2015	EL
EP65	Learning risk management in software projects with a serious game based on intelligent agents and fuzzy systems	2013	EL
EP66	A serious game for supporting training in risk management through project-based learning	2012	EL
EP67	An approach to assess knowledge and skills in risk management through project-based learning	2012	EL
EP68	E-SPM: An online software project management	2012	EL

	game		
EP69	Empirical evaluation of an educational game on software measurement	2009	EL
EP70	Software process modeling for an educational software engineering simulation game	2005	EL
EP71	SimSE: An interactive simulation game for software engineering education	2004	EL
EP72	Using competition to build a stronger team	2004	EL
EP73	Process-Oriented interactive simulation of software acquisition projects	2002	EL
EP74	Version control system gamification: A proposal to encourage the engagement of developers to collaborate in software projects	2014	EL
EP75	Gamification and Human Factors in Quality Management Systems: Mapping from Octalysis Framework to ISO 10018	2016	WOK
EP76	Agile Retrospective Games for Different Team Development Phases	2016	WOK
EP77	Towards a Gamification Framework for Software Process Improvement Initiatives: Construction and Validation	2016	WOK
EP78	Working and Playing with Scrum	2015	WOK
EP79	Gamification as a Disruptive Factor in Software Process Improvement Initiatives	2014	WOK
EP80	CutIT: A Game for Teaching Process Improvement in Software Engineering	2013	WOK
EP81	The Gamification of SPICE	2012	WOK
EP82	Communication and traceability game: a way to improve requirements elicitation process teaching	2010	WOK
EP83	Scrum in software engineering courses: An outline of the literature	2015	SCOPUS
EP84	Teaching software engineering through a collaborative game	2014	SCOPUS
EP85	A systematic survey of games used for software engineering education	2011	SCOPUS
EP86	Requirements elicitation with adapted CUTA cards: First experiences with business process analysis	2010	SCOPUS

ID	Título (Artigos Secundários)	Ano de Publicação	Fonte
ES1	Integration of RPG\ use and ELC\ foundation to examine students' learning for practice	2016	Science Direct
ES2	GSDgame: A Serious Game for the Acquisition of the Competencies Needed in GSD	2016	IEEE
ES3	Adopting collaborative games into Open Kanban	2016	IEEE
ES4	A perspective on blending programming environments and games: Beyond points, badges, and leaderboards	2016	IEEE
ES5	Acceptance Requirements and Their Gamification Solutions	2016	IEEE
ES6	Discovering the essence of Software Engineering an integrated game-based approach based on the SEMAT Essence specification	2015	IEEE
ES7	Crowdsourcing Code and Process via Code Hunt	2015	IEEE
ES8	Transparently Teaching in the Context of Game-based Learning: the Case of SimulES-W	2015	IEEE
ES9	Game Design Techniques for Software Engineering Management Education	2015	IEEE
ES10	Serious games, gamification and game engines to support framework activities in engineering: Case studies, analysis, classifications and outcomes	2014	IEEE
ES11	Reinforcing Software Engineering Learning through Provenance	2014	IEEE
ES12	Minecraft-based preparatory training for software development project	2014	IEEE
ES13	Gamification of software engineering curriculum	2014	IEEE
ES14	A systematic mapping study on practical approaches to teaching software engineering	2014	IEEE
ES15	Enhancing software engineering student team engagement in a high-intensity extreme programming course using gamification	2014	IEEE
ES16	Towards recognizing and rewarding efficient developer work patterns	2013	IEEE
ES17	GAME: Governance for Agile Management of Enterprises: A Management Model for Agile Governance	2013	IEEE
ES18	ScrumTutor: A web-based interactive tutorial for Scrum Software development	2013	IEEE
ES19	Learning software engineering processes through playing games	2012	IEEE
ES20	A Role-Playing Game for a Software Engineering Lab: Developing a Product Line	2012	IEEE
ES21	UbiRE: A game for teaching requirements in the context of ubiquitous systems	2012	IEEE

ES22	Teaching software engineering with SimUES-W	2011	IEEE
ES23	Mission to Mars: An agile release planning game	2011	IEEE
ES24	Simulating the software engineering interview process using a decision-based serious computer game	2011	IEEE
ES25	An agile boot camp: Using a LEGO #x00AE;- based active game to ground agile development principles	2011	IEEE
ES26	Introducing object oriented design patterns through a puzzle-based serious computer game	2011	IEEE
ES27	PlayScrum - A Card Game to Learn the Scrum Agile Method	2010	IEEE
ES28	Work in progress #x2014; An investigation of varied game-based learning systems in engineering education	2010	IEEE
ES29	A conceptual framework of serious games for higher education: Conceptual framework of the game INNOV8 to train students in business process modelling	2010	IEEE
ES30	A Software Process Simulator Machine for Software Engineering Simulation Games	2010	IEEE
ES31	A Software Engineering Education Game in a 3-D Online Virtual Environment	2009	IEEE
ES32	Games-Based Requirements Engineering Training: An Initial Experience Report	2008	IEEE
ES33	Work in progress - a game-based learning system for software engineering education	2008	IEEE
ES34	Applying serious games for supporting idea generation in collaborative innovation processes	2008	IEEE
ES35	Collaboration in Software Engineering: A Roadmap	2007	IEEE
ES36	Improving Productivity of Local Software Development Teams in a Global Software Development Environment	2006	IEEE
ES37	Towards game-based simulation as a method of teaching software engineering	2002	IEEE
ES38	Evaluating software engineering simulation games: The UGALCO framework	2014	IEEE
ES39	Requirements compliance as a measure of project success	2013	IEEE
ES40	A Software Gamification Model for Cross-Cultural Software Development Teams	2017	ACM
ES41	Impact of Gamification on Code Review Process: An Experimental Study	2017	ACM
ES42	Can a Team Coordination Game Help Student Software Project Teams?	2016	ACM
ES43	Designing Game-like Activities to Engage Adult Learners in Higher Education	2016	ACM

ES44	Gamification for Enforcing Coding Conventions	2015	ACM
ES45	Gamifying Software Security Education and Training via Secure Coding Duels in Code Hunt	2015	ACM
ES46	Teaching Students Scrum Using LEGO Blocks	2014	ACM
ES47	Experiences Gamifying Developer Adoption of Practices and Tools	2014	ACM
ES48	Games for Teaching Software Development	2013	ACM
ES49	Learning Software Engineering Processes Through Playing Games: Suggestions for Next Generation of Simulations and Digital Learning Games	2012	ACM
ES50	Semiotic Inspection Method in the Context of Educational Simulation Games	2010	ACM
ES51	The Groupthink Specification Exercise	2005	ACM
ES52	Improving UML Design Tools by Formal Games	2004	ACM
ES53	ScoringTalk: A Tablet System Scoring and Visualizing Conversation for Balancing of Participation	2015	ACM
ES54	An open-source platform for using gamification and social learning methodologies in engineering education: Design and experience	2016	EL
ES55	Gamification proposal for defect tracking in software development process	2016	EL
ES56	gTest Learning: A game for learning basic software testing	2016	EL
ES57	Towards a serious game to teach ISO/IEC 12207 software lifecycle process: An interactive learning approach	2015	EL
ES58	Validation of usability driven web based software process model using simulation	2015	EL
ES59	Board game as a tool to teach software engineering concept - Technical debt	2014	EL
ES60	Supporting the ideation processes by a collaborative online based toolset	2011	EL
ES61	Employing software maintenance techniques via a tower-defense serious computer game	2011	EL
ES62	Educating information systems students on business process management (BPM) through digital gaming metaphors of virtual reality	2009	EL
ES63	Embracing business context in pedagogical simulation games - A case with process disciplined project management	2008	EL
ES64	Learning lean through lean game - A case from the infrastructure industry	2007	EL
ES65	The effects of students motivation, cognitive load and learning anxiety in gamification software engineering education: a structural equation modeling study	2016	EL

ES66	A systematic literature review of empirical evidence on computer games and serious games	2012	EL
ES67	Games' usability and learning - the educational videogame BeTheManager!	2011	EL
ES68	A collaborative learning methodology for enhanced comprehension using TEAMThink&reg	2002	EL
ES69	Multi-Role Project (MRP): A New Project-Based Learning Method for STEM	2016	WOK
ES70	A Simulation and Gamification Approach for IT Service Management Improvement	2016	WOK
ES71	Software Architecture Design Reasoning: A Card Game to Help Novice Designers	2016	WOK
ES72	Software Engineering Education and Games: A Systematic Literature Review	2016	WOK
ES73	DEVELOPING VIRTUAL LEARNING ENVIRONMENTS FOR SOFTWARE ENGINEERING EDUCATION: A LUDIC PROPOSAL	2015	WOK
ES74	SCRUMIA-An educational game for teaching SCRUM in computing courses	2013	WOK
ES75	Game-based ICT Project Formulation based on Agile Approach for Skills Development	2013	WOK
ES76	The Effect of Varied Game-Based Learning Systems in Engineering Education: An Experimental Study	2011	WOK
ES77	Model Development of a Virtual Learning Environment to Enhance Lean Education	2011	WOK
ES78	Integration of game elements with role play in collaborative learning - A case study of Quasi-GBL in chinese higher education	2008	WOK
ES79	KMsim: A meta-modelling approach and environment for creating process-oriented knowledge management simulations	2002	WOK
ES80	Schoolcube: Gamification for learning management system through microsoft sharepoint	2015	SCOPUS
ES81	Moving toward "reality" in team selection for software engineering	2008	SCOPUS

## APÊNDICE C – Estudos Excluídos

ID	Título (Artigos Excluídos - SPI)	Ano de Publicação	Fonte
EE01	A Digital Game Maturity Model (DGMM)	2016	Science Direct
EE02	Initiating private-collective innovation: The fragility of knowledge sharing	2010	Science Direct
EE03	Antecedents and consequences of perceived knowledge update in the context of an ERP\ simulation game: A multi-level perspective	2016	Science Direct
EE04	The ERP\ Challenge: An Integrated E-learning Platform for the Teaching of Practical ERP\ Skills in Universities	2016	Science Direct
EE05	Laboratory Experiences in Software Engineering from a Constructivist Perspective	2013	Science Direct
EE06	EGameFlow: A scale to measure learners' enjoyment of e-learning games	2009	Science Direct
EE07	Competition and open source with perfect software compatibility	2009	Science Direct
EE08	A non-cooperative differential game-based security model in fog computing	2017	IEEE
EE09	An Integrated Approach to the Requirements Engineering and Process Modelling Teaching	2016	IEEE
EE10	Strengthening study skills by using ERPsim as a new tool within the Pupils' academy of serious gaming	2016	IEEE
EE11	A Serious Game to Promote Object Oriented Programming and Software Engineering Basic Concepts Learning	2016	IEEE
EE12	Method of pedagogic instruments design for software engineering	2016	IEEE
EE13	Assessment model in a selection process based in gamification	2016	IEEE
EE14	An analysis of software requirements prioritization techniques: A detailed survey	2016	IEEE
EE15	Leveraging role play to explore software and game development process	2016	IEEE
EE16	Gamification in software engineering teamworks: A systematic literature review	2016	IEEE
EE17	Teaching and Learning Object-Oriented Analysis and Design with 3D Game	2016	IEEE
EE18	Model-Driven Serious Game Development Integration of the Gamification Modeling Language GaML with Unity	2015	IEEE
EE19	Comparative study of the process model of	2015	IEEE

	Serious Game Design through the generic model DICE		
EE20	Requirements gathering and testing by game playing	2015	IEEE
EE21	The Rise of Serious Games and Gamified Application in Software Development	2015	IEEE
EE22	iHEARu-PLAY: Introducing a game for crowdsourced data collection for affective computing	2015	IEEE
EE23	A game-based learning software model	2015	IEEE
EE24	WAP: Cognitive aspects in unit testing: The hunting game and the hunter's perspective	2015	IEEE
EE25	Integrating playful activities in software engineering teaching	2015	IEEE
EE26	ARMI: A risk management incorporation	2014	IEEE
EE27	Design of a pedagogic instrument for teaching software process improvement: Teaching instrument for university and business environments	2014	IEEE
EE28	A general framework for software project management simulation games	2014	IEEE
EE29	A feature modeling approach for domain-specific requirement elicitation	2014	IEEE
EE30	A Game Theory model used to increase the readiness for software process improvement	2014	IEEE
EE31	A Systematic Review on Software Engineering in Pervasive Games Development	2014	IEEE
EE32	Dilemma structures between contracting parties in software development projects	2014	IEEE
EE33	Applying competitive bidding games in software process education	2013	IEEE
EE34	Using experimental games to understand communication and trust in Agile software teams	2013	IEEE
EE35	Games in the workplace: Revolutionary or run-of-the-mill?	2013	IEEE
EE36	Methodology of Teaching Software Engineering: Game-Based Learning Cycle	2013	IEEE
EE37	Teaching agile software development through lab courses	2012	IEEE
EE38	Representations of user feedback in an agile, collocated software team	2012	IEEE
EE39	Towards model-driven game engineering for serious educational games: Tailored use cases for game requirements	2012	IEEE
EE40	Using game level design as an applied method for Software Engineering education	2012	IEEE

EE41	A software development process model for gesture-based interface	2012	IEEE
EE42	An improved way for evaluating competences: A different approach to project management learning	2011	IEEE
EE43	Agile and Kanban in Coordination	2011	IEEE
EE44	Work in progress #x2014; Using social media to teach engineering process	2011	IEEE
EE45	A design aid and real-time measurement framework for Virtual collaborative simulation Environment	2010	IEEE
EE46	An Evolutionary Game of Partners in Peer Review	2010	IEEE
EE47	Design and application of practical teaching framework of ERP course	2009	IEEE
EE48	Teaching Object-Oriented Software Engineering through Problem-Based Learning in the Context of Game Design	2008	IEEE
EE49	Providing Feedback from Application to Family Engineering - The Product Line Planning Game at the Testo AG	2008	IEEE
EE50	Notice of Violation of IEEE Publication Principles Adapting Game Technology to Support Software Engineering Process Teaching: From SimSE to MO-SEProcess	2007	IEEE
EE51	Factors Affecting End Users' Intrinsic Motivation to Use Software	2007	IEEE
EE52	Enhancing software engineering education using teaching aids in 3-D online virtual worlds	2007	IEEE
EE53	An Empirical Model of the Game Software Development Processes	2006	IEEE
EE54	Analysis on Enterprise's Software Project Management Based on Game Theory	2006	IEEE
EE55	Virtual communities for software maintenance	2004	IEEE
EE56	Preparing software engineers for the 'real world'	2002	IEEE
EE57	Learning Lean Philosophy Through 3D Game-based Simulation	2016	ACM
EE58	Teaching University Students Kanban with a Collaborative Board Game	2016	ACM
EE59	Code Review Participation: Game Theoretical Modeling of Reviewers in Gerrit Datasets	2016	ACM
EE60	AxGames: Towards Crowdsourcing Quality Target Determination in Approximate Computing	2016	ACM
EE61	A Gamification Approach for Distributed	2016	ACM

	Agile Delivery		
EE62	The Rise of Serious Games and Gamified Application in Software Development	2015	ACM
EE63	Design Approaches for the Gamification of Production Environments: A Study Focusing on Acceptance	2015	ACM
EE64	Towards a Gamification of Industrial Production: A Comparative Study in Sheltered Work Environments	2015	ACM
EE65	Transparently Teaching in the Context of Game-based Learning: The Case of simulES-W	2015	ACM
EE66	HeapCraft: Interactive Data Exploration and Visualization Tools for Understanding and Influencing Player Behavior in Minecraft	2015	ACM
EE67	Tools for Gamification Analytics: A Survey	2014	ACM
EE68	Process Simulation for Software Engineering Education	2014	ACM
EE69	(Un)Structured Sources of Inspiration: Comparing the Effects of Game-like Cards and Design Cards on Creativity in Co-design Process	2014	ACM
EE70	Learning Analytics and Serious Games: Trends and Considerations	2014	ACM
EE71	Crowdsourcing and Gamification of Enterprise Meeting Software Quality	2014	ACM
EE72	Incrementally Synthesizing Controllers from Scenario-based Product Line Specifications	2013	ACM
EE73	Towards Recognizing and Rewarding Efficient Developer Work Patterns	2013	ACM
EE74	It Was a Bit of a Race: Gamification of Version Control	2012	ACM
EE75	Using Games to Improve Productivity in Software Engineering	2010	ACM
EE76	Maximizing the Value of the Software Development Process by Game Theoretic Analysis	2010	ACM
EE77	Introducing Software Engineering Processes via Games and Simulations: A Tri-P-LETS Initiative	#####	ACM
EE78	A Constructivist Approach to Teaching Software Processes	2007	ACM
EE79	Developing a Computer Strategy Game in an Undergraduate Course in Software Development Using Extreme Programming	#####	ACM
EE80	Teaching Software Engineering Through Game Design	2005	ACM
EE81	SimSE: An Educational Simulation Game for Teaching the Software Engineering	#####	ACM

	Process		
EE82	Problems and Programmers: An Educational Software Engineering Card Game	2003	ACM
EE83	Doom As an Interface for Process Management	2001	ACM
EE84	Assessing the effectiveness of digital game-based learning: Best practices	2016	EL
EE85	Gamification Solutions to Enhance Software User EngagementA Systematic Review	2016	EL
EE86	A Gamification approach for Distributed Agile Delivery	2016	EL
EE87	A Digital Game Maturity Model (DGMM)	2016	EL
EE88	Developing a serious game for business information visualization	2016	EL
EE89	Empirical analysis of the E-business enterprise knowledge transfer and innovation performance based on multi-agent simulation model	2016	EL
EE90	A game of refactoring. Studying the impact of gamification in software refactoring	2016	EL
EE91	Gamifying the learning of design patterns in software engineering education	2016	EL
EE92	Agile workbench: Tying people, process, and tools in distributed agile delivery	2016	EL
EE93	An analysis of software requirements prioritization techniques: A detailed survey	2016	EL
EE94	An agile software engineering process improvement game	2016	EL
EE95	Adopting collaborative games into Open Kanban	2016	EL
EE96	Teaching ISO/IEC 12207 software lifecycle processes: A serious game approach	2016	EL
EE97	Supporting the requirements elicitation process for cyber-physical product-service systems through a gamified approach	2016	EL
EE98	Gamification in software engineering - A systematic mapping	2015	EL
EE99	Experimental evaluation of a serious game for teaching software process modeling	2015	EL
EE100	The use of games on the teaching of programming: A systematic review	2015	EL
EE101	A systematic literature review on serious games evaluation: An application to software project management	2015	EL
EE102	Minecraft-based preparatory training for software development project	2015	EL

EE103	A compromise-negotiation framework based on game theory for eliminating requirements inconsistency	2015	EL
EE104	Discovering the essence of Software Engineering an integrated game-based approach based on the SEMAT Essence specification	2015	EL
EE105	Gamification and persuasion of HP IT service management to improve performance and engagement	2015	EL
EE106	Transparently Teaching in the Context of Game-based Learning: The Case of SimulES-W	2015	EL
EE107	A game of attribute decomposition for software architecture design	2015	EL
EE108	ARMI 2.0: An online risk management simulation	2015	EL
EE109	Process simulation for software engineering education	2015	EL
EE110	Applying gamification in the context of knowledge management	2015	EL
EE111	The Rise of Serious Games and Gamified Application in Software Development	2015	EL
EE112	Game design techniques for software engineering management education	2015	EL
EE113	Gamifying software security education and training via secure coding duels in code hunt	2015	EL
EE114	InspectorX: A game for software inspection training and learning	2014	EL
EE115	Enhancing software engineering student team engagement in a high-intensity extreme programming course using gamification	2014	EL
EE116	Gamification in the development of accessible software	2014	EL
EE117	ARMI: A risk management incorporation	2014	EL
EE118	Software Development Processes for Games: A Systematic Literature Review	2014	EL
EE119	Evaluating multiple aspects of educational computer games: Literature review and case study	2014	EL
EE120	A Systematic Review on Software Engineering in Pervasive Games Development	2014	EL
EE121	A game theory model used to increase the readiness for software process improvement	2014	EL
EE122	A systematic mapping study on practical	2014	EL

	approaches to teaching software engineering		
EE123	Gamification of software engineering curriculum	2014	EL
EE124	Literature Review in Game-Based Learning	2013	EL
EE125	Understanding gamification mechanisms for software development	2013	EL
EE126	Using software engineering concepts in game development - Sharing experiences of two institutions	2013	EL
EE127	Applying competitive bidding games in software process education	2013	EL
EE128	An educational simulation model derived from academic and industrial experiences	2013	EL
EE129	Learning software engineering processes through playing games	2012	EL
EE130	A role-playing game for a software engineering lab: Developing a product line	2012	EL
EE131	UbiRE: A game for teaching requirements in the context of ubiquitous systems	2012	EL
EE132	Applying mLearning in software engineering education: A survey of mobile usage	2012	EL
EE133	HALO (Highly Addictive, socialLly Optimized) software engineering	2011	EL
EE134	Teaching software engineering with SimulES-W	2011	EL
EE135	Mission to Mars: An agile release planning game	2011	EL
EE136	Serious games usability testing: How to ensure proper usability, playability, and effectiveness	2011	EL
EE137	Simulating the software engineering interview process using a decision-based serious computer game	2011	EL
EE138	An agile boot camp: Using a LEGO&reg;-based active game to ground agile development principles	2011	EL
EE139	Introducing object oriented design patterns through a puzzle-based serious computer game	2011	EL
EE140	Turning real-world software development into a game	2011	EL
EE141	Semiotic inspection method in the context of educational simulation games	2010	EL
EE142	CUTA4UML - Bridging the gap between informal and formal requirements for dynamic system aspects	2010	EL

EE143	A conceptual framework of serious games for higher education: Conceptual framework of the game Innov8 to train students in business process modelling	2010	EL
EE144	Work in progress - An investigation of varied game-based learning systems in engineering education	2010	EL
EE145	A software engineering education game in a 3-D online virtual environment	2009	EL
EE146	HACME game: A tool for teaching software security	2009	EL
EE147	Using educational game design to teach software engineering	2009	EL
EE148	A serious game for enterprise knowledge transmission: An agent based perspective	2008	EL
EE149	A constructivist approach to teaching software processes	2007	EL
EE150	Adapting game technology to support software engineering process teaching: From SimSE to MO-SEProcess	2007	EL
EE151	Interactive simulation in crisis management	2007	EL
EE152	Quality of protection: Measuring the unmeasurable?	2006	EL
EE153	Improving productivity of local software development teams in a global software development environment	2006	EL
EE154	MACSIM: Serious gaming in crisis management via script-based simulation	2006	EL
EE155	Design and evaluation of an educational software process simulation environment and associated model	2005	EL
EE156	It's all in the game: Teaching software process concepts	2005	EL
EE157	Design and evaluation of an educational software process simulation environment and associated model	2005	EL
EE158	Engendering an Empathy for Software Engineering	2005	EL
EE159	SimSE: An educational simulation game for teaching the software engineering process	2004	EL
EE160	Problems and programmers: An educational software engineering card game	2003	EL
EE161	An experimental card game for teaching software engineering	2003	EL
EE162	An agile request for proposal (RFP) process	2003	EL
EE163	Being the new guy in an experienced team - Enhancing training on the job	2002	EL

EE164	Towards game-based simulation as a method of teaching software engineering	2002	EL
EE165	Creating an effective training environment for enhancing telework	2000	EL
EE166	Gamification Behind the Scenes Designing a Software Engineering Course	2016	WOK
EE167	A SCRUMBAN INTEGRATED GAMIFICATION APPROACH TO GUIDE SOFTWARE PROCESS IMPROVEMENT: A TURKISH CASE STUDY	2016	WOK
EE168	Interdisciplinary Projects in the Academic Studio	2016	WOK
EE169	Gamification Solutions to Enhance Software User Engagement A Systematic Review	2016	WOK
EE170	Coverage of ISO/IEC 12207 Software Lifecycle Process by a Simulation-Based Serious Game	2016	WOK
EE171	A Gamification Approach to Improve the Software Development Process by Exploring the Personality of Software Practitioners	2016	WOK
EE172	A Learning Tool for the ISO/IEC 29110 Standard: Understanding the Project Management of Basic Profile	2016	WOK
EE173	Software Engineering Management Education through Game Design Patterns	2016	WOK
EE174	Gamifying the Learning of Design Patterns in Software Engineering Education	2016	WOK
EE175	Strengthening Study Skills by using ERPsim as a new Tool within the Pupils' Academy of Serious Gaming	2016	WOK
EE176	An Approach to Gamify an Adaptive Questionnaire Environment	2016	WOK
EE177	A Digital Game Maturity Model (DGMM)	2016	WOK
EE178	Agile Workbench: Tying People, Process, and Tools in Distributed Agile Delivery	2016	WOK
EE179	Practice Makes perfect - Gamification of a Competitive Learning Experience	2016	WOK
EE180	Bridging the Gap Between SPI and SMEs in Educational Settings: A Learning Tool Supporting ISO/IEC 29110	2016	WOK
EE181	Gamification Proposal for Defect Tracking in Software Development Process	2016	WOK
EE182	Process Improving by Playing: Implementing Best Practices through Business Games	2016	WOK
EE183	Gamifying the Onboarding Process for Novice Software Practitioners	2016	WOK

EE184	An Agile Software Engineering Process Improvement Game	2016	WOK
EE185	Leveraging Role Play to Explore Software and Game Development Process	2016	WOK
EE186	Adopting collaborative games into Open Kanban	2016	WOK
EE187	GSDgame: A serious game for the acquisition of the competencies needed in GSD	2016	WOK
EE188	Improving Social Aspects of the Software Development Process: Games, Gamification and Related Approaches	2016	WOK
EE189	Software Engineering Education and Games: A Systematic Literature Review	2016	WOK
EE190	Gamification in software engineering - A systematic mapping	2015	WOK
EE191	GAMED: digital educational game development methodology	2015	WOK
EE192	A COMPROMISE-NEGOTIATION FRAMEWORK BASED ON GAME THEORY FOR ELIMINATING REQUIREMENTS INCONSISTENCY	2015	WOK
EE193	A systematic literature review on serious games evaluation: An application to software project management	2015	WOK
EE194	Validation of Usability Driven Web based Software Process Model using Simulation	2015	WOK
EE195	A Game of Attribute Decomposition for Software Architecture Design	2015	WOK
EE196	Towards a Serious Game to Teach ISO/IEC 12207 Software Lifecycle Process: An Interactive Learning Approach	2015	WOK
EE197	Experimental Evaluation of a Serious Game for Teaching Software Process Modeling	2015	WOK
EE198	Taking Seriously Software Projects Inception Through Games	2015	WOK
EE199	AgiRPG: An Agile-Based Role-Playing Game Design System	2015	WOK
EE200	The Rise of Serious Games and Gamified Application in Software Development	2015	WOK
EE201	ARMI 2.0: An Online Risk Management Simulation	2015	WOK
EE202	Discovering the Essence of Software Engineering an Integrated Game-Based Approach based on the SEMAT Essence Specification	2015	WOK
EE203	Game Design Techniques for Software Engineering Management Education	2015	WOK
EE204	Transparently Teaching in the Context of	2015	WOK

	Game-based Learning: the Case of SimulES-W		
EE205	WAP: Cognitive Aspects in Unit Testing The Hunting Game and the Hunter's Perspective	2015	WOK
EE206	Gamification Proposal for a Software Engineering Risk Management Course	2015	WOK
EE207	InspectorX: A Game for Software Inspection Training and Learning	2014	WOK
EE208	Enhancing Software Engineering Student Team Engagement in a High-Intensity Extreme Programming Course using Gamification	2014	WOK
EE209	ARMI: A Risk Management Incorporation	2014	WOK
EE210	Learning Analytics and Educational Games: Lessons Learned from Practical Experience	2014	WOK
EE211	A General Framework for Software Project Management Simulation Games	2014	WOK
EE212	Reinforcing Software Engineering Learning Through Provenance	2014	WOK
EE213	Board Game as a tool to teach Software Engineering Concept - Technical Debt	2014	WOK
EE214	ON THE DEVELOPMENT OF AN OPEN-SOURCE SYSTEM FOR INTRODUCING GAMIFICATION IN HIGHER EDUCATION	2014	WOK
EE215	Minecraft-Based Preparatory Training for Software Development Project	2014	WOK
EE216	A Game Theory Model used to Increase the Readiness for Software Process Improvement	2014	WOK
EE217	Gamification of Software Engineering Curriculum	2014	WOK
EE218	Serious Games for Motivating into Programming	2014	WOK
EE219	A Systematic Mapping Study on Practical Approaches to Teaching Software Engineering	2014	WOK
EE220	Sparking Creativity: Improving Electronic Brainstorming with Individual Cognitive Priming	2013	WOK
EE221	Learning Risk Management in Software Projects with a Serious Game Based on Intelligent Agents and Fuzzy Systems	2013	WOK
EE222	Applying Competitive Bidding Games in Software Process Education	2013	WOK
EE223	An Educational Simulation Model Derived from Academic and Industrial Experiences	2013	WOK
EE224	Using Experimental Games to Understand Communication and Trust in Agile Software	2013	WOK

	Teams		
EE225	A role-playing game for a software engineering lab: developing a product line	2012	WOK
EE226	e-SPM: An Online Software Project Management Game	2012	WOK
EE227	A Serious Game for Supporting Training in Risk Management through Project-Based Learning	2012	WOK
EE228	DroidSense: A Mobile Tool to Analyze Software Development Processes by Measuring Team Proximity	2012	WOK
EE229	Development of Computer Games for Training Programming Skills	2012	WOK
EE230	Challenges and Issues in the Development of a Software Engineering Simulation Game	2012	WOK
EE231	Teaching Software Engineering with SimulES-W	2011	WOK
EE232	Mission to Mars: An Agile Release Planning Game	2011	WOK
EE233	Hard Choice: A game for balancing strategy for agility	2011	WOK
EE234	An Agile Boot Camp: Using a LEGO (R)-Based Active Game to Ground Agile Development Principles	2011	WOK
EE235	Introducing Object Oriented Design Patterns through a Puzzle-Based Serious Computer Game	2011	WOK
EE236	UML Modeling for Software System of Edu-Game	2010	WOK
EE237	Using Card-Based Games in Semi-Structured Interviews	2010	WOK
EE238	Work in Progress - An Investigation of Varied Game-Based Learning Systems in Engineering Education	2010	WOK
EE239	Organizational Enablers for Agile Adoption: Learning from GameDevCo	2009	WOK
EE240	Empirical evaluation of an educational game on software measurement	2009	WOK
EE241	A Software Engineering Education Game in a 3-D Online Virtual Environment	2009	WOK
EE242	Experiential learning through simulation games: An empirical study	2009	WOK
EE243	hACMEgame: A Tool for Teaching Software Security	2009	WOK
EE244	Study on the Construction of Information System for Project Management in Comprehensive Games	2009	WOK
EE245	Using Educational Game Design to Teach Software Engineering	2009	WOK

EE246	3I-Approach for IT Educational Games Development	2009	WOK
EE247	Teaching Object-Oriented Software Engineering through Problem-Based Learning in the context of game design	2008	WOK
EE248	Developing and evaluating a game-based software engineering educational system	2008	WOK
EE249	A SERIOUS GAME FOR ENTERPRISE KNOWLEDGE TRANSMISSION: AN AGENT BASED PERSPECTIVE	2008	WOK
EE250	Come Play	2008	WOK
EE251	Work in Progress - A Game-based Learning System for Software Engineering Education	2008	WOK
EE252	A constructivist approach to teaching software processes	2007	WOK
EE253	Adapting game technology to support software engineering process teaching: From SimSE to MO-SEProcess	2007	WOK
EE254	Analysis on enterprise's software project management based on game theory	2006	WOK
EE255	Improving productivity of local software development teams in a global software development environment	2006	WOK
EE256	An experimental card game for teaching software engineering processes	2005	WOK
EE257	Design and evaluation of an educational software process simulation environment and associated model	2005	WOK
EE258	Applying story driven modeling to the Paderborn shuttle system case study	2005	WOK
EE259	Training for today's supply chains: An introduction to the distributor game	2005	WOK
EE260	An experimental card game for teaching software engineering	2003	WOK
EE261	Problems and programmers: An educational software engineering card game	2003	WOK
EE262	Impacts of SCM order strategies evaluated by simulation-based 'Beer Game' approach: the model, concept, and initial experiences	2003	WOK
EE263	Developing a tool supporting XP process	2003	WOK
EE264	Games for UML software design	2003	WOK
EE265	Intelligent software agent (ISA) technology for Business Simulation Games	2003	WOK
EE266	Process-oriented interactive simulation of software acquisition projects	2002	WOK
EE267	Evaluating FINESSE: a case-study in group-based CAL	2001	WOK

EE268	Gamification at scraim	2017	SCOPUS
EE269	A Digital Game Maturity Model (DGMM)	2016	SCOPUS
EE270	Teaching ISO/IEC 12207 software lifecycle processes: A serious game approach	2016	SCOPUS
EE271	Gamifying the learning of design patterns in software engineering education	2016	SCOPUS
EE272	Gamification behind the scenes: Designing a software engineering course	2016	SCOPUS
EE273	Coverage of ISO/IEC 12207 software lifecycle process by a simulation-based serious game	2016	SCOPUS
EE274	Gamification Solutions to Enhance Software User Engagement—A Systematic Review	2016	SCOPUS
EE275	Leveraging role play to explore software and game development process	2016	SCOPUS
EE276	A game of refactoring. Studying the impact of gamification in software refactoring	2016	SCOPUS
EE277	Gamification proposal for defect tracking in software development process	2016	SCOPUS
EE278	Developing a serious game for business information visualization	2016	SCOPUS
EE279	Towards a gamification framework for software process improvement initiatives: Construction and validation	2016	SCOPUS
EE280	Towards the effective software development of an elearning platform featuring learning analytics and gamification	2016	SCOPUS
EE281	Agile retrospective games for different team development phases	2016	SCOPUS
EE282	RePizer: a framework for prioritization of software requirements	2016	SCOPUS
EE283	Practice makes perfect - Gamification of a competitive learning experience	2016	SCOPUS
EE284	Software engineering education and games: A systematic literature review	2016	SCOPUS
EE285	Gamifying the onboarding process for novice software practitioners	2016	SCOPUS
EE286	Using educational games and simulation software in a computer science course: learning achievements and student flow experiences	2016	SCOPUS
EE287	An agile software engineering process improvement game	2016	SCOPUS
EE288	Process improving by playing: Implementing best practices through business games	2016	SCOPUS
EE289	Business intelligence tools: Should they be 'gamified'? Project 'GamBIT': Evaluating	2016	SCOPUS

	user engagement		
EE290	An approach to gamify an adaptive questionnaire environment	2016	SCOPUS
EE291	Measuring business process learning with enterprise resource planning systems to improve the value of education	2016	SCOPUS
EE292	Designing game-like activities to engage adult learners in higher education	2016	SCOPUS
EE293	Adopting collaborative games into Open Kanban	2016	SCOPUS
EE294	Game theory-based behavioral decision on it projects risk	2016	SCOPUS
EE295	A serious game to promote object oriented programming and software engineering basic concepts learning	2016	SCOPUS
EE296	Gamification and functional prototyping to support motivation towards software process improvement	2016	SCOPUS
EE297	A learning tool for the ISO/IEC 29110 standard: Understanding the project management of basic profile	2016	SCOPUS
EE298	Bridging the gap between SPI and SMES in educational settings: A learning tool supporting ISO/IEC 29110	2016	SCOPUS
EE299	SOA applied: Engineering software as processing unit of a serious game	2016	SCOPUS
EE300	Analysis of linear sequential and extreme programming development methodology for a gaming application	2016	SCOPUS
EE301	A Gamification approach for Distributed Agile Delivery	2016	SCOPUS
EE302	Agile workbench: Tying people, process, and tools in distributed agile delivery	2016	SCOPUS
EE303	An open-source platform for using gamification and social learning methodologies in engineering education: Design and experience	2016	SCOPUS
EE304	Ace that game: Educating students to gamified design thinking	2016	SCOPUS
EE305	GSDgame: A serious game for the acquisition of the competencies needed in GSD	2016	SCOPUS
EE306	Supporting the requirements elicitation process for cyber-physical product-service systems through a gamified approach	2016	SCOPUS
EE307	Improving social aspects of the software development process: Games, gamification and related approaches	2016	SCOPUS

EE308	A scrumban integrated gamification approach to guide software process improvement: A Turkish case study	2016	SCOPUS
EE309	A gamification approach to improve the software development process by exploring the personality of software practitioners	2016	SCOPUS
EE310	An analysis of software requirements prioritization techniques: A detailed survey	2016	SCOPUS
EE311	The Rise of Serious Games and Gamified Application in Software Development	2015	SCOPUS
EE312	Towards a serious game to teach ISO/IEC 12207 software lifecycle process: An interactive learning approach	2015	SCOPUS
EE313	Short serious games creation under the paradigm of software process and competencies as software requirements. Case study: Elementary math competencies	2015	SCOPUS
EE314	A systematic literature review on serious games evaluation: An application to software project management	2015	SCOPUS
EE315	AgiRPG: An agile-based role-playing game design system	2015	SCOPUS
EE316	Experimental evaluation of a serious game for teaching software process modeling	2015	SCOPUS
EE317	The use of games on the teaching of programming: A systematic review	2015	SCOPUS
EE318	The teacher's role in gamification in software engineering at universities (Field Report) - Or how geeks can be inspired to sing	2015	SCOPUS
EE319	Gamiware: A gamification platform for software process improvement	2015	SCOPUS
EE320	Process simulation for software engineering education	2015	SCOPUS
EE321	Process improvement with retrospective gaming in agile software development	2015	SCOPUS
EE322	Applying gamification in the context of knowledge management	2015	SCOPUS
EE323	Designing games for improving the software development process	2015	SCOPUS
EE324	Game design techniques for software engineering management education	2015	SCOPUS
EE325	The study on knowledge transferring incentive for information system requirement development	2015	SCOPUS
EE326	A game of attribute decomposition for software architecture design	2015	SCOPUS
EE327	Scrum in software engineering courses: An outline of the literature	2015	SCOPUS

EE328	Assessment proposal of teaching and learning strategies in software process improvement	2015	SCOPUS
EE329	Assessment proposal of teaching and learning strategies in software process improvement	2015	SCOPUS
EE330	Working and Playing with Scrum	2015	SCOPUS
EE331	Transparently Teaching in the Context of Game-based Learning: The Case of SimulES-W	2015	SCOPUS
EE332	Taking seriously software projects inception through games	2015	SCOPUS
EE333	Gamification in software engineering - A systematic mapping	2015	SCOPUS
EE334	Discovering the essence of Software Engineering an integrated game-based approach based on the SEMAT Essence specification	2015	SCOPUS
EE335	Minecraft-based preparatory training for software development project	2015	SCOPUS
EE336	ARMI 2.0: An online risk management simulation	2015	SCOPUS
EE337	Gamifying software development environments using cognitive principles	2015	SCOPUS
EE338	Gamification proposal for a software engineering risk management course	2015	SCOPUS
EE339	Gamifying software security education and training via secure coding duels in code hunt	2015	SCOPUS
EE340	Gamification and persuasion of HP IT service management to improve performance and engagement	2015	SCOPUS
EE341	Enhancing software engineering student team engagement in a high-intensity extreme programming course using gamification	2014	SCOPUS
EE342	MRI design review system: A mixed reality interactive design review system for architecture, serious games and engineering using game engines, standard software, a tablet computer and natural interfaces	2014	SCOPUS
EE343	A game theory model used to increase the readiness for software process improvement	2014	SCOPUS
EE344	A prototype for project management game development using high level architecture	2014	SCOPUS
EE345	Design and implementation of serious games for training and education	2014	SCOPUS

EE346	Board game as a tool to teach software engineering concept - Technical debt	2014	SCOPUS
EE347	Gamification as a disruptive factor in software process improvement initiatives	2014	SCOPUS
EE348	Serious games for motivating into programming	2014	SCOPUS
EE349	Reinforcing software engineering learning through provenance	2014	SCOPUS
EE350	A systematic mapping study on practical approaches to teaching software engineering	2014	SCOPUS
EE351	A general framework for software project management simulation games	2014	SCOPUS
EE352	Software Development Processes for Games: A Systematic Literature Review	2014	SCOPUS
EE353	InspectorX: A game for software inspection training and learning	2014	SCOPUS
EE354	ARMI: A risk management incorporation	2014	SCOPUS
EE355	Learning analytics and educational games:Lessons learned from practical experience	2014	SCOPUS
EE356	The development of sokoban game based on android system	2014	SCOPUS
EE357	Gamification in the development of accessible software	2014	SCOPUS
EE358	Quality risk management for computerised systems-a review	2014	SCOPUS
EE359	Gamification of software engineering curriculum	2014	SCOPUS
EE360	A Systematic Review on Software Engineering in Pervasive Games Development	2014	SCOPUS
EE361	Knowledge management approach for serious game development based on user experience	2014	SCOPUS
EE362	Learning risk management in software projects with a serious game based on intelligent agents and fuzzy systems	2013	SCOPUS
EE363	Understanding gamification mechanisms for software development	2013	SCOPUS
EE364	Using experimental games to understand communication and trust in Agile software teams	2013	SCOPUS
EE365	A modeling language to describe reusable learning processes to achieve educational objectives in serious games	2013	SCOPUS
EE366	The role of communication in agile systems development: An analysis of the state of the art	2013	SCOPUS

EE367	Literature Review in Game-Based Learning	2013	SCOPUS
EE368	A multi-agent system for learner assessment in serious games: Application to learning processes in crisis management	2013	SCOPUS
EE369	An educational simulation model derived from academic and industrial experiences	2013	SCOPUS
EE370	Everything is just playing around? New approaches for digital game-based learning of software processes [Alles nur Spielerei? Neue Ansätze für digitales spielbasiertes Lernen von Softwareprozessen]	2013	SCOPUS
EE371	Applying competitive bidding games in software process education	2013	SCOPUS
EE372	Games for teaching software development	2013	SCOPUS
EE373	Methodology of teaching software engineering: Game-based learning cycle	2013	SCOPUS
EE374	Towards model-driven game engineering for serious educational games: Tailored use cases for game requirements	2012	SCOPUS
EE375	The gamification of SPICE	2012	SCOPUS
EE376	eRiskGame: A persistent browser-based game for supporting project-based learning in the risk management context	2012	SCOPUS
EE377	A serious game for supporting training in risk management through project-based learning	2012	SCOPUS
EE378	An approach to assess knowledge and skills in risk management through project-based learning	2012	SCOPUS
EE379	UbiRE: A game for teaching requirements in the context of ubiquitous systems	2012	SCOPUS
EE380	Applying mLearning in software engineering education: A survey of mobile usage	2012	SCOPUS
EE381	Challenges and issues in the development of a Software Engineering simulation game	2012	SCOPUS
EE382	Learning software engineering processes through playing games	2012	SCOPUS
EE383	Workshop: Learning agile through active learning activities	2012	SCOPUS
EE384	Development of computer games for training programming skills	2012	SCOPUS
EE385	It was a bit of a race: Gamification of version control	2012	SCOPUS
EE386	E-SPM: An online software project management game	2012	SCOPUS
EE387	A guideline for game development-based learning: A literature review	2012	SCOPUS
EE388	A role-playing game for a software engineering lab: Developing a product line	2012	SCOPUS

EE389	Hard choice: A game for balancing strategy for agility	2011	SCOPUS
EE390	Intelligent behavior simulation module for software process elements	2011	SCOPUS
EE391	A Software Process Simulator Machine for software engineering simulation games	2011	SCOPUS
EE392	The effect of varied game-based learning systems in engineering education: An experimental study	2011	SCOPUS
EE393	Model development of a virtual learning environment to enhance lean education	2011	SCOPUS
EE394	Mission to Mars: An agile release planning game	2011	SCOPUS
EE395	An agile boot camp: Using a LEGO®-based active game to ground agile development principles	2011	SCOPUS
EE396	Teaching software engineering with SimulES-W	2011	SCOPUS
EE397	Turning real-world software development into a game	2011	SCOPUS
EE398	Introducing object oriented design patterns through a puzzle-based serious computer game	2011	SCOPUS
EE399	Simulating the software engineering interview process using a decision-based serious computer game	2011	SCOPUS
EE400	Employing software maintenance techniques via a tower-defense serious computer game	2011	SCOPUS
EE401	HALO (Highly Addictive, socialLly Optimized) software engineering	2011	SCOPUS
EE402	A conceptual framework of serious games for higher education: Conceptual framework of the game Innov8 to train students in business process modelling	2010	SCOPUS
EE403	Work in progress - An investigation of varied game-based learning systems in engineering education	2010	SCOPUS
EE404	CUTA4UML - Bridging the gap between informal and formal requirements for dynamic system aspects	2010	SCOPUS
EE405	An evolutionary game of partners in peer review	2010	SCOPUS
EE406	Semiotic inspection method in the context of educational simulation games	2010	SCOPUS
EE407	UML modeling for software system of Edu-game	2010	SCOPUS
EE408	Using educational game design to teach software engineering	2009	SCOPUS

EE409	Empirical evaluation of an educational game on software measurement	2009	SCOPUS
EE410	Educating information systems students on business process management (BPM) through digital gaming metaphors of virtual reality	2009	SCOPUS
EE411	Design and application of practical teaching framework of ERP course	2009	SCOPUS
EE412	3I-approach for IT educational games development	2009	SCOPUS
EE413	Work in progress - A game-based learning system for software engineering education	2008	SCOPUS
EE414	The convergence of gaming practices with other media forms: What potential for learning? A review of the literature	2008	SCOPUS
EE415	Embracing business context in pedagogical simulation games - A case with process disciplined project management	2008	SCOPUS
EE416	A serious game for enterprise knowledge transmission: An agent based perspective	2008	SCOPUS
EE417	Developing and evaluating a game-based software engineering educational system	2008	SCOPUS
EE418	Process improvement of peer code review and behavior analysis of its participants	2008	SCOPUS
EE419	Learning lean through lean game - A case from the infrastructure industry	2007	SCOPUS
EE420	A constructivist approach to teaching software processes	2007	SCOPUS
EE421	Collaboration in software engineering: A roadmap	2007	SCOPUS
EE422	Enhancing software engineering education using teaching aids in 3-D online virtual worlds	2007	SCOPUS
EE423	Adapting game technology to support software engineering process teaching: From SimSE to MO-SEProcess	2007	SCOPUS
EE424	Improving productivity of local software development teams in a global software development environment	2006	SCOPUS
EE425	DiRT - Dust in Real-time: The specification process	2005	SCOPUS
EE426	An experimental card game for teaching software engineering processes	2005	SCOPUS
EE427	It's all in the game: Teaching software process concepts	2005	SCOPUS
EE428	Software process modeling for an educational software engineering simulation game	2005	SCOPUS

EE429	Design and evaluation of an educational software process simulation environment and associated model	2005	SCOPUS
EE430	Design and evaluation of an educational software process simulation environment and associated model	2005	SCOPUS
EE431	Engendering an Empathy for Software Engineering	2005	SCOPUS
EE432	SimSE: An educational simulation game for teaching the software engineering process	2004	SCOPUS
EE433	SimSE: An interactive simulation game for software engineering education	2004	SCOPUS
EE434	SimSE: An educational simulation game for teaching the software engineering process	2004	SCOPUS
EE435	An experimental card game for teaching software engineering	2003	SCOPUS
EE436	Problems and programmers: An educational software engineering card game	2003	SCOPUS
EE437	Process-oriented interactive simulation of software acquisition projects	2002	SCOPUS
EE438	Towards game-based simulation as a method of teaching software engineering	2002	SCOPUS
EE439	A feasible user story tool for agile software development?	2002	SCOPUS
EE440	Doom as an interface for process management	2001	SCOPUS

ID	Título (Artigos Excluídos - MED.)	Ano de Publicação	Fonte
EE01	EGameFlow: A scale to measure learners' enjoyment of e-learning games	2009	Science Direct
EE02	Roam, a seamless application framework	2004	Science Direct
EE03	The use of electronic curriculums in occupational education to evaluate and improve the cognitive capacity of candidate software engineers	2011	Science Direct
EE04	Requirements, Human Values, and the Development Technology Landscape	2017	IEEE
EE05	A Size Estimation Model for Board-Based Desktop Games	2017	IEEE
EE06	The pupils' academy of serious gaming: Strengthening study skills with ERPsim	2016	IEEE
EE07	Gamifying the learning of design patterns in software engineering education	2016	IEEE
EE08	Agile Workbench: Tying People, Process, and Tools in Distributed Agile Delivery	2016	IEEE

EE09	Beliefs, Practices, and Personalities of Software Engineers: A Survey in a Large Software Company	2016	IEEE
EE10	Experimental Evaluation of a Serious Game for Teaching Software Process Modeling	2015	IEEE
EE11	Crowdsourcing Code and Process via Code Hunt	2015	IEEE
EE12	CodeMetropolis: Eclipse over the city of source code	2015	IEEE
EE13	Enhancing software engineering student team engagement in a high-intensity extreme programming course using gamification	2014	IEEE
EE14	Minecraft-based preparatory training for software development project	2014	IEEE
EE15	Tools for Gamification Analytics: A Survey	2014	IEEE
EE16	Crowdsourcing and Gamification of Enterprise Meeting Software Quality	2014	IEEE
EE17	Mobile game size estimation: COSMIC FSM rules, UML mapping model and Unity3D game engine	2014	IEEE
EE18	Towards recognizing and rewarding efficient developer work patterns	2013	IEEE
EE19	Games in the workplace: Revolutionary or run-of-the-mill?	2013	IEEE
EE20	An educational simulation model derived from academic and industrial experiences	2013	IEEE
EE21	Agile and Kanban in Coordination	2011	IEEE
EE22	Turning Real-World Software Development into a Game	2011	IEEE
EE23	Work in progress #x2014; An investigation of varied game-based learning systems in engineering education	2010	IEEE
EE24	A conceptual framework of serious games for higher education: Conceptual framework of the game INNOV8 to train students in business process modelling	2010	IEEE
EE25	A Game-Theoretical Model for Task Assignment in Project Management	2006	IEEE
EE26	Impact of Gamification on Code Review Process: An Experimental Study	2017	ACM
EE27	Teaching University Students Kanban with a Collaborative Board Game	2016	ACM
EE28	Measuring Code Behavioral Similarity for Programming and Software Engineering Education	2016	ACM
EE29	Tools for Gamification Analytics: A Survey	2014	ACM
EE30	Process Simulation for Software Engineering Education	2014	ACM
EE31	Teaching Students Scrum Using LEGO	2014	ACM

	Blocks		
EE32	Crowdsourcing and Gamification of Enterprise Meeting Software Quality	2014	ACM
EE33	Towards Recognizing and Rewarding Efficient Developer Work Patterns	2013	ACM
EE34	Designing game strategies: An analysis from knowledge management in software development contexts	2017	EL
EE35	The pupils' academy of serious gaming: Strengthening study skills with ERPsim	2016	EL
EE36	Strengthening study skills by using ERPsim as a new tool within the Pupils' academy of serious gaming	2016	EL
EE37	Leveling up by gamifying freshman engineering clinic	2016	EL
EE38	Towards a serious game to teach ISO/IEC 12207 software lifecycle process: An interactive learning approach	2015	EL
EE39	Minecraft-based preparatory training for software development project	2015	EL
EE40	ScoringTalk: A tablet system scoring and visualizing conversation for balancing of participation	2015	EL
EE41	Gamification in the development of accessible software	2014	EL
EE42	Teaching students scrum using LEGO blocks	2014	EL
EE43	Crowdsourcing and gamification of enterprise meeting software quality	2014	EL
EE44	Requirements compliance as a measure of project success	2013	EL
EE45	Games in the workplace: Revolutionary or run-of-the-mill?	2013	EL
EE46	Towards recognizing and rewarding efficient developer work patterns	2013	EL
EE47	Empirical study upon software testing learning with support from educational game	2011	EL
EE48	A test case for GameDNA: Conceptualizing a serious game to measure personality traits	2011	EL
EE49	Work in progress - An investigation of varied game-based learning systems in engineering education	2010	EL
EE50	A model of cross-functional coopetition in software development project teams	2010	EL
EE51	To game or not to game?	2009	EL
EE52	Empirical evaluation of an educational game on software measurement	2009	EL
EE53	Learning lean through lean game - A case from the infrastructure industry	2007	EL

EE54	The effects of students' motivation, cognitive load and learning anxiety in gamification software engineering education: a structural equation modeling study	2016	WOK
EE55	The Pupils' Academy of Serious Gaming: Strengthening Study Skills with ERPsim	2016	WOK
EE56	Strengthening Study Skills by using ERPsim as a new Tool within the Pupils' Academy of Serious Gaming	2016	WOK
EE57	An Approach to Gamify an Adaptive Questionnaire Environment	2016	WOK
EE58	Agile Workbench: Tying People, Process, and Tools in Distributed Agile Delivery	2016	WOK
EE59	Towards a Serious Game to Teach ISO/IEC 12207 Software Lifecycle Process: An Interactive Learning Approach	2015	WOK
EE60	Minecraft-Based Preparatory Training for Software Development Project	2014	WOK
EE61	Tools for Gamification Analytics: A Survey	2014	WOK
EE62	Crowdsourcing and Gamification of Enterprise Meeting Software Quality	2014	WOK
EE63	Teaching Students Scrum using LEGO Blocks	2014	WOK
EE64	Requirements Compliance as a Measure of Project Success	2013	WOK
EE65	Towards Recognizing and Rewarding Efficient Developer Work Patterns	2013	WOK
EE66	BUSINESS MANAGEMENT SIMULATORS IN UNIVERSITY TEACHING: A STUDENT ASSESSMENT	2012	WOK
EE67	The Effect of Varied Game-Based Learning Systems in Engineering Education: An Experimental Study	2011	WOK
EE68	Games' Usability and Learning - the Educational Videogame BeTheManager!	2011	WOK
EE69	Work in Progress - An Investigation of Varied Game-Based Learning Systems in Engineering Education	2010	WOK
EE70	CREATING SUCCESSFUL GLOBAL TEAMS WITH STRATEGIC MANAGEMENT SIMULATION SOFTWARE	2010	WOK
EE71	Empirical evaluation of an educational game on software measurement	2009	WOK
EE72	Designing game strategies: An analysis from knowledge management in software development contexts	2017	SCOPUS
EE73	Does gamification increase engagement with online programs? A systematic review	2017	SCOPUS
EE74	An approach to gamify an adaptive	2016	SCOPUS

	questionnaire environment		
EE75	Measuring business process learning with enterprise resource planning systems to improve the value of education	2016	SCOPUS
EE76	Designing serious games with the game of games	2016	SCOPUS
EE77	The effects of students' motivation, cognitive load and learning anxiety in gamification software engineering education: a structural equation modeling study	2016	SCOPUS
EE78	Strengthening study skills by using ERPsim as a new tool within the Pupils' academy of serious gaming	2016	SCOPUS
EE79	Towards a serious game to teach ISO/IEC 12207 software lifecycle process: An interactive learning approach	2015	SCOPUS
EE80	Serious games analytics: Methodologies for performance measurement, assessment, and improvement	2015	SCOPUS
EE81	Minecraft-based preparatory training for software development project	2015	SCOPUS
EE82	Gaming technology for critical thinking: Engagement, usability, and measurement	2014	SCOPUS
EE83	Version control system gamification: A proposal to encourage the engagement of developers to collaborate in software projects	2014	SCOPUS
EE84	Tools for gamification analytics: A survey	2014	SCOPUS
EE85	Teaching students scrum using LEGO blocks	2014	SCOPUS
EE86	Evaluating software engineering simulation games: The UGALCO framework	2014	SCOPUS
EE87	Combining game-flow and learning objectives in educational games	2014	SCOPUS
EE88	Crowdsourcing and gamification of enterprise meeting software quality	2014	SCOPUS
EE89	Gamification in the development of accessible software	2014	SCOPUS
EE90	Requirements compliance as a measure of project success	2013	SCOPUS
EE91	Towards recognizing and rewarding efficient developer work patterns	2013	SCOPUS
EE92	Evaluating simulation software components with player rating systems	2013	SCOPUS
EE93	A systematic literature review of empirical evidence on computer games and serious games	2012	SCOPUS
EE94	The effect of varied game-based learning systems in engineering education: An	2011	SCOPUS

	experimental study		
EE95	Games' usability and learning - the educational videogame BeTheManager!	2011	SCOPUS
EE96	Agile & Kanban in coordination	2011	SCOPUS
EE97	Empirical study upon software testing learning with support from educational game	2011	SCOPUS
EE98	A conceptual framework of serious games for higher education: Conceptual framework of the game Innov8 to train students in business process modelling	2010	SCOPUS
EE99	Work in progress - An investigation of varied game-based learning systems in engineering education	2010	SCOPUS
EE100	A model of cross-functional cooperation in software development project teams	2010	SCOPUS
EE101	Protection poker: The new software security "game"	2010	SCOPUS
EE102	Empirical evaluation of an educational game on software measurement	2009	SCOPUS
EE103	To game or not to game?	2009	SCOPUS
EE104	Measuring reuse during the analysis phase of domain framework development	2007	SCOPUS
EE105	Learning lean through lean game - A case from the infrastructure industry	2007	SCOPUS