

Ensino e Aprendizagem de Testes de Software a partir do uso de Jogos: Uma Revisão Sistemática da Literatura

Gabriel Afonso Pantoja da Silva¹, Sandro Ronaldo Bezerra Oliveira¹, Isaac Souza Elgrably²

¹Programa de Pós-Graduação em Ciência da Computação (PPGCC) – Universidade Federal do Pará (UFPA)

Rua Augusto Corrêa, 01, Guamá, Belém – PA – Brasil

²CESUPA - Centro Universitário do Pará

Av. Alcindo Cacela, 1523, Nazaré, Belém – PA – Brasil

bielbj@gmail.com, srbo@ufpa.br, isaacelgrably@hotmail.com

Abstract. *Teaching software testing faces significant challenges, as many students have difficulty understanding its importance and practical application, which can result in superficial and demotivating learning. On the other hand, the use of games has proven to be effective in education by increasing student motivation and engagement. This paper aims to present a systematic literature review on the use of gamification or games for teaching software testing, identifying the best techniques for implementing gamification in software testing. It was possible to identify several ways to apply games to teaching testing, providing some important insights on how to use game elements and mechanics in order to achieve the expected results and cause positive impacts on the user experience.*

Resumo. *O ensino de testes de software enfrenta desafios significativos, pois muitos alunos têm dificuldade em compreender sua importância e aplicação prática, o que pode resultar em aprendizado superficial e desmotivador. Por outro lado, o uso de jogos tem se mostrado eficaz na educação ao aumentar a motivação e o engajamento dos alunos. Esse trabalho objetiva apresentar uma revisão sistemática da literatura sobre o uso de gamificação ou jogos para o ensino de testes de software, identificando as melhores técnicas para implementar jogos em testes de software. Foi possível identificar diversas formas de aplicar jogos ao ensino de testes, fazendo alguns insights importantes sobre como utilizar elementos e mecânicas de jogos de forma a alcançar os resultados esperados e causar impactos positivos na experiência do usuário.*

1. Introdução

O ensino de teste de software é uma parte fundamental na formação de profissionais de engenharia de software, sendo vital para garantir a confiabilidade de sistemas complexos. No entanto, essa disciplina frequentemente apresenta desafios para os estudantes, que relatam dificuldades em absorver conceitos abstratos e em se manter engajados ao longo do aprendizado, especialmente quando submetidos a métodos tradicionais de ensino, mais teóricos do que práticos (Elgrably e Oliveira, 2018).

Garousi *et al.* (2020), em uma revisão da literatura, afirmam que o ensino de testes de software precisa ser mais efetivo, com métodos de aprendizado ativo e prático,

para que os estudantes desenvolvam as competências e habilidades necessárias para o mercado de trabalho. A partir desse contexto, os jogos surgem como uma abordagem promissora para contornar os problemas no aprendizado de testes. Amplamente utilizado para melhorar a motivação e o engajamento, especialmente no ensino superior, o jogo faz uso de elementos, como recompensas e desafios, que são eficazes para aumentar o interesse e o comprometimento dos estudantes nas atividades educacionais (García-López *et al.*, 2023).

Embora os jogos ofereçam benefícios, sua implementação pode ter efeitos mistos. Antonaci *et al.* (2019), em um mapeamento sistemático, identificaram que elementos, como *rankings* e recompensas, apesar de populares, podem reduzir a motivação intrínseca, quando priorizam a competitividade em detrimento da compreensão do conteúdo. A presença de *rankings* pode criar um ambiente excessivamente competitivo, impactando a autoestima e o desejo de aprender de estudantes em posições inferiores. Esses aspectos evidenciam a necessidade de um planejamento cuidadoso ao implementar o jogo para equilibrar engajamento e aprendizado eficaz.

Este artigo tem como objetivo mostrar os resultados obtidos a partir da execução de uma Revisão Sistemática da Literatura (RSL) para identificar propostas de jogos que ensinam testes de software e também discutir e analisar os resultados obtidos, buscando entender melhor como o uso de jogos pode impactar no ensino de testes de software. A contribuição deste trabalho está em fornecer uma análise sistemática dos elementos de jogos e seus impactos no aprendizado de testes de software, contribuindo para práticas educacionais que maximizem o engajamento sem comprometer a compreensão dos conceitos.

Para fins desta revisão, consideram-se como "jogos" todas as propostas educacionais que apresentam objetivos explícitos, regras estruturadas, mecânicas de interação e *feedback* sistemático ao jogador, mesmo que se autodenominem "gamificação". Tal abordagem visa garantir a coerência da análise e respeitar a predominância dessas características nos estudos selecionados.

Além desta introdução, o artigo está estruturado da seguinte forma: a Seção 2 analisa alguns trabalhos correlatos; a Seção 3 descreve o protocolo usado na RSL; a Seção 4 apresenta os resultados da RSL; a Seção 5 discute os achados; e, finalmente, a Seção 6 contém as considerações finais, limitações e sugestões para futuros trabalhos.

2. Trabalhos Relacionados

O uso de gamificação no ensino de testes de software tem sido explorado por diversas revisões sistemáticas que destacam os benefícios e desafios dessa abordagem educacional. Díaz *et al.* (2020) realizaram uma revisão sistemática sobre a aplicação de gamificação na educação em engenharia de software, analisando como a introdução de elementos de jogo pode aumentar o engajamento dos alunos e a compreensão de conceitos complexos. Essa abordagem é especialmente relevante para o ensino de testes de software, pois a gamificação tem o potencial de transformar um conteúdo tradicionalmente teórico em experiências práticas e dinâmicas.

Além disso, Hassan *et al.* (2021) examinaram o impacto da gamificação no ensino de engenharia de software, destacando as metodologias de ensino mais eficazes e os resultados percebidos pelos alunos. Embora seu foco não seja exclusivamente em

testes de software, os *insights* sobre o uso de jogos e dinâmicas de gamificação são cruciais para entender como essas práticas podem ser aplicadas ao ensino de testes.

Em uma abordagem mais específica, Hussain *et al.* (2021) conduziram uma revisão sistemática sobre a gamificação no ensino de testes de software, abordando diretamente o impacto dessa estratégia no aprendizado dos alunos e os tipos de mecânicas mais eficazes. A revisão evidenciou a importância de escolher corretamente os elementos de jogos para equilibrar o engajamento com a aprendizagem efetiva, evitando sobrecarga cognitiva ou desmotivação.

Finalmente, Li *et al.* (2019) realizaram uma revisão sistemática focada exclusivamente na gamificação em testes de software, fornecendo uma visão detalhada sobre como jogos e outros elementos lúdicos podem ser integrados ao ensino de conceitos de testes. Seus achados reforçam a importância de uma aplicação cuidadosa de gamificação, com base nas necessidades do público-alvo e nos objetivos educacionais desejados.

Essas revisões sistemáticas fornecem uma base sólida para a implementação de gamificação no ensino de testes de software, revelando a importância de escolher os elementos e as mecânicas adequadas para promover tanto o aprendizado quanto a motivação dos alunos. A presente pesquisa busca expandir esses achados ao analisar como diferentes abordagens de gamificação impactam no ensino de testes de software.

3. Materiais e Métodos

Para a execução da RSL, seguiu-se os passos definidos em (Kitchenham, Budgen e Brereton, 2015): (i) Planejamento da Revisão, a partir da identificação das questões de pesquisa, definição da estratégia de pesquisa, termos, palavras-chave, *strings* de pesquisa, critérios para inclusão e exclusão de estudos e conjunto de dados para a extração em estudos relevantes; (ii) Condução da Revisão, a partir da identificação e extração dos dados de estudos relevantes, aplicação dos critérios de seleção estabelecidos e sintetização dos dados obtidos; e (iii) Revisão da Documentação, com o desenvolvimento, validação e elaboração de um relatório de revisão.

A presente pesquisa tem como objetivo identificar na literatura estudos e trabalhos que abordem a utilização de gamificação ou jogos no ensino de teste de software e analisar a utilização dos elementos e das mecânicas de jogos presentes nas propostas. Para formalizar o objetivo deste estudo, foi utilizado o método *Goal-Question-Metric* (GQM) definido por Basili (1992). Assim, este estudo tem como objetivo: **Analisar** - jogos e gamificação que ensinam testes de software; **Com o propósito de** - entender a forma usada para ensinar testes de software; **Com respeito ao** - aumento da capacitação dos envolvidos; **Do ponto de vista de** - docentes e discentes; **No contexto de** - cursos/disciplinas de Tecnologia da Informação.

Para definir as questões de pesquisa, foi utilizado o modelo PICOC (População, Intervenção, Comparação, Resultado e Contexto), que facilita a formulação de perguntas de pesquisa estruturadas e focadas. Assim, para esta pesquisa definiu-se: **População** - discentes; **Intervenção** - identificar propostas existentes de jogos que ensinam testes de software; **Comparação** - não se aplica; **Resultado** - capacitar discentes para realizarem testes de software; **Contexto** - acadêmico. A partir disso, foram definidas as seguintes questões de pesquisa principal (QP) e secundárias (QS): **QP.** Tem como capacitar discentes para realizarem testes de software utilizando jogos

no contexto acadêmico? **QS1.** Quais iniciativas de jogos voltadas para o ensino de testes de software? **QS2.** Quais elementos de jogos os estudos encontrados utilizam? **QS3.** Quais mecânicas de jogos os estudos encontrados utilizam? **QS4.** Como esses jogos foram avaliados? **QS5.** Qual o contexto de aplicação desses jogos? **QS6.** Quais os efeitos percebidos pelos alunos diante da implementação dos diferentes elementos de gamificação?

Os repositórios utilizados para a busca e seleção dos estudos foram as bibliotecas digitais da IEEE Xplore, Scopus e Web Of Science, escolhidas por suas relevâncias e pela ampla cobertura de estudos retornados a partir da *string* definida. A ACM Digital Library não foi incluída por restrição de acesso via Portal da CAPES. No entanto, parte de seus trabalhos é indexada pela Scopus, e a quantidade e diversidade dos estudos obtidos foi considerada suficiente para atender os objetivos desta revisão. Por meio desses repositórios e utilizando o acesso institucional foi possível obter acesso aos conteúdos, com a possibilidade de acesso na íntegra dos trabalhos. Com base nos objetivos e nas questões de pesquisa, foram desenvolvidas as seguintes *strings* de busca: em inglês - (*gamification OR game*) AND (*teaching OR education OR learning*) AND (*software*) AND (*test**); e em português - (*gamificação OR jogo*) AND (*ensino OR educa* OR aprend**) AND (*software*) AND (*test**).

Para selecionar os estudos, foram definidos os critérios de inclusão e exclusão que mais se encaixam na proposta e nos objetivos da pesquisa, a saber: **Critérios de Inclusão** - (i) estudos com resultados da avaliação da aplicação da abordagem de ensino de teste de software, (ii) pesquisas com participantes reais: estudantes e/ou profissionais de testes de software, (iii) trabalhos sobre ensino de testes de software usando jogos ou gamificação; e **Critérios de Exclusão** - (i) trabalhos duplicados, (ii) trabalhos indisponíveis, (iii) trabalhos que não estejam disponíveis em inglês ou português, (iv) trabalhos que não sejam *full paper* ou *short paper*.

A metodologia de documentação usada nas etapas da revisão usou a ferramenta de apoio Parsifal (<https://parsif.al/>) para o armazenamento digital de estudos, referências e anotações dos pesquisadores, além da ferramenta Google Planilhas para o armazenamento dos dados extraídos a partir dos estudos selecionados na revisão.

4. Resultados da RSL

A seleção dos estudos foi realizada em fases, com o intuito de aplicar os critérios de inclusão e exclusão para a obtenção dos estudos relevantes e alinhados à proposta do trabalho. Assim, os resultados obtidos foram: (i) Número de estudos retornados nas bases a partir da *string* = 30818; (ii) Número de estudos não duplicados = 28446; (iii) Número de estudos após a análise do título e resumo = 131; e (iv) Número de estudos após a aplicação dos critérios de inclusão e exclusão = 28. Os resultados mostram que a maior parte dos artigos foram removidos após a aplicação da fase (ii), indicando que a *string* de busca retornou muitos estudos que não eram relevantes para o contexto da pesquisa, porém, o número de estudos que passaram dessa fase foi suficientemente satisfatório para dar continuidade à execução da RSL. A lista dos estudos selecionados, juntamente com os códigos (EP – Estudo Primário) usados para evidenciar as respostas às questões de pesquisa, encontra-se em <https://zenodo.org/records/14940597> e no Quadro 1 do Apêndice deste trabalho.

Alguns estudos incluídos (EP16, EP17, EP18 e EP27) não possuem foco exclusivo no ensino de testes de software, mas abordam o tema de forma relevante em

partes de suas propostas, como atividades ou conteúdos relacionados a testes. Já os estudos EP05, EP09, EP15 e EP20 tratam da mesma proposta, o jogo Code Defenders, e por isso são analisados em conjunto quando aplicável.

As subseções a seguir apresentarão os resultados das extrações de dados realizadas ao longo da condução da RSL para responder cada Questão Secundária (QS) definida no protocolo, mostrando os achados obtidos em cada Estudo Primário (EP). Quanto à Questão Principal (QP), esta será respondida/discutida na Seção 5.

4.1. QS1. Quais iniciativas de jogos voltadas para o ensino de testes de software?

Essa questão buscou entender quais as diferentes iniciativas de jogos que são utilizadas para o ensino de testes. O objetivo é mapear essas iniciativas para entender melhor como elas foram utilizadas para melhorar a qualidade do ensino de teste nas propostas.

No EP01 tem-se um jogo de simulação, onde o aluno interpreta um gerente de testes além de alguns momentos agir como um testador de softwares ao fazer as interações com os funcionários testadores. O gerente pode gastar recursos (orçamento e tempo) para treinar seus funcionários, com isso recebendo perguntas mais fáceis. O objetivo do jogo é responder 10 perguntas no tempo máximo de 10 minutos. Já no EP02 há um jogo 2D estilo plataforma, onde o jogador avança por níveis que correspondem às técnicas de teste de software. Cada nível possui várias fases e para passar cada fase o jogador deve realizar as ações específicas de cada uma.

No EP03 é apresentado um jogo que utiliza atividades interativas, como desafios de adivinhação de conceitos, *drag-and-drop* para categorização, captura de conceitos corretos e exploração de caminhos em mapas para resolução de problemas. Após cada desafio, os jogadores recebem pontuação e *feedback* detalhado para aprimorar seu aprendizado, além de poderem competir por posições em uma tabela de classificação. O jogo é acessível via navegador, permitindo acesso simultâneo a múltiplos usuários e oferecendo aprendizado personalizado no ritmo do jogador, com foco em uma experiência prática e envolvente.

No EP04 há um jogo de cartas para ser jogado em grupo, onde os jogadores utilizam cartas para resolver desafios relacionados a testes de software. O EP05, EP09, EP15 e EP20 são estudos relacionados a mesma proposta de jogo, o *code defenders*, que é um jogo competitivo entre equipes, onde uma equipe ataca e a outra defende. A equipe atacante cria mutantes, enquanto a equipe defensora escreve testes unitários que detectam esses mutantes. Já o EP06 apresenta um jogo no estilo *tower defense*, onde o jogador coloca minas mágicas ao longo do caminho, com o objetivo de impedir a passagem dos *critters mutantes*, deixando passar somente os *critters* normais.

No EP07 tem-se um jogo estilo *tower defense*, onde cada torre representa uma pergunta feita pelo jogador, onde para derrubar a torre o jogador atacante precisa responder a pergunta corretamente. O EP08 apresenta um jogo que consiste em uma espécie de *battle royale* em rodadas, onde em cada uma os jogadores desenvolvem códigos e testes unitários que vão ser executados no código dos outros jogadores, ganhando uma pontuação de acordo com o desempenho do seu código e dos seus testes. Já no EP10 apresenta uma plataforma web onde os jogadores competem para resolver problemas específicos da plataforma, em duelos entre jogadores.

No EP11 há um jogo focado no aprendizado de testes funcionais e especificações de testes. O jogo é um *puzzle* de labirinto, onde o jogador deve usar

raciocínio para escolher os caminhos do labirinto, que é a representação de um fluxograma. Cada bifurcação no caminho percorrido pelo jogador significa um caso de teste diferente, onde o objetivo do jogador é realizar todos os casos de testes possíveis com o menor número de tentativas. No EP12 é mostrado o *gTest Learning*, que é um jogo *single-player* (jogo para somente um jogador), em estilo plataforma, onde o jogador é desafiado a derrotar inimigos a partir de pequenos questionários em fase de turno, adicionando uma técnica de diálogo usada em jogos de RPG.

No EP16 há um jogo *multiplayer*, onde um grupo de jogadores tenta ser o primeiro a concluir um projeto de software, sendo orientados pelas cartas que eles compram. No EP17 tem-se o jogo *coverbot*, onde o jogador controla um personagem que tem como objetivo derrotar os inimigos de cada nível. Ele precisa derrotar os inimigos na menor quantidade de tentativas possível. Já no EP18 tem-se o *bughunter*, um jogo onde os alunos são propostos a realizar tarefas específicas, como responder perguntas, resolver problemas, e com isso melhorar o seu avatar. O EP19 e EP22 objetiva ensinar testes exploratórios, sendo ambientado em um universo de piratas, onde os alunos realizam atividades de testes, e recebem um *feedback* sobre seus resultados.

No EP20 há um jogo competitivo entre equipes, onde uma equipe ataca e a outra defende. A equipe atacante cria mutantes, enquanto a equipe defensora escreve testes unitários que detectam esses mutantes. O EP21 propõe uma solução para o aprendizado de testes de software utilizando software embarcado, hardwares que imitam dispositivos de IoT (*Internet of Things*) reais. O objetivo dos alunos é encontrar *bugs* na VU-SmartHome, um ambiente que imita uma casa inteligente, feito com lego. Já no EP23 a proposta é dividida em 3 partes: o jogo, que introduz os conceitos de testes unitários; uma videoaula, que aprofunda e fixa os conceitos aprendidos anteriormente; e, por fim, uma atividade prática, que serve como desafio para avaliar o aprendizado.

No EP24 os alunos estão na 'corrida pela imortalidade' e criam casos de teste para encontrar falhas em programas defeituosos. A eficácia desses testes determina o progresso dos alunos na "corrida". O EP25 trata de um jogo 2D, estilo plataforma, onde o jogador avança por níveis que correspondem às técnicas de teste de software. Cada nível possui várias fases, e para passar cada fase o jogador deve realizar as ações específicas de cada uma. No EP26 há um jogo onde, de forma lúdica, o jogador resolve vários desafios relacionados a testes unitários. No EP27 o jogo foi pensado para criar um ambiente competitivo durante as aulas, onde os alunos eram pontuados de acordo com as atividades em sala e comportamento. Por fim, no EP28 tem-se um jogo baseado no fluxo de controle (CFG) e o objetivo é ajustar os valores de entrada até que os '*bubbles*' fiquem verdes. A proposta tem como objetivo aumentar a velocidade na cobertura da geração de testes

Quanto aos resultados, não foi possível obter resultados muito concretos devido a grande diversidade de iniciativas dentro dos jogos, são jogos bem diferentes e cada um tem suas particularidades. Contudo, é possível correlacionar as iniciativas e os tipos de teste que são abordados dentro dos jogos, como por exemplo os estudos EP07, EP12 e EP22 são jogos focados em testes unitários e utilizam iniciativas e mecânicas de jogos pensados para serem jogados por 1 jogador. Por outro lado, os estudos EP06, EP09, EP15 e EP20 são focados em testes de mutação, que utilizam iniciativas e mecânicas de jogos *multiplayer* ou de duelo. Esses levantamentos apontam que para tipos diferentes de teste de software, algumas iniciativas podem ser mais adequadas que outras, como

por exemplo para testes exploratórios, jogos com a iniciativa de tentativa e erro podem se adequar mais.

4.2. QS2. Quais elementos de jogos os estudos encontrados utilizam?

Essa questão busca entender e mapear os elementos de jogos encontrados dentro das propostas analisadas e permite construir uma base sólida para entender como elementos de jogo são aplicados ao ensino de testes de software e seus resultados, oferecendo tanto um panorama do estado atual quanto *insights* práticos para o desenvolvimento de futuras iniciativas gamificadas.

A pontuação, observada nos estudos EP01, EP02, EP03, EP04, EP05, EP06, EP07, EP08, EP09, EP10, EP11, EP13, EP14, EP15, EP18, EP19, EP20, EP22, EP24, EP25, EP26, EP27 e EP28, pode ser considerada um elemento central no jogo, pois pode dar um mecanismo de *feedback* imediato além de incentivar o engajamento e a progressão dos participantes. A pontuação pode fomentar uma competição saudável, aumentando o interesse e a dedicação ao conteúdo, ajudando os participantes a entenderem suas habilidades e identificarem áreas de melhoria. O *ranking*, encontrado nos estudos EP01, EP07, EP08, EP12, EP18, EP24 e EP27, também é um elemento que pode gerar um retorno sobre o desempenho dos alunos dentro da proposta, porém ele acaba gerando mais competição dentro do ambiente de ensino. É um elemento que pode estimular o desejo de melhoria contínua, pois os jogadores buscam alcançar posições mais altas no *ranking*, oferecendo visibilidade aos esforços individuais e destacando o progresso em relação aos outros.

Já os avatares, aplicados nos estudos EP01, EP02, EP03, EP08, EP12, EP14, EP17, EP18, EP19, EP23, EP25 e EP26, são frequentemente utilizados no jogo como uma forma de personalizar a experiência dos participantes. Eles podem servir como representações visuais dos usuários dentro de um sistema gamificado, permitindo interação com o ambiente ou outros jogadores. Em algumas aplicações, avatares são configuráveis, oferecendo opções de personalização que podem variar desde aparência até habilidades específicas. A utilização desse elemento pode estar associada a objetivos como imersão, identificação com o contexto ou simplesmente o aumento do engajamento visual. O *storytelling*, observado nos estudos EP06, EP08, EP13, EP15, EP26 e EP27, é um elemento poderoso no jogo, pois conecta os participantes emocionalmente ao conteúdo ao criar narrativas imersivas e significativas. Ele fornece um contexto para as ações, tornando as tarefas mais envolventes e compreensíveis, ao mesmo tempo que aumenta a motivação intrínseca ao associar desafios e recompensas a um enredo envolvente.

Os níveis, aplicados nos estudos EP01, EP02, EP03, EP25 e EP28, são elementos de progressão que organizam o conteúdo ou desafios em etapas crescentes de dificuldade, oferecendo um caminho claro para o avanço e o domínio de habilidades. Ao atingir um novo nível, os participantes sentem-se recompensados e motivados a continuar explorando e superando obstáculos. Esse elemento também pode servir para estruturar o aprendizado, colocando o aluno em contato com os conteúdos no momento certo.

A competição, observada em EP05 e EP09, é um elemento que desperta a motivação extrínseca ao criar desafios entre os participantes, estimulando-os a atingir o melhor desempenho. Ela promove engajamento, pois transforma tarefas comuns em um ambiente competitivo, onde o progresso é valorizado em comparação aos outros. Além

disso, esse elemento também pode aparecer por causa de outros, como por exemplo o ranking, que acaba gerando competição. Os itens, aplicados nos estudos EP08, EP12 e EP22, são elementos tangíveis ou virtuais que representam conquistas, recompensas ou recursos dentro de uma experiência gamificada. Eles podem ser utilizados como ferramentas de progresso, símbolos de *status* ou incentivos para engajamento. Dependendo do design do sistema, a implementação de itens dentro de um jogo pode ser feita de forma muito variada e impactar significativamente na interação do aluno com a proposta, tornando ela mais rica e interativa.

Como é possível notar, alguns elementos são bem mais recorrentes que outros, como é o caso da pontuação e avatar. Esse dado serve para mostrar que alguns elementos podem ser mais interessantes dependendo do objetivo do jogo, por exemplo, a pontuação pode dar uma clara visualização do desempenho do aluno dentro do conteúdo que está sendo abordado no jogo, além disso também pode servir como elemento motivador, já que o aluno pode sentir vontade de alcançar pontuações mais altas. Outro exemplo é o elemento avatar, que é a representação do aluno dentro do jogo, que serve muito bem para criar uma certa imersão e prender a atenção dos alunos.

4.3. QS3. Quais mecânicas de jogos os estudos encontrados utilizam?

Essa questão busca entender quais as diferentes mecânicas de jogos que são utilizadas dentro do jogo para o ensino de testes. O objetivo é mapear essas mecânicas para entender melhor como elas podem ser utilizadas para melhorar a qualidade do ensino de teste nas propostas.

Igualmente ao caso das iniciativas, as mecânicas vistas nos estudos são bastante variadas, o que dificulta chegar a conclusões mais certeiras sobre quais as mecânicas são melhores ou mais eficazes. Por outro lado, fazendo a mesma relação das iniciativas com os tipos de testes abordados nos trabalhos, como por exemplo nos estudos EP04, EP09, EP15 e EP20 onde o tipo são abordados os testes de mutação, também são utilizadas mecânicas de “combate”, onde dois jogadores ou duas equipes disputam a vitória.

Os trabalhos EP02, EP03, EP06, EP14, EP17 e EP26 utilizam a mecânica de níveis, que pode ser utilizada para estruturar o progresso dos participantes em etapas crescentes de dificuldade, criando um senso de conquista ao dividir objetivos maiores em metas menores e mais alcançáveis, mantendo o engajamento ao oferecer recompensas ou desbloquear novos conteúdos à medida que os participantes avançam.

A mecânica de jogos de cartas é amplamente utilizada em jogos por sua versatilidade, capacidade de criar interatividade e promover estratégia. Essa mecânica aparece nos trabalhos EP04, EP13, EP16 e EP26. As cartas podem representar recursos, ações, desafios ou habilidades, sendo usadas para construir estratégias, resolver problemas ou competir com outros participantes. Além disso, cartas podem ser utilizadas para apresentar conceitos de forma visual e prática, enquanto promovem a colaboração ou competição. Essa mecânica, ao mesmo tempo simples e adaptável, torna o aprendizado ou a experiência mais envolvente e desafiadora.

4.4. QS4. Como esses jogos foram avaliados?

O objetivo dessa questão é levantar quais os diferentes métodos de avaliação abordados, para entender quais são os métodos mais adequados para colocar dentro de uma futura proposta de jogo a ser feita.

Dos estudos lidos, a grande maioria realizou um estudo controlado para avaliar e medir a eficácia da proposta. Outros também fizeram essa avaliação por meio de questionários, que seguiam a escala de likert. Esses dois métodos de avaliação mostraram-se adequados ao contexto de jogos, já que o objetivo do jogo é não só melhorar o aprendizado, mas também melhorar o engajamento do aluno.

A escala de Likert, usada nos estudos EP01, EP04, EP05, EP09, EP11, EP14, EP15, EP17 e EP23, é amplamente utilizada na avaliação de propostas de jogos devido a sua capacidade de medir percepções, opiniões e níveis de concordância dos usuários de maneira quantitativa. Em um contexto gamificado, essa escala permite capturar dados subjetivos relacionados à usabilidade, engajamento, motivação e eficácia da experiência, fornecendo *insights* valiosos sobre a percepção dos participantes.

O estudo experimental, aplicado nos estudos EP14, EP16, EP17, EP18, EP19, EP22, EP24 e EP28, é uma metodologia essencial na avaliação de propostas de jogos, pois permite investigar causalidades ao manipular variáveis e observar seus efeitos em um ambiente controlado. Em um estudo desse tipo, os participantes geralmente são divididos em grupos, como grupo experimental (que interage com a proposta gamificada) e grupo de controle (que utiliza métodos tradicionais ou não gamificados). Essa abordagem possibilita isolar o impacto direto dos elementos de jogos, como *rankings*, níveis ou *feedback*, sobre métricas como aprendizado, motivação ou desempenho.

4.5. QS5. Qual o contexto de aplicação desses jogos?

Essa questão tem como objetivo entender qual o contexto das pessoas que testaram a aplicação, se foram alunos de alguma disciplina específica da graduação, alunos de pós-graduação, profissionais que já atuam na área de testes e etc. Foi possível perceber que a grande maioria (EP01, EP03, EP04, EP08, EP09, EP11, EP14, EP16, EP21, EP22, EP23, EP26 e EP28) dos estudos eram voltados para alunos da graduação de cursos de Engenharia de Software e Ciência da Computação, principalmente nas disciplinas de engenharia de software e qualidade de software.

4.6. QS6. Quais os efeitos percebidos pelos alunos diante da implementação dos diferentes elementos de jogos?

O objetivo dessa questão é entender quais os efeitos sentidos pelos usuários ao utilizar as propostas mostradas nos estudos. É importante saber como o aluno sentiu-se ao participar da jogo, se o assunto ficou mais fácil, e sentiu-se mais motivado a aprender.

Os EP05, EP09 e EP15 são diferentes estudos sobre o mesmo jogo, o *Code Defenders*, que enfatizam que o uso dos elementos servem para ter os efeitos de melhorar o aprendizado e satisfação, aumentar a motivação e prender a atenção. É possível observar que todos os jogos onde o elemento pontuação não está presente, não houve relatos de melhora na motivação, enquanto nos jogos onde a pontuação foi utilizada, todos tiveram melhora na motivação. Portanto, esse elemento pode incentivar o aluno a participar mais da dinâmica proposta, seja com o objetivo de alcançar pontuações mais altas ou simplesmente melhorar suas pontuações anteriores.

Nos estudos EP11 e EP14, que são os únicos que utilizam o elemento *storytelling*, é possível notar que eles não têm mecânicas de *multiplayer* (Ex: duelos, turnos). Nesse sentido, pode-se afirmar que em jogos projetados para serem jogados por um único jogador, elementos de imersão podem ajudar a facilitar o aprendizado e tornar

a dinâmica mais divertida. No estudo EP17, para a avaliação da proposta foi criada uma versão do jogo onde foi retirado da interface toda a gamificação, sistema de combate, pontuação, progressão de nível e vida. Portanto, é seguro afirmar que os efeitos percebidos pelos alunos estão diretamente ligados aos elementos utilizados.

Os trabalhos EP16 e EP23 utilizam somente competição como elemento de jogos e em ambos houve melhora no aprendizado e divertimento, além de prender a atenção dos usuários. Isso reforça a utilização desse elemento para obter os efeitos citados. O trabalho EP01 utiliza algumas mecânicas diferentes dos outros trabalhos (simulação e gerenciamento de recursos) e mesmo assim consegue alcançar os mesmos efeitos vistos nos trabalhos anteriores. Isso pode indicar que diferentes combinações de elementos e mecânicas podem gerar os mesmos resultados.

Um trabalho não citado anteriormente, o EP18, utilizou os elementos avatar, pontos, medalhas e *ranking*, sendo percebida uma piora (em comparação com os métodos tradicionais) tanto no aprendizado quanto na satisfação dos alunos. Isso pode ocorrer pelo fato de que o jogo não possui nenhuma mecânica.

5. Discussão

A análise dos resultados revelou que o jogo é uma ferramenta poderosa, com grande potencial para impactar positivamente o ensino de testes de software no contexto de alunos de TI (Tecnologia da Informação). Ela permite a combinação de diferentes mecânicas e elementos, possibilitando a criação de abordagens gamificadas diversas e com focos específicos. Por exemplo, o trabalho EP06 utiliza mecânicas de defesa de torre para abordar testes de mutação, enquanto o trabalho EP13 emprega mecânicas de jogos de cartas para ensinar testes unitários. Essa flexibilidade destaca como diferentes *designs* podem ser adaptados para atender a objetivos educacionais distintos, ampliando as possibilidades de engajamento e aprendizado.

Porém, embora o jogo tenha um grande potencial para aprimorar o ensino, é necessário selecionar cuidadosamente os elementos de jogos adequados para evitar resultados negativos nos efeitos percebidos pelos alunos, como foi o caso do EP18. A escolha incorreta de mecânicas ou elementos pode levar a experiências desmotivadoras, causar frustração ou até mesmo desviar o foco dos objetivos de aprendizado. Por exemplo, elementos competitivos, como *rankings*, podem gerar ansiedade ou desinteresse em alunos menos confiantes, enquanto desafios excessivamente complexos podem desmotivar participantes que ainda estão desenvolvendo habilidades básicas.

Portanto, respondendo a QP dessa pesquisa, os jogos podem sim ser usados para ensinar testes de software. Sua aplicação permite transformar conceitos muitas vezes abstratos e desafiadores em experiências interativas, dinâmicas e engajantes, facilitando o aprendizado e promovendo maior retenção de conteúdo. A combinação de diferentes elementos e mecânicas de jogo, quando bem planejada, pode atender a vários objetivos educacionais, adaptando-se às necessidades e aos perfis dos alunos.

6. Conclusão

Como evidenciado ao longo deste trabalho, o uso de jogo demonstrou ser uma ferramenta eficaz para o ensino de testes de software, abordando os desafios tradicionais do aprendizado nessa área, como a complexidade de conceitos que muitas vezes são difíceis de compreender sem o uso de abordagens mais dinâmicas.

Este estudo destacou que é possível ensinar testes de software de maneira envolvente e prática a partir do uso de jogos, além de explorar diferentes formas de combinar elementos e mecânicas de jogos para alcançar esse objetivo. Entretanto, nem todos os trabalhos analisados forneceram respostas completas para as questões levantadas neste estudo, o que limitou algumas conclusões a uma parte específica dos estudos avaliados.

Como continuidade da pesquisa, o objetivo para futuros trabalhos é utilizar as informações obtidas nesta revisão sistemática para desenvolver uma proposta gamificada que potencialize o ensino de testes de software.

Agradecimentos

Os autores gostariam de agradecer a CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pela concessão de bolsa de mestrado para realização desta pesquisa.

Referências

- Antonaci, A.; Fischer, H.; Graf, S. (2019) “The use of gamification in learning management systems: A systematic review”. *International Journal of Educational Technology in Higher Education*, v. 16.
- Basili, V. R.; Weiss, D. M. (1984) “A Methodology for Collecting Valid Software Engineering Data”. *IEEE Transactions on Software Engineering*, v. 10, n. 6, p. 728–738.
- Díaz, C.; Romero, L.; Sampedro, J. (2020) “Gamification in Software Engineering Education: A Systematic Review”. *International Journal of Engineering Education*, v. 36, n. 4.
- Elgrably, I. S.; Oliveira, S. R. B. (2018) “Gamification and Evaluation of the Use of Agile Tests in Software Quality Subjects: The Application of Case Studies”. 13th International Conference on Evaluation of Novel Approaches to Software Engineering, Funchal, Portugal.
- García-López, J.; Pérez-López, C.; Sánchez-Ruiz, A. (2023) “Gamification in Software Testing Education: Enhancing Motivation and Learning Outcomes”. *Journal of Educational Technology Society*, v. 26, n. 3, p. 145–158.
- Garousi, V.; Felderer, M.; Mesa, D. A. (2020) “Software test maturity assessment and test process improvement: A systematic literature review”. *Information and Software Technology*, v. 123.
- Hassan, S.; Azeem, M.; Raza, S. (2021) “The Role of Gamification in Teaching Software Engineering: A Systematic Review”. *Software Engineering and Knowledge Engineering*, v. 19.
- Hussain, S.; Zhang, Y.; Wong, K. W. (2021) “A Systematic Review of the Impact of Gamification on Software Testing Education”. *Journal of Software: Evolution and Process*, v. 33, n. 3.
- KitchenhamITCHENHAM, B.; BUDGEN, D.; BRERETON, P. *Evidence-Based Software Engineering and Systematic Reviews*. CRC Press, Boca Raton, FL, 2015.
- Li, X.; Tang, W.; Su, C. (2019) “Systematic Review on the Use of Gamification in Software Testing Education”. *Journal of Computing Sciences in Colleges*, v. 34.

Apêndice

Quadro 1. Estudos Selecionados na RSL

ID	Estudo
EP01	Oliveira, B.; Afonso, P. J.; Costa, H. (2016) TestEG: A Computational Game for Teaching Software Testing. <i>35th International Conference of the Chilean Computer Science Society (SCCC)</i> .
EP02	Valle, P. H. D.; Rocha, R. V.; Maldonado, J. C. (2017) Testing Game: An Educational Game to Support Software Testing Education. <i>SBES '17: Proceedings of the XXXI Brazilian Symposium on Software Engineering</i> .
EP03	Ribeiro, T. P. B.; Paiva, A. C. R. (2015) iLearnTest: Educational Game for Learning Software Testing. <i>2015 10th Iberian Conference on Information Systems and Technologies (CISTI)</i> .
EP04	Beppé, T. A.; De Araújo, I. L.; Santos, I. S., et al. (2018) GreaTest: A Card Game to Motivate the Software Testing Learning. <i>Conferência de Educação Tecnológica</i> .
EP05	Fraser, G.; Gambi, A.; Kreis, M.; Rojas, J. M. (2019) Gamifying a Software Testing Course with Code Defenders. <i>SIGCSE '19: The 50th ACM Technical Symposium on Computer Science Education</i> .
EP06	Straubinger, P.; Caspari, L.; Fraser, G. (2023). Code Critters: A Block-Based Testing Game. <i>2023 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)</i>
EP07	Kiron, N.; Vassileva, J. (2019) Tower of Questions: Gamified Testing to Engage Students in Peer Evaluation. <i>ITS Workshops</i> .
EP08	Materazzo, A.; Fulcini, T.; Coppola, R.; Torchiano, M. (2023) Survival of the Tested: Gamified Unit Testing Inspired by Battle Royale. <i>2023 IEEE/ACM 7th International Workshop on Games and Software Engineering (GAS)</i> .
EP09	Rojas, J. M.; White, T. D.; Clegg, B. S.; Fraser, G. (2017) Code Defenders: Crowdsourcing Effective Tests and Subtle Mutants with a Mutation Testing Game. <i>IEEE/ACM 39th International Conference on Software Engineering (ICSE)</i> .
EP10	Tillmann, N.; De Halleux, J.; Xie, T.; Bishop, J. (2013) Pex4Fun: A Web-Based Environment for Educational Gaming via Automated Test Generation. <i>28th IEEE/ACM International Conference on Automated Software Engineering (ASE)</i> .
EP11	Severo, J.; Lelli, V. (2023) Testing Maze: An Educational Game for Teaching Functional Testing. <i>SBES '23: Proceedings of the XXXVII Brazilian Symposium on Software Engineering</i> .
EP12	Silva, C. G.; De Farias Junior, I.; Teixeira, M.; De Aquino, C. (2016) gTest Learning: Um Jogo para Ensino Básico de Teste de Software. <i>Proceedings of the Regional Conference on Educational Technologies</i> .
EP13	Marabesi, M.; Silveira, I. F. (2019) Towards a Gamified Tool to Improve Unit Test Teaching. <i>XIV Latin American Conference on Learning Technologies (LACLO)</i> .
EP14	Hsueh, N. L.; Xuan, Z. H.; Daramsenge, B. (2023) Design and Implementation of Gamified Learning System for Mutation Testing. <i>International Journal of Information and Education Technology</i> , v. 13, n. 7, p. 1150–1155.

ID	Estudo
EP15	Rojas, J. M.; Gordon, F. (2016) Code Defenders: A Mutation Testing Game. <i>2016 IEEE Ninth International Conference on Software Testing, Verification and Validation Workshops (ICSTW)</i>
EP16	Baker, A.; Navarro, E. O.; Van Der Hoek, A. (2003) Problems and Programmers: An Educational Software Engineering Card Game. <i>25th International Conference on Software Engineering</i> .
EP17	Sherif, E.; Liu, A.; Nguyen, B.; Lerner, S.; Griswold, W. G. (2020) Gamification to Aid the Learning of Test Coverage Concepts. <i>IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T)</i> .
EP18	De Jesus, G. M.; Ferrari, F. C.; Paschoal, L. N.; Souza, S. R. S. (2019) Is It Worth Using Gamification on Software Testing Education?. <i>SBQS '19: Proceedings of the XVIII Brazilian Symposium on Software Quality.?</i>
EP19	Ferreira Costa, I. E.; Bezerra Oliveira, S. R. (2020) The Use of Gamification to Support the Teaching and Learning of Software Exploratory Testing. <i>IEEE Frontiers in Education Conference (FIE)</i> .
EP20	Fraser, G.; Gambi, A.; Rojas, J. M. (2018) A Preliminary Report on Gamifying a Software Testing Course with the Code Defenders Testing Game. <i>CECSEE '18: Proceedings of the 3rd European Conference of Software Engineering Education</i> .
EP21	Silvis-Cividjian, N.; Visser, G.; Veltman, J., et al. (2023) House of the Rising Flames: A Hands-on, Bug-Centered Tutorial on Embedded Software Testing. <i>ITiCSE 2023: Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 2</i> .
EP22	Costa, I. E. F.; Oliveira, S. R. B. (2019). A Systematic Strategy to Teaching of Exploratory Testing using Gamification. <i>ENASE 2019: Proceedings of the 14th International Conference on Evaluation of Novel Approaches to Software Engineering</i>
EP23	Gomes, R. F.; Lelli, V. (2022) GAMUT: GAME-based learning approach for teaching Unit Testing. <i>SBQS '21: Proceedings of the XX Brazilian Symposium on Software Quality</i> .
EP24	Blanco, R.; Trinidad, M.; Suárez-Cabal, M. J., et al. (2023) Can Gamification Help in Software Testing Education? Findings from an Empirical Study. <i>Journal of Systems and Software</i> .
EP25	Valle, P. H. D.; Toda, A. M.; Barbosa, E. F.; Maldonado, J. C. (2017). Educational Games: A Contribution to Software Testing Education. <i>2017 IEEE Frontiers in Education Conference (FIE)</i>
EP26	Marabesi, M.; Silveira, I. F. (2020) Evaluation of Testable, a Gamified Tool to Improve Unit Test Teaching. <i>14th International Technology, Education and Development Conference</i>
EP27	Elgrably, I. S.; Oliveira, S. R. B. (2018) Gamification and Evaluation of the Use of Agile Tests in Software Quality Subjects: The Application of Case Studies. <i>13th International Conference on Evaluation of Novel Approaches to Software Engineering</i> , Funchal, Portugal.
EP28	Moosavi, S.; Haghighi, H.; Sahabi, H., et al. (2019) Greenify: A Game with the Purpose of Test Data Generation for Unit Testing. <i>Lecture Notes in Computer Science</i> .