

ConES: Um RPG como Instrumento de Apoio ao Ensino da Construção de Produto de Software

Anderson dos Santos Guerra^{1,2}, Sandro Ronaldo Bezerra Oliveira¹, Julio Cezar Costa Furtado²

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

²Departamento de Ciências Exatas e Tecnológicas - Universidade Federal do Amapá (UNIFAP), Macapá – AP – Brasil

anderson.guerra@icen.ufpa.br, srbo@ufpa.br, furtado@unifap.br

Abstract. *Software Engineering is one of the subjects that allow students to be able to work in software development. One of the difficulties in relation to its teaching is to identify ways of learning that can motivate students and make them implement the necessary knowledge to be used in the industry. Based on this, this paper presents a role-playing game prototype that can be used as a tool to assist construction of software products teaching. Two evaluations were made with students with different levels of experience in Software Engineering, all of them were undergraduates in Computer Science. The prototype evaluations indicated that the elaborated mechanics are satisfactory for the game to be used in a classroom context, so that it is possible to verify its effectiveness as a teaching support tool.*

Keywords. *RPG, Software Product Construction, Evaluation.*

Resumo. *A Engenharia de software é uma das disciplinas que permite que acadêmicos consigam realizar o desenvolvimento de software. Uma das dificuldades em relação ao seu ensino é identificar formas de aprendizagem que consigam motivar os acadêmicos e fazer com que eles efetivem o conhecimento necessário para ser utilizado na indústria. Com base nisso, este trabalho apresenta um protótipo de um role-playing game que possa ser utilizado como ferramenta para auxiliar no ensino da construção de produtos de software. Foram feitas duas avaliações com estudantes de diferentes níveis de experiência na área de Engenharia de Software, todos cursavam graduação em Ciência da Computação. As avaliações do protótipo indicaram que as mecânicas elaboradas são satisfatórias para que o jogo seja usado em um contexto de sala de aula, para que seja possível verificar a sua eficácia como ferramenta de apoio ao ensino.*

Palavras-Chave. *RPG, Construção de Produto de Software, Avaliação.*

1. Introdução

A pesquisa de Lemos (2019) apresenta alguns problemas que são enfrentados durante o ensino de disciplinas de Engenharia de Software (ES), sendo alguns deles: a falta de interesse na disciplina; o foco excessivo na parte teórica da disciplina; e um foco muito alto em conteúdo ao invés de competência, fazendo com o que o método de ensino não seja tão adequado em relação ao que o mercado profissional exige.

O uso de jogos no ensino faz com que os alunos consigam divertir-se enquanto estão aprendendo determinado assunto. No geral, jogos possuem uma característica lúdica, além de influenciarem de maneira positiva o processo de ensino e aprendizagem, uma vez que o participante tem a capacidade de adquirir novos conceitos, habilidades e conhecimentos por meio da experiência ao longo da execução dessas práticas [Mettler e Pinto 2015]. Partindo de que o uso de jogos pode ser uma importante ferramenta para o auxílio do aprendizado de acadêmicos, este trabalho visa apresentar um protótipo de RPG (*Role-Playing Game*) que pode ser utilizado em sala de aula para auxiliar na efetivação do conhecimento do processo de construção de um software.

O RPG foi selecionado para ser o estilo de jogo do protótipo tomando como base a pesquisa de Guerra *et al.* (2021), que apresenta uma revisão sistemática dos principais estilos de jogos utilizados no ensino de ES. A partir dessa revisão, foi realizada uma análise das principais características que cada tipo de jogo apresentou, verificando quais os pontos positivos e negativos em relação ao uso no processo de ensino aprendizagem. Por fim, foi estabelecido que o RPG é o estilo que mais adequa-se ao objetivo que os pesquisadores querem atingir. Para a confecção do jogo, foi realizada a leitura de dois trabalhos: a pesquisa de Navarro (2006), onde existe um apêndice contendo o que seriam as principais regras existentes dentro do mundo da Engenharia de Software; e a leitura do SWEBOK (*Software Engineering Body of Knowledge*) [Bourque et al. 2014]. Após a leitura foi realizado um mapeamento entre as regras de Navarro e as áreas do conhecimento dentro do SWEBOK, visando verificar em que área cada uma das regras encaixava-se.

Após esse mapeamento, foi decidido trabalhar na criação de um jogo voltado para a área da Construção de Software. Os dois fatores principais que levaram a essa escolha foram: (1) a experiência do pesquisador com área de construção de software; e (2) a construção de software está intimamente ligada com outras duas áreas do SWEBOK (*design* de software e teste de software), pois seu processo envolve uso intensivo das atividades relacionadas ao *design* e ao teste, o que permite que seja possível estabelecer aventuras focadas na construção de software, mas que também abordam (em um grau menor) essas outras duas áreas. Após o mapeamento, o jogo começou a ser elaborado, o sistema de regras GURPS (*Generic Universal RolePlaying System*) [Jackson 2004] foi utilizado para nortear a criação das regras e esse sistema foi escolhido por possuir características que vão desde a abrangência de regras simples a detalhadas. Este sistema foi projetado para dar ênfase no realismo e por encorajar que os jogadores utilizem a interpretação.

Este trabalho está estruturado da seguinte forma: a Seção 2 apresenta a fundamentação teórica; a Seção 3 apresenta alguns trabalhos relacionados; a Seção 4 é usada para descrever o jogo que foi elaborado; a Seção 5 apresenta a forma de avaliação dos protótipos do jogo; a Seção 6 apresenta as conclusões do trabalho.

2. Fundamentação Teórica

O ensino de ES necessita focar tanto em *hard* quanto *soft skills*. Isso significa que competências técnicas como engenharia de requisitos, programação e testes são necessárias, bem como habilidades comunicativas, de cooperação e de pensar criticamente também são desejadas [Ouhbi e Pombo 2020]. Segundo o SWEBOK [Bourque et al. 2014], a Construção de Software é o termo que se refere ao trabalho

detalhado de um software funcional, fazendo uso de uma combinação de codificação, verificação, testes unitários e testes de integração.

Para Nicholson (2013), os jogos de cartas, tabuleiros, computadores, interpretação de papéis e consoles fazem parte do que pode ser considerado jogo. Os pesquisadores Leontiev e Frias (2007) definem como requisito básico para identificar uma brincadeira haver duas características: um objetivo consciente; e uma operação pela qual será efetuada a atividade. Dentre os diversos tipos de jogos, é possível citar o RPG, que é uma narrativa do jogo caracterizada por ser interativa, episódica e participativa [Schmit 2008]. Esse formato de jogo promove a construção de metáforas e representações imaginárias que podem levar à liberdade, ao estímulo ao questionamento, à diálogo entre diferentes tipos de saberes, e ao exercício da diversão [Aragão 2009; Rosa 2010].

Ainda no contexto de jogos de RPG, aventura é o termo comumente utilizado para representar uma missão ou mesmo uma trama simples que acontece durante o jogo de RPG. Ela pode ocorrer em apenas uma sessão de jogo ou pode durar várias sessões. As sessões são gerenciadas pelo mestre, que é o responsável pelo andamento do jogo. Ele faz a escolha das aventuras que serão realizadas, bem como descreve aos jogadores o que está acontecendo, além de avaliar resultados de ações e de ajudar na distribuição de pontos [Jackson 2004].

3. Trabalhos Relacionados

O trabalho de Wei-Fan Chen *et al.* (2008) consiste em um jogo eletrônico em que o participante pode jogar de maneira *singleplayer* (como gerente de projeto) ou *multiplayer* (escolhendo seu papel no processo de desenvolvimento de software). O jogo foi testado com 34 alunos de graduação na Universidade de Tecnologia de Taiwan. Após a execução do experimento, os autores puderam concluir que o jogo apresentou resultados positivos no processo de aprendizado dos estudantes. A pesquisa de Kemell *et al.* (2018) apresenta um jogo que visa auxiliar no ensino introdutório de tópicos de Engenharia de Software e do Essence (SEMAT). O jogo apresentado pode ser considerado um jogo de mesa interpretativo, no qual os participantes assumem o papel de gerente de projeto e de outros papéis dentro do processo de desenvolvimento de software (possuindo seus próprios níveis de *soft skills* e *hard skills*), encorajando o trabalho em time ao invés de competitividade.

O jogo apresentado no trabalho de Silva e Jacome (2017) é descrito como um “livro jogo”, no qual o participante faz parte de uma aventura solo por meio da leitura do livro, com o objetivo de apresentar sobre *eXtreme Programming*. O jogador cria uma ficha de personagem e na sequência começa a sua aventura interpretando um gerente de projetos. O trabalho de Begosso *et al.* (2019) apresenta um RPG eletrônico para colaborar no ensino da metodologia ágil Scrum por meio de uma simulação de processo de desenvolvimento de software. O jogo possui algumas características de RPG de mesa: elementos de narrativa, desenvolvimento de personagem e elemento de imersão.

4. O Jogo

O ConES é um jogo de RPG que possui como objetivo primário ser um instrumento de auxílio no ensino de Engenharia de Software, com foco na área da Construção de Software, de maneira que os participantes envolvidos consigam experimentar algumas dificuldades relacionadas ao processo de desenvolvimento de Software. O jogo baseia-

se, primariamente, na interpretação de papéis e no uso de blocos de montar. A aventura insere os jogadores dentro de um time que foi contratado por um rei para prestar o serviço de construção de um castelo, para isso o time receberá as demandas do rei (requisitos) e deverá utilizar os blocos de montar para satisfazê-las.

Para que os jogadores e professores consigam familiarizar-se com os termos, as dinâmicas e as mecânicas do ConES foi realizada a elaboração de dois manuais que condensam as regras. O primeiro tem o nome de Manual do Jogador, que possui a descrição de todas as regras necessárias para que os jogadores consigam executar uma partida de maneira satisfatória. Já o segundo manual é chamado de Manual do Mestre, onde é possível acessar uma visão resumida das regras dos jogadores e possui orientações específicas para auxiliar o mestre ao longo de uma partida.

O jogo é dividido em uma série de fases, onde, inicialmente, os jogadores devem realizar a criação de seus personagens e, posteriormente, trabalhar na construção do produto para poder entregar as demandas do cliente. Cada entrega consiste em uma construção que foi feita utilizando blocos de montar. Ao longo dessa sessão será feita a explicação dos principais elementos do jogo: Papéis Envolvidos, Fases, Instrumentos Utilizados e por fim as Mecânicas que existem no jogo.

4.1. Papéis Envolvidos e Instrumentos Utilizados

O ConES foi criado com o intuito de que cada jogador consiga atuar de maneira diferente no desenvolvimento de software, onde para isso foi utilizado um sistema de papéis, em que cada jogador pode escolher um papel para interpretar ao longo da execução do jogo.

Os papéis do ConES são divididos em dois tipos: papéis do mestre e papéis dos jogadores. Os papéis do mestre só podem ser interpretados pelo mestre do jogo, enquanto os outros papéis podem ser escolhidos pelos jogadores.

O papel do Cliente é melhor descrito como a pessoa que representa o dono do produto, que sabe quais as regras de negócio e como o produto deve funcionar. O Cliente é um personagem do mestre, que não pode ser escolhido pelos jogadores. O Gerente é o líder do projeto e membro que tem interesse em remover obstáculos para que o time consiga efetuar seu trabalho com sucesso. Atuando como Gerente, o mestre deverá verificar quais são os principais impedimentos dos outros jogadores e tentar resolver esses problemas. Assim como o Cliente, o mestre é responsável por fazer o papel do Gerente.

O jogador que interpretar o Arquiteto é responsável por fazer a arquitetura das demandas do Cliente, de forma que ela fique bem estruturada para que seja possível ser feita a sua implementação. O Arquiteto deve utilizar a ferramenta LeoCAD para fazer o desenho da construção que utiliza blocos de montar. Esse desenho será a fonte que os Construtores irão utilizar para fazer a construção da demanda do Cliente.

O Construtor tem como função implementar a demanda que foi arquitetada, devendo fazer uso de bons métodos e práticas para conseguir implementar a demanda de maneira satisfatória. No papel de Construtor, o jogador deve utilizar os blocos disponíveis para solucionar a demanda que foi desenhada pelos arquitetos.

Sendo um Testador, o jogador acompanha a execução da construção, preenchendo um *Checklist* de Conformidade. Ao término da construção por parte dos construtores, o Testador deve indicar quais as diferenças encontradas entre o desenho do arquiteto e a construção dos construtores para que sejam corrigidos em outro turno

(rodada). Para a execução do ConES, uma série de instrumentos foram criados para serem utilizados ao longo da execução de uma partida. Dentre os instrumentos é possível destacar:

- **Ficha de Personagem:** é um documento onde o jogador pode inserir o nome e papel de seu personagem, além disso também existe um espaço para que o jogador distribua pontos ao longo de três atributos (sendo eles *soft skills*, *hard skills* e domínio do escopo) e um espaço para adicionar pontos de experiência sempre que o jogador chegar na fase de Retrospectiva;
- **Cartas de Ação:** são um conjunto de cartas que o jogador pode utilizar durante a fase de Execução do Turno, seja para conseguir algum benefício no turno atual ou para facilitar de alguma forma um turno posterior. As cartas de ação só podem ser utilizadas uma vez e são associadas diretamente ao papel do jogador, que significa que o arquiteto, o construtor e o testador possuem um conjunto distintos de cartas de ação;
- **Cartas de Demanda:** representam uma demanda do Cliente. Ela é formada por um texto que informa o título da demanda e por uma série de critérios de aceitação, que indicam o que o time deve realizar para que a demanda seja construída de maneira satisfatória para o Cliente;
- **Cartas de Evento:** é usada para representar um requisito aleatório que foi alterado pelo Cliente, sendo necessário que o time realize seu ajuste no turno que irá iniciar;
- **Checklist de Conformidade:** é uma ficha que o jogador interpretando o Testador possui acesso. Nela o Testador anota os critérios de aceitação das construções que estão sendo realizadas durante o turno e indica se elas estão de acordo com o que foi solicitado pelo Cliente, de modo que seja fácil de acompanhar o que deve ser feito pelo time;
- **Alterações Solicitadas pelo Cliente:** é uma ficha que o time utilizará durante a fase de Entrega das Demandas, após o time apresentar as construções que foram feitas. O Cliente indicará as alterações desejadas, onde estas alterações devem ser anotadas na ficha de Alterações Solicitadas pelo Cliente para que o time consiga rastrear e realizá-las em turnos posteriores.

A Figura 1 apresenta um exemplo de uma carta de ação de um Construtor, um exemplo de carta de evento e um exemplo de carta de demanda. Os instrumentos utilizados no jogo podem ser consultados no link <https://zenodo.org/record/6727928>.

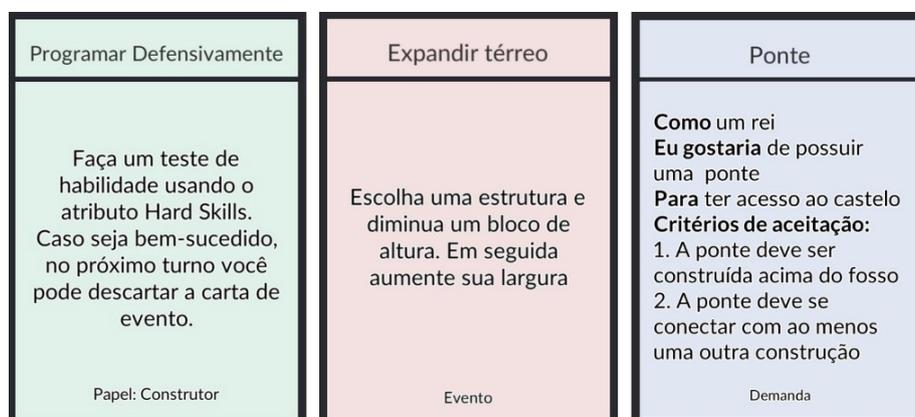


Figura 1. Exemplos de carta de ação, de evento e de demanda

4.2. Mecânicas

O ConES possui um conjunto de mecânicas que ajudam a padronizar e descrever como uma maneira de se jogar com base nas ações que são possíveis de serem realizadas dentro do jogo, sendo:

- **Atributos:** ao preencher a ficha de personagem o jogador deve distribuir seus pontos de atributos entre três categorias: *Soft Skills*, *Hard Skills* e Domínio do Escopo. A distribuição desses pontos influenciará em quão bem o personagem conseguirá executar as cartas de ação do seu papel;
- **Experiência:** é um ponto que o jogador deve atribuir permanentemente a um atributo de referência. Um exemplo é caso o jogador possua 3 pontos em domínio do escopo e decidir adicionar um ponto de experiência nesse atributo, então para todos os efeitos o jogador possui 4 pontos em domínio do escopo;
- **Teste de Habilidade:** consiste em uma jogada de dados feita quando é necessário “testar” um dos atributos do jogador. Quando um jogador utilizar uma carta de ação, ele deve rolar um número de dados equivalente ao número de atributos envolvidos na carta de ação mais um. A ação será bem-sucedida caso o resultado dos dados for menor ou igual a soma dos atributos envolvidos do personagem. Se não for, então o teste falhou;
- **Blocos de Montar:** o time deve trabalhar na construção das demandas do cliente, para isso eles farão uso de blocos de montar que serão dados no início da Execução do Turno.

4.3. Fases

O ConES é um jogo de RPG executado ao longo de cinco fases, onde cada uma das fases possui suas próprias características e objetivos a serem realizados.

A primeira fase é a de **Introdução e Criação de Personagens**. Nessa fase o mestre irá explicar a aventura e qual o procedimento necessário para os jogadores preencherem as fichas de personagens. Após essa fase, os jogadores serão apresentados ao contexto da aventura. A segunda fase tem o nome de **Planejamento do Turno**, onde o Gerente terá acesso às demandas do Cliente e descobrirá quais demandas devem ser entregues no turno que irá iniciar. O mestre deve indicar que estão entrando no Planejamento do Turno e fazer a seleção das cartas de demandas e lê-las em voz alta para que os jogadores entendam.

Execução do turno é o nome da terceira fase do jogo, onde os jogadores devem implementar as demandas que foram estabelecidas. O mestre entrega os blocos de construção para o time e aqui os jogadores podem realizar dois tipos de passos: (1) pode trabalhar efetivamente na construção das demandas do cliente (de acordo com o seu papel); e (2) fazer uso de cartas de ações junto com o teste de habilidade para facilitar o turno atual ou turnos posteriores.

A quarta fase é chamada de **Entrega das Demandas**, onde os jogadores devem apresentar ao Cliente as demandas que foram construídas ao longo do turno para que sejam validadas se foram atendidas de maneira satisfatória ou não. Caso existam correções solicitadas pelo Cliente, elas devem ser anotadas na Ficha de Alterações Solicitadas pelo Cliente e corrigidas nos turnos posteriores. No fim da Entrega das Demandas as peças de bloco que não foram utilizadas devem ser retornadas ao Cliente.

A quinta fase possui o nome de **Retrospectiva**, onde os jogadores adquirem um ponto de experiência devido a um ciclo de desenvolvimento que foi encerrado. Durante

a Retrospectiva os jogadores podem aproveitar para refletir sobre as dificuldades encontradas ao longo dos turnos que já passaram e discutir estratégias de construção para possíveis turnos posteriores. A Figura 2 apresenta um fluxo de como ocorre a execução de cada uma das fases do jogo, desde a Introdução e Criação de Personagens até o fim da partida.

5. Avaliação do Jogo

Para verificar se as mecânicas criadas para o jogo eram válidas, foi necessário realizar uma avaliação utilizando participantes externos e obter o retorno destes para saber se as mecânicas estabelecidas eram funcionais ou deveriam ser alteradas. Foram realizadas duas avaliações para testar se o conjunto de mecânicas desenvolvidas para a execução do jogo é satisfatório. Os passos realizados na avaliação dos protótipos consistiram primariamente em: selecionar um grupo de alunos para jogar o protótipo; obter o *feedback* do grupo de alunos após a aplicação do protótipo. Em cada aplicação do jogo foi feita a gravação do áudio da partida para que fosse possível obter dados adicionais que não foram identificados inicialmente, bem como para avaliar se o tempo estimado para cada fase estava acontecendo da maneira que foi planejada ou não.

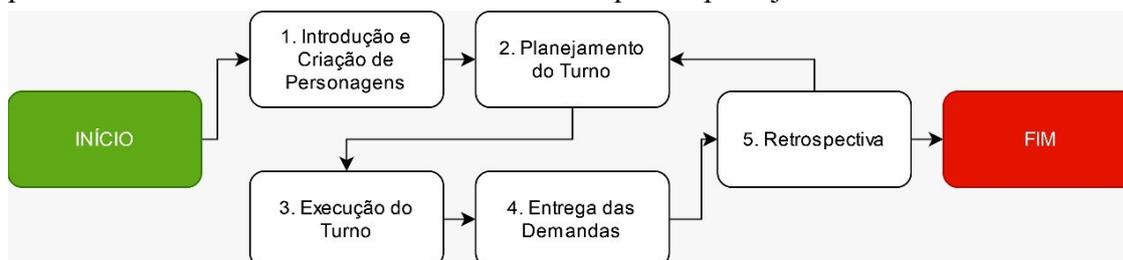


Figura 2. Fluxo de Fases do Jogo

Vale salientar que o primeiro grupo de alunos jogou a **primeira** versão do protótipo. Após a obtenção e análise do *feedback* do primeiro grupo foi realizada uma série de correções no jogo e, conseqüentemente, os grupos posteriores testaram a **segunda** versão do protótipo.

5.1. Relatos da Primeira Avaliação

O primeiro grupo de participantes (Time A) foi formado por acadêmicos que já possuíam certo grau de experiência na área. No momento da execução do jogo os participantes estavam cursando a disciplina de Engenharia de Software II. Os integrantes estavam no sexto semestre do curso e testaram a primeira versão do protótipo.

O Time A foi formado por participantes que já se conheciam e, conseqüentemente, foram bastante comunicativos para resolver problemas. Os participantes possuíam as seguintes características: **Participante A1 (Arquiteto):** participante bastante ativo, ao longo das rodadas foi perguntando ao mestre dúvidas em relação aos requisitos das demandas tanto para ajudar no desenho quanto para explicar aos outros jogadores; **Participante A2 (Construtor):** o jogador apresentou-se bastante atento aos detalhes dos desenhos feitos pelo arquiteto e fez uso de cartas que o permitiram ajudar no desenho; **Participante A3 (Testador):** o jogador participou ativamente levando em conta as limitações dadas pelo seu papel de testador.

Após o *feedback* e análise da gravação da partida, foi possível identificar pontos de ajustes, em que correções eram necessárias. O Quadro 1 exibe os pontos

identificados e quais as soluções utilizadas. Após a aplicação de todas as correções listadas nos Pontos de Solução, foi confeccionada a segunda versão do protótipo do ConES.

Quadro 1. Pontos de Ajustes e Pontos de Soluções da Primeira Avaliação

Pontos de Ajuste	Pontos de Soluções
A ferramenta para desenhar apresentou travamentos (<i>PublishYourDesign</i>)	Por pensar que os travamentos ocorreram devido a máquina do participante, foi escolhido não alterar a ferramenta
O testador está muito parado enquanto os outros jogam	Criar um <i>template</i> de <i>Checklist</i> de Conformidade que o testador deve preencher em tempo real enquanto a construção é feita
Testador pensou que só poderia utilizar as cartas de ação depois que o construtor terminasse de construir	Reescrever o manual de modo que fique explícito que as Cartas de Ações podem ser utilizadas em qualquer ordem dentro da fase de Execução do Turno
O jogador pensou que não existisse um tempo limite nas ações	Por mais que o tempo estimado de cada ação estivesse no manual do jogador, o mestre é quem deve prestar auxílio e indicar para os jogadores como se encontra o tempo disponível em cada fase. Para isso, foi elaborado um manual do mestre contendo essas e outras informações que podem ajudar durante as partidas
As Cartas de Demanda possuem muitos detalhes de construção internos, dificultando a elaboração por parte do time	Reescrever as Cartas de Demanda com foco na parte externa das construções
A separação da fase de Execução do turno em uso de cartas de ação e depois a construção está confusa	Reescrever a fase de Execução do Turno para que o uso de cartas de ação e a construção da demanda possam ocorrer juntas

5.2. Relatos da Segunda Avaliação

A segunda avaliação foi realizada com dois grupos distintos de participantes (Time B e Time C), ambos testaram a segunda versão do protótipo. Além disso, um professor da área de Engenharia de Software participou como observador durante as duas partidas.

No Time B um dos participantes parecia não conhecer os demais (construtor e arquiteto), mas isso não impediu a boa comunicação entre todos os três integrantes. Os participantes estavam no sexto semestre e cursavam a disciplina de Engenharia de Software II e possuíam as seguintes características: **Participante B1 (Arquiteto)**: inicialmente, o jogador estava um pouco quieto, mas ao longo da partida foi contribuindo com o time e ajudando na solução dos problemas propostos; **Participante B2 (Construtor)**: bastante comunicativo com todos os integrantes. No início teve um pouco de problemas nas construções (montava e ao tentar montar outra coisa acabava destruindo a primeira sem querer, etc.), mas no segundo turno já havia superado esse problema; **Participante B3 (Testador)**: participou ativamente em todos os turnos do jogo, fazendo uso do *checklist* de conformidade e avisando em tempo real de inconsistências entre a construção e os requisitos.

Já o Time C foi formado por três integrantes introvertidos e, por causa disso, inicialmente, havia pouca comunicação entre os participantes, mas esse quadro foi sendo alterado por volta do fim do segundo turno em diante. Os participantes estavam no segundo semestre e cursavam a disciplina de Programação II. Possuíam as seguintes

características: **Participante C1 (Arquiteto)**: fazia bastante perguntas em relação aos requisitos para saber se estava no caminho certo no desenho. Também dava alguns conselhos para o construtor para que ele usasse determinadas peças; **Participante C2 (Construtor)**: devido à falta de comunicação inicial do time, o construtor não conseguiu suprir as demandas iniciais de maneira mais satisfatória, mas no decorrer do jogo foi aumentando seu grau de conversa com os outros integrantes para entender melhor o que deveria ser construído; **Participante C3 (Testador)**: inicialmente o testador preenchia o *checklist* de conformidade, no entanto não alertava sobre as inconsistências encontradas na construção (talvez devido a introversão).

O Quadro 2 é responsável por exibir os ajustes e soluções pensadas após obter o *feedback* dos times que jogaram a segunda versão do protótipo.

Quadro 2. Pontos de Ajustes e Pontos de Soluções da Segunda Avaliação

Pontos de Ajustes	Pontos de Soluções
A ferramenta para desenhar apresentou travamentos (<i>PublishYourDesign</i>)	Como já havia acontecido um travamento anterior na avaliação inicial, como relatado no Quadro 1, a solução foi utilizar alguma ferramenta alternativa que não apresenta travamentos (LeoCAD)
A execução do primeiro turno aparenta ser difícil para os jogadores	Duas soluções foram pensadas: (i) começar o jogo com uma construção base; (ii) no primeiro turno, pedir só uma construção
Os jogadores podem ter uma forma de conseguir mais tempo para terminar uma demanda	Criar uma carta que adiciona tempo na fase execução do turno
Adicionar mais um construtor pode gerar uma melhor analogia ao ambiente real de trabalho	Equipe deve ser formada por 4 integrantes, para ter dois construtores mexendo na mesma construção ao mesmo tempo
Algumas cartas de demanda possuem uma escrita confusa ou muito “vazia”	Revisar e reescrever todas as cartas de demanda do jogo
Fase de Introdução e Criação de Personagem está passando do tempo estimado	Elaborar um questionário rápido que os jogadores devem preencher na fase de Introdução e Criação de Personagem, para caso algum jogador esteja em dúvida, o questionário seja o guia para indicar qual papel ele deve interpretar

6. Conclusões

Este trabalho detalhou um protótipo de jogo de RPG que busca ser um instrumento que auxilia no ensino da construção de produtos de software. Para atingir a elaboração do jogo e suas mecânicas, foi realizada a leitura de guias de referência na área da Engenharia de Software e foi feito um estudo de um sistema de RPG que possui regras genéricas.

Duas avaliações foram realizadas para analisar as mecânicas do protótipo, após a primeira avaliação, uma série de correções foram feitas para a execução da segunda avaliação. Os pontos identificados na segunda avaliação foram ajustados nas dinâmicas e mecânicas do ConES, para que ele seja utilizado em um ambiente de sala de aula.

Trabalhos futuros incluem a validação do jogo em um ambiente de ensino real, onde o instrumento será utilizado como apoio no processo de ensino aprendizagem do processo de Engenharia de Software, com foco na construção do produto.

Posteriormente, uma análise dos dados desta validação será realizada para verificar a efetividade do que o jogo se propõe a alcançar como meta.

Referências

- Aragão, R. M. L. (2009) “Role Playing Games No Ensino Do Marketing: Uma Experiência com o Rpg Didático”, *Revista Eletrônica de Educação*, v. 3, p. 162–175.
- Begosso, L. R., Franco, L. H. B., da Cunha, D. S. e Begosso, L. C. (2019) “SimScrumF: a game for supporting the process of teaching Scrum”, In *Proceedings of the 9th International Conference on Information Communication and Management*. ACM.
- Bourque, P. e Fairley, R. E. (2014) “SWEBOK: guide to the software engineering body of knowledge”, IEEE Computer Society.
- Chen, W.-F., Wu, W.-H., Wang, T.-L. E Su, C.-H. Su (2008) “Work in progress - a game-based learning system for software engineering education”, In *2008 38th Annual Frontiers in Education Conference*. IEEE.
- Guerra, A. dos S., Neto, E. M., Furtado, J. C. C. e Oliveira, S. R. B. (2021) “Uma Análise Bibliométrica da Aplicação de Jogos para o Ensino de Engenharia de Software a Partir de uma Revisão Sistemática da Literatura”, *18th CONTECSI, Brasil*.
- Jackson, S. (2004) “Generic Universal RolePlaying System – Lite”, Steve Jackson Games, 4. ed.
- Kemell, K.-K., Risku, J., Evensen, A., Abrahamsson, P., Dahl, A. M., Grytten, L. H., Jedryszek, A., Rostrup, P. e Nguyen-Duc, A. (2018) “Gamifying the Escape from the Engineering Method Prison”, In *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. IEEE.
- Lemos, W., Cunha, J. e Saraiva, J. (2019) “Ensino de Engenharia de Software em um Curso de Sistemas de Informação: Uma Análise dos Problemas e Soluções na Perspectiva de Professores e Alunos”, In *Anais do Workshop sobre Educação em Computação (WEI)*. Sociedade Brasileira de Computação (SBC).
- Leontiev, A. e Frias, R. E. F. (2007) “Psicologia e pedagogia: bases psicológicas da aprendizagem e do desenvolvimento”, São Paulo: Centauro.
- Mettler, T. e Pinto, R. (2015) “Serious Games as a Means for Scientific Knowledge Transfer—A Case From Engineering Management Education”, *IEEE Transactions on Engineering Management*, v. 62, n. 2, p. 256–265.
- Navarro, E. (2006) “SimSE: A Software Engineering Simulation Environment for Software Process Education”, University Of California.
- Ouhbi, S. e Pombo, N. (2020) “Software Engineering Education: Challenges and Perspectives”. In *2020 IEEE Global Engineering Education Conference (EDUCON)*. IEEE.
- Rosa, M. (2010) “Electronic and Online RPG in the Mathematics Education Context”, *Jornal Internacional de Estudos em Educação Matemática*, v.2, n.1, p. 27.
- Schmit, W. L. (2008) “RPG e educação: alguns apontamentos teóricos”, Universidade Estadual de Londrina.
- Silva, L. F. e Jacome, W. C. (2017) “SoftBook: Software Development as an Adventure”, *IEEE Latin America Transactions*, v. 15, n. 6, p. 1205–1211.