

SOFTWARE CITY - JOGO EDUCACIONAL EM MUNDO VIRTUAL 3D PARA APOIAR O ENSINO E APRENDIZAGEM DE CONCEITOS DE ENGENHARIA DE SOFTWARE

Nikollas Kauê Soares Gomes 

Universidade Federal de Santa Maria
ngomes@inf.ufsm.br

Giliane Bernardi 

Universidade Federal de Santa Maria
giliane.bernardi@ufsm.br

Andre Zanki Cordenonsi 

Universidade Federal de Santa Maria
andre.cordenonsi@ufsm.br

Jonathan Donato Pippi 

Universidade Federal de Santa Maria
jonathan.pippi@gmail.com

Resumo: O ensino de Engenharia de Software pode ser considerado um dos principais balizadores para a formação de profissionais na área de computação. Constituída por um arcabouço de metodologias, práticas e teorias, a disciplina tem sido tradicionalmente apresentada para os alunos de forma pouco atrativa e bastante teórica. Buscando mitigar estes problemas, este artigo apresenta o desenvolvimento de um jogo educacional, do tipo quiz game, imerso em um Mundo Virtual 3D (metaverso), como uma forma alternativa para trabalhar os conceitos de Engenharia de Software, com ênfase nos aspectos mais teóricos. Como principais resultados, destaca-se a interação e exploração do Mundo Virtual, além da possibilidade de ampliação escalar do jogo a partir da inserção de novas perguntas.

Palavras-chave: Jogos Educacionais; Mundos Virtuais 3D; Metaverso; Engenharia de Software.

SOFTWARE CITY – AN EDUCATIONAL 3D VIRTUAL WORLD GAME TO SUPPORT TEACH AND LEARN SOFTWARE ENGINEERING CONCEPTS

Abstract: Teach Software Engineering can be considered one of the main targets in the computation professional's education process. Constituted by a framework of methodologies, practices and theories, the discipline has traditionally been presented in an unattractive way to students. So, to mitigate these problems, this paper presents the development of an educational quiz game, embedded in a metaverse. This game intends to represent an alternative way to teach the Software Engineering concepts, emphasizing the more theoretical aspects of the area. As main results, we highlight the interaction and exploration of the Virtual World, as well as the possibility of scaling the game through the insertion of new questions.

Keywords: Educational Games, 3D Virtual Worlds, Metaverse, Software Engineering

Introdução

A Engenharia de Software (ES) envolve um conjunto de práticas que guiam o desenvolvimento de sistemas computacionais, abrangendo as teorias e conhecimentos necessários para se obter um software de qualidade e que satisfaça os requisitos dos usuários e clientes (PRESMANN, 2016). Ao considerar o âmbito acadêmico, é necessário formar futuros profissionais capacitados em prover tais soluções, no entanto, o que se observa é que, em decorrência de muitos conteúdos teóricos, bem como das dificuldades em trabalhar os mesmos por meio de diferentes abordagens, a dinâmica acaba envolvendo, em sua maioria, aulas expositivas (ROSA *et al.*, 2021; PORTELA *et al.*, 2017; ARAÚJO *et al.*, 2016; SILVA *et al.*, 2011), que tendem a ter um envolvimento menor do estudante no processo de aprendizagem. Destaca-se, assim, a importância de trabalhar as teorias que envolvem o processo de desenvolvimento de software aliando às práticas necessárias. Para Mendes *et al.* (2019), “é fundamental o sucesso no processo de aprendizagem dos conteúdos da mesma para a boa formação dos estudantes que serão futuros profissionais engenheiros de software”.

No que tange à importância de obter os conhecimentos necessários sobre as atividades que envolvem o desenvolvimento de software, bem como as dificuldades inerentes ao processo de ensino e aprendizagem dos conteúdos mais teóricos, o que se busca é a diversificação das estratégias para trabalhar tais conceitos, necessários para a formação do estudante. Entre estas estratégias, pode-se citar os jogos educacionais, que têm tido cada vez mais destaque na área de Engenharia de Software (ROSA *et al.*, 2021; LEMOS *et al.*, 2020; PETRI *et al.*, 2017). Os jogos são considerados propostas metodológicas que podem auxiliar na construção de conhecimentos (SOUZA e FRANÇA, 2016), sendo que, por meio dos mesmos, torna-se possível envolver o estudante no processo de aprendizagem, objetivando a compreensão das teorias e conceitos que envolvem o desenvolvimento de software de maneira mais lúdica e, na maioria dos casos, deixando os estudantes mais motivados e satisfeitos.

Buscando esta maior motivação, o projeto proposto neste artigo busca, ainda, integrar jogos com Mundos Virtuais (MV) 3D, como possibilidade de potencializar a imersão e envolvimento do estudante no processo. Tais ambientes virtuais têm ganho maior destaque

e se popularizado recentemente a partir do anúncio do Metaverso proposto por Mark Zuckerberg, criador do Facebook e que, recentemente, em outubro de 2021, inclusive mudou o nome de sua organização para Meta em referência a seu novo posicionamento de mercado, de unir mundo real e virtual. No entanto, pesquisas discutindo o uso destes ambientes no meio educacional já permeiam o meio acadêmico há mais de uma década, com o uso de plataformas como o Second Life¹ e o OpenSimulator². Para Herpich *et al.* (2020), uma das vantagens do uso de MV é a possibilidade de extrapolar os limites da sala de aula física, incentivando sentimentos de presença mesmo quando os estudantes estão geograficamente dispersos, bem como a possibilidade de simulação de situações reais ou fictícias para o ensino de técnicas e conteúdo, por meio do uso de avatares, interação com objetos, personagens artificiais e conteúdos educacionais. Reforçando as potencialidades de MV, Krassmann *et al.* (2017) afirmam que a utilização de tais ambientes vem sendo ampliada, principalmente “devido às possibilidades trazidas por esses ambientes, como imersão aos conteúdos, interação síncrona de usuários e simulação de situações reais, facilitando abordagens ativas e atividades de experimentação”.

Considerando todas as colocações expostas acima, o objetivo do presente artigo é apresentar o desenvolvimento de um jogo educacional imerso em um Mundo Virtual 3D, denominado *Software City*, que tem como foco central trabalhar de forma diferenciada os principais conceitos que envolvem a disciplina de Engenharia de Software. Para tanto, a próxima seção destaca alguns trabalhos correlatos envolvendo jogos e Mundos Virtuais 3D aplicados à educação, em especial com foco em Engenharia de Software; na sequência, é apresentada a metodologia e desenvolvimento do jogo proposto; a penúltima seção apresenta a aplicação e avaliação dos resultados; e, por fim, tem-se a última seção, que discute as considerações finais deste trabalho e tece algumas considerações para sua continuidade.

Trabalhos Correlatos

A busca por trabalhos correlatos envolveu a pesquisa de jogos voltados para o ensino e aprendizagem de Engenharia de Software, bem como aplicações educacionais imersas em Mundos Virtuais. Um exemplo de jogo voltado para o ensino de conceitos de Engenharia de

Software é o SimulES – Simulador de Engenharia de Software (BRITO *et al.*, 2018), um jogo de tabuleiro e cartas, podendo ser jogado por até oito pessoas simultaneamente na mesma partida, com foco em conceitos de ES, que simula o processo de desenvolvimento de software. Segundo os autores, o jogo se mostrou uma boa opção para apoiar o processo de ensino e aprendizagem de gerência de projetos de software, “fazendo com que os alunos se sintam em um ambiente de trabalho real e exerçam as funções pertinentes a um gerente de software” (BRITO *et al.*, 2018, p. 10).

Em Lemos *et al.* (2020) é detalhado o desenvolvimento do jogo AD-RPG, desenvolvido para apoiar o ensino e aprendizagem de levantamento de requisitos de software. O jogo, do gênero RPG (Role Playing Game), foca em duas técnicas específicas de elicitación de requisitos, a entrevista e a análise de documentos, conduzindo o jogador/estudante por uma narrativa na idade média, fazendo com o que mesmo resolva desafios e ajude a população usando as habilidades destas duas técnicas. Segundo os autores, a partir de uma avaliação usando o modelo MEEGA+, foi possível constatar que os estudantes se sentiram motivados e divertiram-se com o jogo, sentindo-se confiantes com sua própria aprendizagem. Os autores ainda destacam que “o jogo AD-RPG se apresenta promissor como ferramenta complementar para o ensino e aprendizagem de elicitación de requisitos, trazendo ao mesmo tempo uma atividade educacional e lúdica” (LEMOS *et al.*, 2020, p. 1).

Uma proposta mais antiga, mas que merece menção por tratar especificamente de conceitos de Engenharia de Software, é o jogo A Ilha dos Requisitos (GONÇALVES *et al.*, 2011). Nesse jogo, o estudante assume o papel de Jack Reqs, um analista de requisitos que, após sofrer um acidente, cai em uma ilha com 2 tribos rivais, além de um vulcão prestes a entrar em erupção e destruir a ilha e seus habitantes. O objetivo do jogador é solucionar os desafios propostos, negociando conflitos com as tribos, por meio dos conhecimentos na área de Engenharia de Requisitos de software. Por fim, destaca-se o EAREq-Game (PETRI e CHIAVEGATTI, 2015), um RPG *single-player* que trabalha conceitos de levantamento e priorização de requisitos. Nele, o estudante atua como o engenheiro de requisitos Ralph que precisa coletar, organizar e priorizar requisitos, com a finalidade de desenvolver um sistema acadêmico. O jogo foi avaliado sob os aspectos de motivação, experiência do usuário e aprendizagem, sendo que os resultados foram considerados satisfatórios.

Considerando o uso de metaversos ou Mundos Virtuais, um exemplo de jogo imerso em um MV é o apresentado por Rosa *et al.* (2021), voltado para apoiar o ensino e aprendizagem de levantamento de requisitos, em específico, a técnica de entrevista. O jogo, do gênero simulação, denominado LevReq3D, foi desenvolvido utilizando a plataforma OpenSimulator e simula o dia-a-dia de um engenheiro de software, que precisa coletar requisitos para a informatização de um sistema para uma pizzaria. Dois mundos virtuais foram criados, um para representar a empresa de software e outro para representar o espaço da pizzaria, sendo que o aluno se teletransporta entre os dois espaços, interagindo com funcionários da pizzaria para a coleta dos requisitos. Os funcionários são agentes de software que foram implementados como *chatbots*, possuindo uma base de conhecimento que permite ao jogador, por meio da entrevista, coletar informações e elicitar os requisitos funcionais para o desenvolvimento do sistema. Segundo os autores, como principais resultados da avaliação do jogo em aulas de Engenharia de Software, destacam-se a facilidade de uso, o potencial para aprendizagem dos conteúdos e as regras claras no que se refere às diretrizes para aprendizagem da técnica de coleta de dados por meio de entrevista.

Outro exemplo é o trabalho de Silva *et al.* (2011), no qual foi desenvolvido um jogo denominado JETS (Jogo da Equipe de Teste de Software), para trabalhar habilidades e conhecimentos da área de testes. O jogo simula a atuação de um funcionário em uma empresa de software, imersa em um ambiente virtual. Nesta empresa, o jogador tem acesso ao setor de testes de software, onde pode trabalhar com os problemas e situações que ocorrem nesta área, atuando em diferentes papéis, desde um testador a gerente da equipe de testes. Ainda considerando o uso de mundos virtuais na educação, tem-se outros exemplos, não voltados para Engenharia de Software, mas que podem ser destacados por abordarem o uso de jogos educacionais imersos em tais ambientes. É o caso do MV apresentado em Krassmann *et al.* (2022), que tem como objetivo apoiar o ensino de Matemática Financeira; do MV destacado em Maceu e Sousa (2020), visando o ensino de Sistemas Operacionais; e, por fim, um MV desenvolvido para servir de instrumento à educação patrimonial, buscando resgatar a identificação do sujeito com o patrimônio histórico que o cerca, apresentado em Mendonça *et al.* (2020).

Todos os trabalhos analisados trouxeram subsídios para o projeto e desenvolvimento do jogo apresentado neste artigo, corroborando com a premissa de que jogos educacionais

imersos em mundos virtuais podem colaborar com um maior engajamento dos estudantes, apresentando os conteúdos de uma forma mais lúdica e envolvente.

Aspectos Metodológicos

O quadro 1 apresenta o percurso metodológico que norteou o desenvolvimento do jogo apresentado, que iniciou com uma revisão de literatura acerca dos temas centrais que percorrem este projeto, para depois conduzir às etapas de desenvolvimento e avaliação da proposta.

Quadro 1 – Percurso metodológico do projeto

ETAPAS	ATIVIDADES DESENVOLVIDAS
1. Revisão Bibliográfica	a. Estudo sobre Engenharia de Software (ES) b. Estudo sobre Jogos Educacionais e Mundos Virtuais (MV)
2. Análise de Trabalhos Correlatos	a. Análise de jogos voltados para o ensino de ES b. Análise de aplicações envolvendo MV
3. Modelagem e Desenvolvimento do Mundo Virtual (MV) e do Jogo	a. Elaboração do enredo do jogo b. Definição da arquitetura do MV c. Modelagem e importação de objetos para o MV d. Definição das questões de ES e. Implementação dos scripts para as interações com o <i>Quiz Game</i> f. Implementação do script para o teletransporte g. Implementação do script para o desafio final
4. Avaliação e Análise dos Resultados	a. Definição do modelo de avaliação e planejamento da execução b. Aplicação do jogo c. Análise e discussão dos resultados

Fonte: dos autores

Conforme já mencionado, o projeto partiu da a etapa de revisão bibliográfica, que buscou aprofundar os conhecimentos dos pesquisadores sobre Engenharia de Software e o uso de jogos e Mundos Virtuais no contexto educacional, explorados brevemente na introdução deste artigo. Na sequência, foram pesquisados trabalhos correlatos que pudessem agregar conhecimentos, traçar um panorama de desenvolvimento atual envolvendo as duas temáticas, bem como subsidiar a presente proposta. Tais trabalhos foram explorados na seção anterior.

Nas próximas subseções, serão detalhadas as atividades realizadas na etapa 3, de Modelagem e Desenvolvimento, sendo que a avaliação e os resultados são discutidos em seção separada, na sequência desta. Destaca-se que a ferramenta escolhida para desenvolver o ambiente virtual 3D foi a OpenSimulator, por ser uma ferramenta gratuita e de código aberto, que conta com um grande número de alternativas relacionadas a funcionalidades, cenários e possibilidades de interação. Além disso, esta plataforma já é utilizada pelo grupo de pesquisa no qual este trabalho está inserido, o que contribuiu para esta decisão. Para a etapa de avaliação, o jogo foi aplicado junto à estudantes de cursos da área de computação que, após interagirem com o mundo e jogo, responderam a um questionário de avaliação de jogos, adaptado de SAVI (2011).

Enredo do Jogo

O jogo inicia com o jogador visualizando parte da história do jogo, que é apresentada por meio de painéis na entrada da cidade. Por meio desses, o jogador descobre que uma equipe de desenvolvedores de software criou uma cidade tecnológica, chamada *Software City*. Dentre os componentes desta equipe, destaca-se o Engenheiro de Software Lex, que é considerado uma das mentes mais brilhantes em sua área, porém, de difícil personalidade. Paranoico, Lex acredita que seu trabalho é menosprezado pelos membros da equipe e decide sabotar o projeto em que estava trabalhando, programando uma bomba em Software City. Para salvar a cidade, é necessário inserir, no computador localizado no subterrâneo da cidade, um número que o jogador receberá em formato binário. Esse número está separado em dez fragmentos espalhados pelos quinze computadores da cidade. Para obter cada fragmento, o jogador precisa responder a uma pergunta relacionada a área de Engenharia de Software. Ou seja, das quinze perguntas, cinco não possuem uma parte do código, acrescentando o fator aleatoriedade e sorte, o que é considerado um aspecto importante em jogos.

O jogador, após ser apresentado à história, é orientado a desarmar a bomba da cidade, sem saber quais prédios contêm partes do código, sendo obrigado a explorar o cenário até localizar os dez números do código para, ao final, convertê-lo em um número decimal e inseri-lo no computador do subterrâneo da cidade. Se o aluno errar o código, a bomba explode e ele

perderá o jogo automaticamente. Acertando o número, o jogador desarma o sistema central, salva a cidade e ganha o jogo.

Arquitetura e Modelagem dos Objetos

A arquitetura da cidade foi definida a partir da história do jogo; sendo assim, o modelo de cidade definido para o jogo foi de um ambiente de negócios, envolvendo um número significativo de prédios, para retratar o aspecto de uma empresa de grande porte, localizada em uma grande cidade. A busca pelo cenário principal foi realizada no repositório Zadaroo³, onde é possível encontrar construções e objetos para montar um ambiente virtual, disponibilizados gratuitamente. Além dos prédios, o principal cenário é o do subterrâneo da cidade, onde está localizada a bomba. Para implementar este cenário, foi criada uma nova ilha no ambiente virtual e projetado o subterrâneo. O jogador é automaticamente teletransportado para o mundo subterrâneo quando acertar as 10 questões do jogo que contém os fragmentos do código. A figura 1 apresenta uma visão da cidade de *Software City*.

Figura 1 – Cenário principal da cidade de *Software*



Fonte: dos autores

O cenário do subterrâneo foi moldado a partir da criação de objetos dentro do ambiente virtual, que receberam texturas de bombas e dinamites. As mesmas foram reproduzidas em grande quantidade pelo local, forçando o jogador a acreditar que se encontrava realmente no subterrâneo da cidade. A intenção era apresentar um cenário de alta tensão e drama. A inserção dos objetos no ambiente virtual foi realizada por meio de um visualizador, ferramenta necessária para acessar o servidor do OpenSimulator. Para este projeto, foi utilizado o visualizador de MV Imprudence⁴. A figura 2 apresenta uma visualização do subterrâneo de *Software City*.

Figura 2 – Subterrâneo de *Software City*



Fonte: dos autores

Após realizar a criação da cidade e do subterrâneo, passou-se para a etapa de inserção dos demais objetos e elementos para compor o ambiente, de acordo com as características relatadas no enredo do jogo. Escritórios com computadores, móveis, sofás, tapetes, cadeiras e armários também foram inseridos, com o objetivo de aumentar a imersão do jogador na história apresentada. Com relação aos computadores, estes receberam cuidado especial, pois representam o ponto nevrálgico da interação. É com os computadores que o estudante precisa interagir para responder as questões dos desafios.

Definição das Questões de Engenharia de Software

Conforme descrito anteriormente, o jogo desenvolvido se baseia em desafios, no estilo quiz game, envolvendo questões da área de Engenharia de Software. Assim, a próxima etapa, após a definição do ambiente, foi a seleção das questões que iriam compor o jogo. As mesmas foram selecionadas com base em um repositório de questões disponibilizado por uma professora de Engenharia de Software da universidade onde esta pesquisa foi desenvolvida, que utiliza tais questões em suas aulas. A escolha das questões foi balizada por assuntos que, usualmente, são trabalhados de forma mais teórica, enfatizando a possibilidade de uma exploração didática alternativa através do jogo. Após a escolha das perguntas que seriam utilizadas na aplicação, estas foram modeladas em forma de imagens. Essas imagens, ao serem inseridas no jogo, foram utilizadas como texturas em objetos criados dentro do Mundo Virtual. A figura 3 apresenta um exemplo de questão inserida no jogo, com opções de resposta certa (3a) e errada (3b).

Figura 3 – Exemplo de apresentação e resolução de questões



Fonte: dos autores

As questões encontram-se ocultas no ambiente, localizadas dentro de objetos no formato de computadores que estão espalhados pelo Mundo Virtual. Ao clicar nestes computadores, o objeto com a questão e suas respectivas alternativas é exibido ao jogador.

Ao lado de cada pergunta, é exibido um pequeno objeto na forma de um cubo, no qual foi inserido um script que possibilita sua interação com o usuário. Para marcar a alternativa desejada, basta o estudante clicar no objeto correspondente. A próxima subseção discute o funcionamento dos scripts.

Desenvolvimento dos Scripts

As interações necessárias para a etapa de resolução das questões são baseadas em dois scripts principais, sendo um exclusivo para as alternativas erradas, e outro para as alternativas corretas. Cada questão possui cinco alternativas, sendo que, destas, quatro são erradas e apenas uma certa. Inicialmente, para contabilizar o número de acertos e erros do jogador, foi implementada uma conexão do Mundo Virtual com um banco de dados MySQL, onde foi criada uma tabela contendo informações, como um identificador para o jogador, o nome do avatar do mesmo e o número de acertos e erros associado ao mesmo. A implementação feita no Mundo Virtual acessa um programa desenvolvido em PHP, responsável pela conexão com o banco criado.

Os scripts, independentemente de tratarem de acertos e erros, possuem funções referentes a identificar o usuário que está interagindo com o objeto dentro do jogo. Ao ser executado, o script coleta informações do avatar que o executou e checa no banco de dados se ele já está registrado. Em caso afirmativo, os valores do banco são atualizados, dependendo da ação do jogador, seja para incrementar acertos ou erros, para checar se o jogador conseguiu responder corretamente as perguntas do jogo ou se ele atingiu o limite de erros e perdeu o jogo. A figura 4 apresenta um trecho do script que verifica se o estudante excedeu o número de erros e, com isso, perdeu o jogo, ou se apenas o contador de erros deverá ser incrementado.

Figura 4 – Script de verificação de erros

```
http_response( key request_id, integer status, list metadata, string body )
{
  // Recebe a resposta da base de dados - se existir
  if(request_id == requestKey)
  {
    integer resp = 1;
    resp = (integer) body;

    //Significa que já atingiu os 10 erros
    if( resp == 0 ) {

      list my_list = llParseString2List(avatarName,[" "],["."]);
      llSay(0, "Você cometeu 10 erros e perdeu o jogo. Será expulso da cidade. Game Over!!!");
      llSleep(5.0);
      osKickAvatar((string)llList2String(my_list, 0), (string)llList2String(my_list, 1), "SoftwareCity");
    }
    //Significa que não foi atingido o limite de erros.
    else
    {
      llSay(0, "Voce errou a alternativa. Cuidado ein. Uma cidade inteira depende de você!!!");
    }
  }
}
```

Fonte: dos autores

Já a figura 5 apresenta um trecho do script que verifica se o estudante atingiu o número total de acertos e pode ser teletransportado para o subterrâneo, para resolver o desafio final.

Figura 5 – Script de verificação de acertos e possível teletransporte

```
if( (integer)resp == 1)
{
  //isto significa que já atingiu os 10 acertos

  llSay(0, "Parabéns, você acertou todos os desafios e agora será teletransportado para os subterrâneos da cidade, onde poderá nos ajudar a desarmar a bomba e evitar que SoftwareCity exploda!");
  llSleep(5.0);

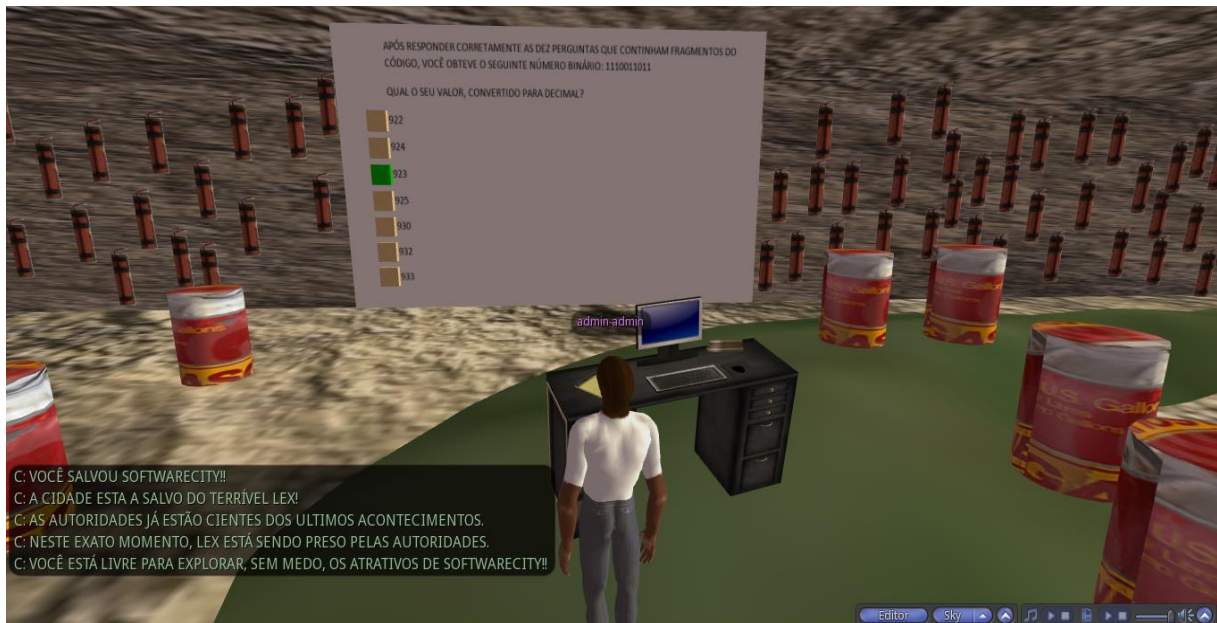
  //***** Script de teletransporte *****
  osTeleportAgent(toucher, "bomba", LandingPoint, LookAt);
}
else
{
  llSay(0, "Parabéns, voce acertou! Digito binário desta pergunta: 1");
  llSay(0, "CONTINUE PROCURANDO ANTES QUE A BOMBA EXPLODA!");
}
```

Fonte: dos autores

A pergunta final conta com scripts semelhantes aos usados na superfície da cidade. A diferença é que os scripts não realizam checagem no banco de dados. O script das alternativas erradas, automaticamente, executa a exclusão do jogador e encerramento da aplicação, informando, por meio de uma mensagem e efeito sonoro, que as bombas irão explodir e não

foi possível salvar a cidade do vilão. Se a alternativa certa for executada, o jogador é informado por uma mensagem e, após, é teletransportado para a superfície de *Software City*, para poder explorar a cidade, após ter salvo a mesma. Uma imagem desta cena pode ser visualizada na figura 6.

Figura 6 – Cena de Vitória do jogo



Fonte: dos autores

O teletransporte final foi usado porque, com o jogo em andamento, o jogador pode não ter explorado com calma o ambiente criado, podendo assim, ao final, realizar a ação com mais calma e aproveitar os recursos disponíveis da aplicação, respondendo novamente questões que errou, circulando pela cidade e interagindo com o ambiente.

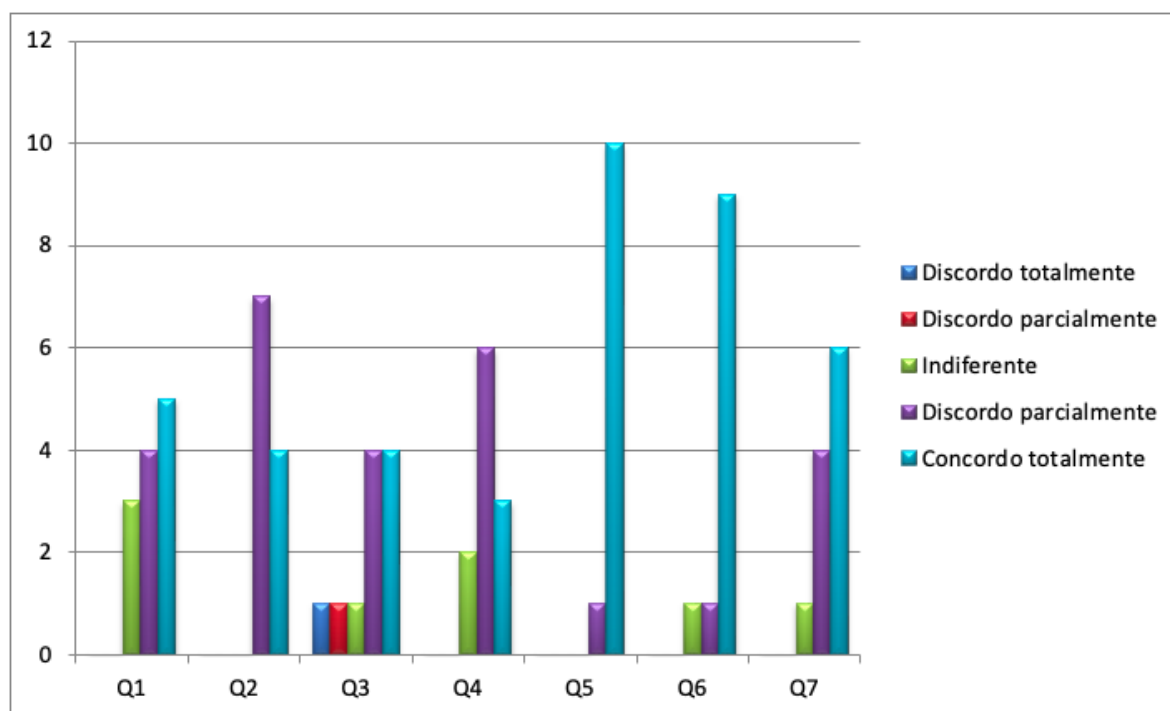
Avaliação e Discussão dos Resultados

Com o desenvolvimento concluído, passou-se para a fase de aplicação e avaliação dos resultados. A avaliação foi realizada com onze estudantes da disciplina de Engenharia de Software dos cursos de Ciência da Computação e Sistemas de Informação de uma universidade pública, que foram orientados a explorar o jogo e responder os desafios que se encontravam no ambiente. Cada estudante, ao fim da execução, respondeu a um questionário

de avaliação, adaptado do modelo MEEGA (SAVI, 2011), envolvendo três subcomponentes centrais: Motivação pelo jogo, Experiência do Usuário e Aprendizagem.

Para análise da motivação, foram disponibilizadas sete perguntas que buscaram coletar dados do jogo que possam ter chamado a atenção do jogador. A intenção era verificar se o jogador/estudante aprendeu o conteúdo e se divertiu ao mesmo tempo. Das perguntas propostas, foram solicitadas as opiniões referentes a elementos do jogo como o cenário, objetos, seu conteúdo, seu funcionamento e o que atraiu a atenção do jogador no início do jogo. A figura 7 apresenta o gráfico com as respostas coletadas.

Figura 7 – Avaliação da motivação do jogo Software City



Legenda – Q1: O design do jogo é atraente (cenário e objetos).

Q2: Houve algo interessante no início do jogo que capturou minha atenção.

Q3: O conteúdo do jogo é relevante para os meus interesses.

Q4: O funcionamento do jogo está adequado ao meu jeito de aprender.

Q5: Foi fácil entender o jogo e começar a utilizá-lo como material de estudo.

Q6: Ao passar pelas etapas do jogo senti confiança de que estava aprendendo.

Q7: É por causa do meu esforço pessoal que consigo avançar no jogo.

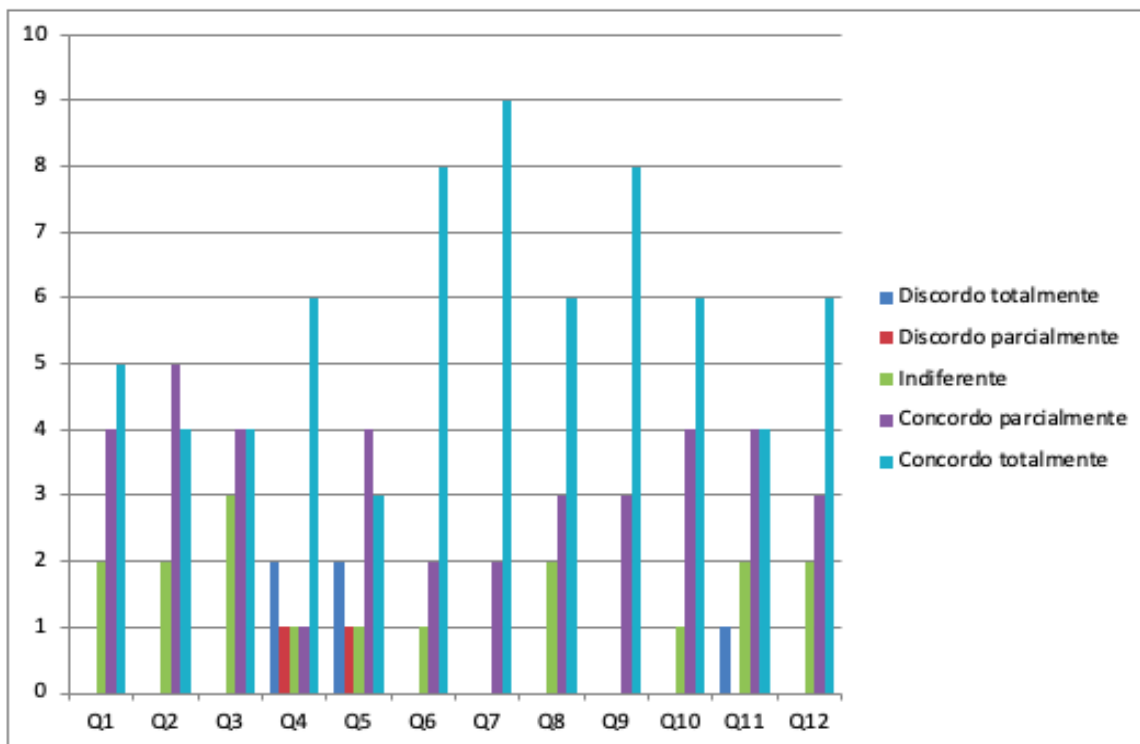
Fonte: dos autores

Dos dados coletados, não houve avaliações negativas quanto ao cenário e objetos do jogo, sendo que o conteúdo do jogo apresentou oito avaliações positivas. O funcionamento do jogo e o nível de dificuldade em relação ao entendimento da dinâmica também foram fatores abordados na pesquisa. Na questão envolvendo o funcionamento da aplicação em relação ao modo de aprender dos jogadores, não houve avaliações negativas. Quase todos os jogadores encontraram facilidade em jogar a aplicação, ainda que não houvesse avaliações negativas. Por fim, foi verificado se os jogadores sentiam que seu esforço era responsável pelo avanço no jogo. A maioria dos jogadores concordou que pelos próprios méritos iam crescendo no jogo.

A etapa de avaliação da experiência do usuário buscou coletar dados referentes ao momento em que o jogador usava a aplicação, explorando fatores externos, como a reação do mesmo à possíveis interrupções e sua reação ao fim da experiência na aplicação, e fatores internos, por exemplo, se o jogador se divertiu durante o jogo. Foram doze perguntas nesta etapa, e delas, destaca-se o fato de alguns jogadores, ao usarem o jogo, não interagiram com outros jogadores, visto que não foi possível realizar os testes simultaneamente, algo que é visível nas respostas de algumas perguntas do questionário nesta etapa. Quando perguntados sobre divertirem-se com outros jogadores e sobre interagirem com os mesmos, houve respostas negativas. Já quando questionados sobre divertir-se dentro do jogo, não houve avaliações negativas. O mesmo pode-se dizer sobre as perguntas quanto ao nível das questões e se o jogo seria recomendado a outras pessoas pelos usuários. Sobre as perguntas referentes ao nível de foco na aplicação durante os testes, a maioria apresentou avaliações positivas quanto a este fator. A figura 8 apresenta o gráfico resultante das avaliações dos usuários.

A terceira parte da avaliação envolveu questões referentes à aprendizagem. Nesta etapa, foram feitas três perguntas que indagavam, sob o ponto de vista dos estudantes, se consideravam que o jogo poderia contribuir com a aprendizagem de conteúdos teóricos de Engenharia de Software. Por fim, foi disponibilizado um espaço dissertativo para que expusessem, caso desejassem, destaques positivos encontrados no jogo, bem como pontos que possam ser melhorados em uma futura versão do jogo. Das perguntas, foi concluído que a aplicação trouxe, à maioria dos usuários, uma boa experiência quanto a abordagem do conteúdo, e que estes puderam obter, por meio de uma experiência nova, uma forma alternativa de explorar o conteúdo proposto.

Figura 8 – Avaliação da experiência do jogador no jogo Software City



Legenda – Q1: Temporariamente esqueci das minhas preocupações do dia-a-dia, fiquei totalmente concentrado no jogo.

Q2: Eu não percebi o tempo passar enquanto jogava, quando vi o jogo acabou.

Q3: Me senti mais no ambiente do jogo do que no mundo real, esquecendo do que estava ao meu redor.

Q4: Pude interagir com outras pessoas durante o jogo.

Q5: Me diverti junto com outras pessoas.

Q6: Este jogo é adequadamente desafiador para mim, as tarefas não são muito fáceis nem muito difíceis.

Q7: Me diverti com o jogo.

Q8: Quando interrompido, fiquei desapontado que o jogo tinha acabado (gostaria de jogar mais).

Q9: Eu recomendaria este jogo para meus colegas.

Q10: Gostaria de utilizar este jogo novamente.

Q11: Consegui atingir os objetivos do jogo por meio das minhas habilidades.

Q12: Tive sentimentos positivos de eficiência no desenrolar do jogo.

Fonte: dos autores

Das observações positivas dos estudantes, ressalta-se os fatores da dinâmica dentro do mundo, ou seja, o fato de poder explorar um ambiente para interagir com o quiz game, que foi bem aceito. Isso vai ao encontro dos resultados obtidos em Rosa *et al.* (2021), onde o

questão sensação de presença e imersão foram destaque na avaliação e apontados por Krassmann *et al.* (2017), no que se refere a questão da navegação e experimentação.

O enredo foi outro ponto elogiado pelos avaliadores, assim como o modo de explorar as perguntas. Foi citado este ponto por boa parte dos usuários, onde salientam que, por meio do jogo, aprender e diversão tornaram-se aliados. Dos pontos a serem melhorados, alguns estudantes sugeriram que o jogo deveria ser desenvolvido para outras áreas da Engenharia de Software, expandindo o universo de questões, considerando que para esta primeira versão o conjunto de perguntas e respostas era limitado e fixo. Isto faz com que a rejogabilidade não exista, ou seja, depois que os estudantes passaram pelos desafios propostos, percorrer novamente o ambiente não alteraria o conjunto de questões. Outra sugestão mencionada foi a criação de uma sequência obrigatória para responder perguntas, onde uma pergunta leve a outra, por meio de pistas. Na versão atual, o jogo conta com 15 questões a serem respondidas, sem uma ordem predefinida, bastando o estudante localizar as mesmas ao percorrer a cidade. De acordo com a sugestão, as questões passariam a ser apresentadas de acordo com a sequência com que os conteúdos educacionais são apresentados na disciplina.

Analisando os resultados, foi observada a boa aceitação da proposta. Questões referentes ao modo de apresentação do conteúdo, e aliar diversão com o aprender obtiveram êxito na sua execução, aspectos já destacados em Herpich *et al.* (2020) como potenciais para promover o engajamento de estudantes em atividades educacionais. O nível de atenção dedicada à aplicação faz com que os objetivos do jogo tenham sido alcançados, visto que reter a atenção do estudante para que ele possa absorver o conteúdo do jogo era uma das preocupações.

Considerações Finais

Este artigo apresentou o desenvolvimento e avaliação do jogo *Software City*, que tinha como objetivo central promover o ensino e aprendizagem de conceitos teóricos de Engenharia de Software, por meio de recursos interativos e lúdicos, tendo como apoio o uso de Mundos Virtuais 3D.

Com o desenvolvimento do jogo, foi possível abordar os conteúdos da disciplina de uma forma um pouco mais dinâmica, trazendo uma nova experiência a alunos e professores.

O uso de um enredo para estimular o envolvimento do jogador com a aplicação mostrou-se eficaz. O cenário criado para que o usuário pudesse sentir-se parte da história, além dos elementos explorados e a solução de problemas da área foram muito bem recebidos. Dos trabalhos vistos anteriormente, destaca-se no aplicativo a unificação dos conceitos trabalhados em ambientes virtuais e jogos educacionais.

Considerando a boa aceitação dos estudantes que avaliaram a proposta e apontamentos positivos feitos com relação ao formato de apresentação do conteúdo e questões, acredita-se que os objetivos do jogo foram satisfeitos, onde foi possível propor uma estratégia educacional diversificada para trabalhar os conteúdos teóricos da disciplina de Engenharia de Software que, até então, em sua maioria, somente eram trabalhados de forma expositiva.

Para dar seguimento a esta proposta, algumas atividades podem ser elencadas, como o desenvolvimento de um repositório com questões diversas, cadastradas pelo próprio professor. Atualmente, o jogo conta com um conjunto limitado de 15 questões, o que, como destacado nas colocações da avaliação, não incentiva o estudante a jogar novamente. Além disso, pretende-se refinar os elementos/objetos do MV, de forma a torná-lo esteticamente melhores. Por fim, destaca-se a necessidade de nova avaliação com grupos de estudantes, de forma a ampliar o número de contribuições acerca do potencial da proposta descrita neste artigo.

Referências

- ARAÚJO, D. N., CRUZ, M. L. P. DE M., PIMENTEL, J. H., DUQUE, M., ALENCAR, F. Jogos Educativos no Ensino da Engenharia de Requisitos. In: **IX Fórum de Educação em Engenharia de Software (FEES)**, Maringá, 2016.
- BRITO, L. F. *et al.* ANALYSIS OF THE GAME SIMULES IN SUPPORT OF TEACHING AND LEARNING SOFTWARE ENGINEERING. **CADERNOS EDUCACAO TECNOLOGIA E SOCIEDADE**, v. 11, n. 4, p. 680-692, 2018.
- GONÇALVES, R. Q., THIRY, M., ZOUCCAS, A. Avaliação da aprendizagem em experimentos com jogo educativo de engenharia de requisitos. In: **SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE (SBQS)**, 10, Curitiba, 2011. **Anais...** Porto Alegre: SBC, 2011.
- HERPICH, F.; NUNES, F. B.; LIMA, J. V.; TAROUCO, L. M. R. Mundos Virtuais e Realidade Aumentada no âmbito educacional: reflexões e perspectivas. **ReTER**, v.1, n.1, 2020.
- KRASSMANN, A. L.; TAROUCO, L. M. R.; BERCHT, M. Diretrizes para a Promoção do Senso de Presença na Educação a Distância. **Revista Brasileira de Informática na Educação**, v. 30, p. 542–572, 2022.

KRASSMANN, A., FALCADE, A., JARDIM, R., MEDINA, R., BERCHT, M. Um panorama de teses e dissertações brasileiras sobre Mundos Virtuais 3D na educação. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 28, Recife, 2017. **Anais...** Porto Alegre: SBC, 2017.

LEMONS, E. L.; CONCEIÇÃO, M. H.; NUNES, F. B.; MEDINA, R. D.; BERNARDI, G. AD-RPG: Elicitando requisitos na idade média utilizando um jogo educacional. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 31, 2020. **Anais...** Porto Alegre: SBC, 2020.

MACEU, J. S.; SOUSA, V. L. P. Um ambiente virtual imersivo para o ensino de Sistemas Operacionais: concepção e primeiras análises. **RENOTE**, v. 16, n. 1, 2018.

MENDES, J.; COSTA, Y.; FRAZÃO, K.; SANTOS, R.; SANTOS, D.; RIVERO, L. Identificação das expectativas e dificuldades de alunos de graduação no ensino de engenharia de software. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 27, Belém, 2019. **Anais...** Porto Alegre: SBC, 2019.

MENDONÇA, Fabiana Mendes et al. Gamificação na Educação Patrimonial: um Caso Aplicado à Plataforma Opensimulator. **Revista Brasileira de Expressão Gráfica**, v. 8, n. 2, 2020.

PETRI, G., VON WANGENHEIM, C. G., & BORGATTO, A. F. Qualidade de jogos digitais e não digitais utilizados para o ensino de engenharia de software no Brasil. **Revista de Gestão e Avaliação Educacional**, v. 7, n.14, 2017.

PETRI, G., CHIAVEGATTI, N. C. Um role playing game para o ensino de elicitación e análise de requisitos. **RENOTE**, v. 13 n. 1, 2015.

PORTELA, C., VASCONCELOS, A., OLIVEIRA, S. R. B. Um Modelo Iterativo para o Ensino de Engenharia de Software Baseado em Abordagens Focadas no Aluno. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 28, Recife, 2017. **Anais...** Porto Alegre: SBC, 2017.

PRESSMAN, R. S., MAXIM, B. R. **Engenharia de software: uma abordagem profissional**. 8. ed. Porto Alegre: AMGH, 2016.

ROSA, L. H.; BERNARDI, G.; DUARTE MEDINA, R. Mundos virtuais como apoio ao ensino e aprendizagem de engenharia de software: Simulando a técnica de entrevista. **RENOTE**, Porto Alegre, v. 19, n. 1, 2021.

SAVI, R., VON WANGENHEIM, C.G., BORGATTO, A. F. A Model for the Evaluation of Educational Games for Teaching Software Engineering. In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE, São Paulo, 2011. **Anais...** Porto Alegre: SBC, 2011.

SILVA, T. G., BERNARDI, G., MÜLLER, F. M. Abordagem de Apoio ao Ensino e Aprendizagem de Teste de Software Baseada em Jogos Sérios e Mundos Virtuais. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 22, Aracajú, 2011. **Anais...** Porto Alegre: SBC, 2011.

SILVA, T. G., MÜLLER, F. M., E BERNARDI, G. Panorama do Ensino de Engenharia de Software em Cursos de Graduação Focado em Teste de Software: Uma Proposta de Aprendizagem Baseada em Jogos. **RENOTE**, v. 9, n. 2, 2011.

SOUZA, M., FRANÇA, C. O Sucesso dos Jogos para Ensino de Disciplinas de Engenharia de Software sob a Ótica de uma Teoria Motivacional. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 27, Uberlândia, 2016. **Anais...** Porto Alegre: SBC, 2016.

¹ <https://secondlife.com/>

² <http://opensimulator.org/>

³ <https://zadaroo.com/>

⁴ <https://sourceforge.net/projects/imprudence.team-purple.p/>