



INSTITUTO ENSINAR BRASIL
REDE DE ENSINO DOCTUM

UniDOCTUM

CENTRO UNIVERSITÁRIO DOCTUM DE TEÓFILO OTONI

Artigo apresentado ao curso de Sistemas de Informação do Centro Universitário Doctum de Teófilo Otoni – Unidoctum à disciplina de Trabalho de Conclusão de Curso II como requisito parcial para obtenção de título de Bacharel em Sistemas de Informação.

Linha de Pesquisa II: Informática na Educação

SCRUMBOARD CHALLENGER: UMA ABORDAGEM LÚDICA PARA O ENSINO PRÁTICO DA METODOLOGIA SCRUM

SCRUMBOARD CHALLENGER: A PLAYFUL APPROACH TO PRACTICAL TEACHING OF SCRUM METHODOLOGY

Leonardo Moreira Diniz¹

Wilbert Viana²

Resumo: Este trabalho apresenta o “ScrumBoard Challenge”, um jogo de tabuleiro educacional desenvolvido para proporcionar uma experiência prática e imersiva na Metodologia Scrum. Integrando elementos do Scrum com cartas de habilidades relacionadas à programação, frameworks e hardware, o jogo visa preparar os alunos do curso de Sistemas de Informação para os desafios do desenvolvimento ágil de software. Através de uma abordagem lúdica e interativa, o “ScrumBoard Challenge” busca melhorar a compreensão prática do Scrum, promover o desenvolvimento de habilidades essenciais e oferecer uma perspectiva única sobre os processos ágeis.

Palavras Chave: ScrumBoard, Scrum, Metodologia, Sistema da Informação.

Abstract: This paper introduces the “ScrumBoard Challenge,” an educational board game designed to provide a practical and immersive experience in Scrum methodology. By combining Scrum elements with skill cards related to programming, frameworks, and hardware, the game aims to prepare students in the Information Systems program for the challenges of agile software development. Through a playful and interactive approach, the “ScrumBoard Challenge” seeks to enhance practical understanding of Scrum, foster the development of essential skills, and offer a unique perspective on agile processes.

Key Words: ScrumBoard, Scrum, Methodology, Information Systems.

¹ moreiraleo40@gmail.com

² prof.wilbert.barbosa@doctum.edu.br

Introdução

O ensino eficaz de metodologias ágeis, como o Scrum, é crucial para preparar os estudantes para os desafios dinâmicos da Engenharia de Software. No entanto, a complexidade inerente a esses métodos muitas vezes resulta em dificuldades de compreensão por parte dos alunos, o que impacta diretamente sua capacidade de aplicação prática. Este artigo propõe uma abordagem inovadora para superar esse desafio, utilizando um jogo de tabuleiro projetado especificamente para ensinar e vivenciar o Scrum durante o processo de aprendizagem.

O problema abordado nesta pesquisa reside na dificuldade dos estudantes em assimilar conceitos relacionados a metodologia *Scrum*. A complexidade teórica muitas vezes não se traduz em uma compreensão prática sólida, impactando negativamente a aplicação dessas metodologias em projetos reais de Engenharia de Software. O desafio adicional reside na falta de vivência prática do Scrum durante o ensino, limitando a capacidade dos alunos de internalizar os princípios e práticas ágeis.

O principal objetivo deste trabalho é desenvolver um jogo de tabuleiro educacional que proporcione aos alunos uma experiência prática e imersiva na aplicação do Scrum.

A justificativa para este estudo é fundamentada na necessidade de abordagens inovadoras para tornar o ensino de metodologias ágeis mais acessível e prático.

A metodologia adotada consistirá em duas fases principais. Inicialmente, será realizado um levantamento bibliográfico para embasar teoricamente a construção do jogo de tabuleiro. Em seguida, será desenvolvido o protótipo do jogo, incorporando os princípios do Scrum de maneira fiel.

Este artigo contribuirá para a educação em Engenharia de Software ao propor uma abordagem prática e inovadora para o ensino do Scrum, enfrentando os desafios identificados e promovendo uma compreensão mais profunda e aplicável das metodologias ágeis por parte dos estudantes.

Engenharia de Software

A Engenharia de Software é uma disciplina em constante evolução, moldada pela necessidade de lidar com as crescentes complexidades dos sistemas de software. Este

texto explora a trajetória da Engenharia de Software, destacando seus princípios fundamentais, os desafios enfrentados pela comunidade e as inovações que impulsionam seu desenvolvimento contínuo.

O modelo em cascata, proposto por Royce (1970), foi uma das primeiras tentativas de estruturar o processo de desenvolvimento. No entanto, ao longo dos anos, modelos mais flexíveis e adaptativos, como o modelo incremental de Boehm (1988), ganharam destaque, refletindo a busca por abordagens mais ágeis.

A Engenharia de Software é fundamentada em princípios essenciais para garantir a qualidade e a eficácia do software desenvolvido. O trabalho seminal de Sommerville (2011) destaca a importância da gestão de requisitos, a colaboração entre equipes e a validação contínua. Esses princípios fornecem uma base sólida para o desenvolvimento de software em diferentes contextos.

Apesar dos avanços significativos, a Engenharia de Software enfrenta desafios persistentes. Pressões para desenvolver software mais rapidamente, lidar com sistemas distribuídos e garantir a segurança cibernética são apenas alguns dos desafios destacados por Pressman (2014). A complexidade crescente dos sistemas exige abordagens inovadoras para superar esses obstáculos.

Inovações recentes têm transformado a paisagem da Engenharia de Software. A ascensão de métodos ágeis, como Scrum e Kanban, tem proporcionado maior flexibilidade e capacidade de resposta às mudanças nos requisitos do projeto (Sommerville, 2011). Além disso, técnicas de DevOps, enfatizando a colaboração entre desenvolvimento e operações, tornaram-se cruciais para a entrega contínua e a manutenção eficiente de software.

Metodologia Scrum

De acordo com Schwaber & Sutherland, 2020, o *Scrum* é um *framework* utilizado para organizar o desenvolvimento de software em ciclos curtos e iterativos conhecido como *sprints*. Suas práticas incluem reuniões diárias, planejamento de *sprint*, revisão e retrospectiva do mesmo.

A metodologia ágil Scrum tem se destacado como uma abordagem eficaz para o desenvolvimento de software, proporcionando flexibilidade e adaptabilidade em um ambiente dinâmico. No cerne do Scrum, encontramos uma estrutura organizacional que

valoriza a colaboração, a transparência e a entrega contínua de incrementos de software funcionais. Este texto explora os princípios fundamentais do Scrum, sua aplicação prática e seu impacto positivo nas equipes de desenvolvimento.

O Scrum é baseado em princípios sólidos que fundamentam suas práticas e rituais. A divisão do desenvolvimento em ciclos iterativos, chamados sprints, promove a entrega incremental e possibilita uma resposta rápida às mudanças nos requisitos do projeto. Como ressaltado por Schwaber e Sutherland (2017), criadores do Scrum, essa abordagem visa maximizar a satisfação do cliente ao entregar valor de forma constante.

O Scrum utiliza um framework composto por papéis distintos, como Scrum Master, Product Owner e membros do time de desenvolvimento. Schwaber e Sutherland (2017) destacam a importância desses papéis para garantir a eficácia do Scrum. O Scrum Master atua como um facilitador, removendo obstáculos para o time, enquanto o Product Owner é responsável por definir as prioridades do backlog do produto. Os membros do time de desenvolvimento, por sua vez, colaboram para alcançar os objetivos estabelecidos.

O ciclo de vida do Scrum é dividido em sprints, cada um com uma duração pré-definida. Durante cada sprint, o time de desenvolvimento se compromete a entregar um conjunto de funcionalidades. Isso permite uma abordagem iterativa e incremental, como apontado por Cohn (2010), contribuindo para a entrega rápida e consistente de software de alta qualidade.

A implementação bem-sucedida do Scrum oferece uma série de benefícios. Schwaber e Sutherland (2017) argumentam que a flexibilidade do Scrum permite a adaptação contínua aos requisitos do cliente, promovendo uma maior satisfação. No entanto, é vital reconhecer os desafios, como a necessidade de uma mudança cultural nas organizações, conforme destacado por Ambler (2010).

Uso de jogos no ensino e gamificação

O uso de jogos como ferramenta educacional, conhecido como gamificação, tem ganhado destaque nas últimas décadas como uma estratégia eficaz para engajar e motivar os alunos, proporcionando uma abordagem lúdica e interativa ao processo de aprendizagem. Esta seção explora a literatura sobre o uso de jogos no ensino e destaca a relevância dos jogos de tabuleiro como alternativa valiosa aos jogos virtuais.

A gamificação, definida como a aplicação de elementos e mecânicas de jogos em contextos não lúdicos, tem sido estudada por sua capacidade de estimular a participação e a aprendizagem ativa. Segundo Deterding et al. (2011), a gamificação envolve a incorporação de características como recompensas, desafios e narrativas para tornar atividades educacionais mais atrativas. Isso cria um ambiente propício para a exploração, experimentação e resolução de problemas, fundamentais para o processo de aprendizagem (Anderson & Dron, 2011).

Além da gamificação digital, os jogos de tabuleiro têm se destacado como uma alternativa eficaz e tangível para promover a aprendizagem em diversos contextos. O trabalho de Hunicke et al. (2004) destaca que jogos de tabuleiro proporcionam uma experiência colaborativa, incentivam a tomada de decisões estratégicas e estimulam a resolução de problemas de forma analítica. A natureza física desses jogos promove a interação face a face, contribuindo para o desenvolvimento de habilidades sociais e cognitivas.

Os jogos de tabuleiro oferecem benefícios específicos para o processo de ensino-aprendizagem. Segundo Gómez et al. (2011), esses jogos estimulam a criatividade, o raciocínio lógico e a tomada de decisões, ao mesmo tempo em que proporcionam um ambiente descontraído e inclusivo. A imersão proporcionada pelos jogos de tabuleiro pode criar uma atmosfera propícia para a aplicação prática de conceitos teóricos, promovendo uma compreensão mais profunda e duradoura (Shapiro, 2014).

A escolha entre jogos de tabuleiro e jogos virtuais no contexto educacional depende das metas pedagógicas e das preferências dos educadores. Enquanto os jogos virtuais oferecem simulações interativas e imersivas, os jogos de tabuleiro fornecem uma experiência tangível e analógica. Segundo Egenfeldt-Nielsen (2006), os jogos de tabuleiro destacam-se pela simplicidade, acessibilidade e menor dependência de tecnologia, tornando-os uma opção viável, especialmente em ambientes educacionais com recursos limitados.

O uso de jogos no ensino, seja por meio da gamificação digital ou de jogos de tabuleiro, representa uma abordagem inovadora e eficaz para promover a aprendizagem ativa e envolvente. Os benefícios específicos oferecidos pelos jogos de tabuleiro, como a interação social e a aplicação prática de conceitos, destacam-nos como uma alternativa valiosa no contexto educacional. Educadores podem explorar essas abordagens para enriquecer o processo de ensino e motivar os alunos, criando experiências educacionais memoráveis e impactantes.

Desenvolvimento do Jogo

Com base no conceito da metodologia Scrum abordada anteriormente, o presente trabalho tem como objetivo apresentar o projeto inicial de *boardgame*, para fins educacionais, com intuito de auxiliar no aprendizado e domínio dessa ferramenta.

De acordo com Gee, 2003, Os jogos educacionais são ferramentas eficazes para promover a aprendizagem significativa, proporcionando ambientes simulados para a aplicação prática do conhecimento.

Na prática, os jogos para o ensino de computação necessitam mostrar o impacto esperado de aprendizagem e/ou o engajamento que eles prometem [Caulfield et al., 2011]

Baseando-se na idéia de um jogo educativo para aprendizado do *framework*, esta sendo desenvolvido o ScrumBoard Challenge, que incorpora elementos do *Scrum* em um jogo de tabuleiro, adicionando cartas de habilidades que representam conceitos práticos de programação, banco de dado e convivência.

Esse jogo incorpora elementos do *Scrum*, como o quadro *Kanban*, reuniões diárias e *sprints*, proporcionando uma experiência autêntica de desenvolvimento ágil.

Para proporcionar maior fidelidade à metodologia, o *ScrumBoard Challenge* leva em consideração os princípios pedagógicos ativos e a abordagem centrada no aluno, pensando nisso, para se iniciar o jogo são necessários no mínimo tres jogadores, onde cada jogador pode escolher sua função no jogo usando um dado D6.

Cada jogador recebe um tabuleiro de tamanho A4 para acompanhar as habilidade e um dado D20.

Cada jogador tem 5 proficiências para distribuir entre as áreas de serviço: Back-End, Front-End, Tester, Banco de dados. A distribuição é feita antes do início do jogo e determina as cartas de habilidade que o jogador recebe.

No processo, o professor cria uma história do cliente para engajar os alunos e definir o contexto do serviço solicitado.

Assim, o jogo simula um sprint de 2 semanas (12 dias de trabalho no tabuleiro). O primeiro dia é de planejamento, e o último é para revisão e retrospectiva.

Com o quadro *kanban* no tabuleiro, pode-se visualizar e gerenciar o progresso do projeto, com isso, os jogadores rolam o dado D20 para testar o desempenho na conclusão das tarefas diárias a serem organizadas posteriormente no quadro.

Concluindo 3 tarefas durante 1 semana, os jogadores ganham um emblema.

No caso de conclusão 5 tarefas, quando os jogadores passarem para a próxima semana, ira ficar acumulado as 2 tarefas extras completas, só faltando 1 tarefa para receber o emblema. O mesmo vale no caso do de ficar faltando tarefas durante a semana para completar o emblema.

Os jogadores podem gastar as proficiências para trocar por cartas de habilidade, com limite de 5 proficiências na primeira rodada, e pode ganhar ponto de 1 proficiência a cada Sprint/Reunião concluído com sucesso. Além de precisar tirar um número maior que 10 no D20, caso o jogador tire um 20 no D20 ele ganha 2 proficiência.

Algumas funções que foram descritas anteriormente tem algumas vantagens sobre as outras, de acordo com a hierarquia apresentada, sendo elas:

Product Owner, que pode ajudar o grupo, dando +1 de bônus no dado D20.

Não pode passar mais de 2 dias no mesmo grupo.

Scrum Master pode mover um desenvolvedor para outro grupo 1 vez por rodada, se tirar mais de 12 no D20.

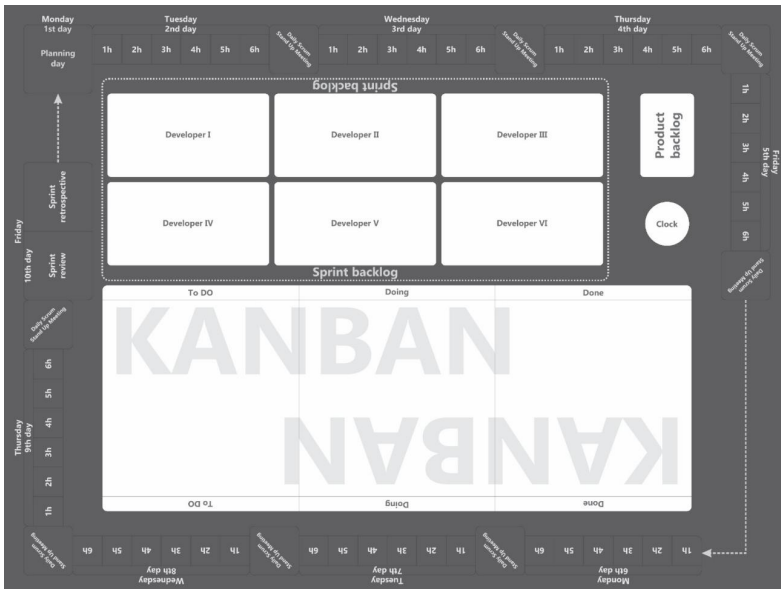
Pode ajudar nas tarefas diárias a cada 3 dias, dando +5 de bônus no dado para todos os jogadores.

O jogo não tem perdedores; o objetivo é concluir todas as tarefas e acumular os emblemas necessários. Para isso, os jogadores precisam juntar 5 emblemas no total, sendo que a dificuldade no jogo depende da tarefa a ser executada e da proficiência do jogador e da área de serviço.

Realizar tarefas fora da área de serviço do jogador aumenta a dificuldade. Tendo que tirar +4 no dado para a conclusão do serviço.

A seguir são apresentados os itens utilizados no jogo.

Para o tabuleiro, de dimensões específicas, representa visualmente o ciclo de um sprint Scrum, com casas que simbolizam os dias de trabalho e ênfase nas práticas diárias, como demonstrado na figura 1.



(Figura 1 - Tabuleiro do jogo)

No jogo, cada jogador assume um papel específico (*Product Owner*, *Scrum Master*, *Membros do Time*) e utiliza dados para introduzir elementos de aleatoriedade, simulando a incerteza do ambiente de desenvolvimento.

Outros componentes do jogo são as cartas de habilidades, que oferecem uma dimensão extra ao jogo, como demonstrado nas figuras a seguir.

Back-End: Cartas com os símbolos de JavaScript, PHP e Python.



(Figura 2 - Cartas Back-end)

Front-End: Cartas com os símbolos de HTML5 e CSS.



(Figura 3 - Cartas Front-end)

Banco de Dados: Cartas com os símbolos de MySQL, Postgres e OracleDB.



(Figura 4 - Cartas banco de dados)

Tester: Carta com o símbolo de tester.



(Figura 5 - Cartas Software testers)

Outro elemento adicionado ao jogo são os emblemas que representam as recompensas por concluir as tarefas e levando a conclusão do jogo.



(Figura 6 - Emblema de placa mãe)



(Figura 7 - Emblema de memória ram)



(Figura 8 - Emblema de placa de video)



(Figura 9 - Emblema de processador)



(Figura 10 - Emblema desenvolvimento de software)

O intuito do jogo sobre a metodologia ágil é proporcionar aos alunos uma abordagem prática da metodologia *Scrum*, para aprimorar sua compreensão e retenção dos conceitos abordados em sala de aula.

Considerações Finais

O desenvolvimento e a implementação do "ScrumBoard Challenge" representam uma iniciativa inovadora e promissora no contexto do ensino de metodologias ágeis, especificamente o Scrum, aos estudantes de Sistemas de Informação. Este jogo de tabuleiro educacional se destaca por oferecer uma abordagem prática e imersiva, buscando superar as dificuldades comuns associadas à compreensão e aplicação do Scrum em ambientes de Engenharia de Software.

Ao longo deste artigo, exploramos a complexidade inerente ao ensino de metodologias ágeis e destacamos a necessidade de abordagens inovadoras para tornar esse aprendizado mais acessível e prático. O "ScrumBoard Challenge" surge como uma resposta a esses desafios, proporcionando aos alunos uma oportunidade única de vivenciar o Scrum de maneira lúdica e interativa.

A pesquisa abordou a dificuldade dos estudantes em assimilar os conceitos do Scrum e identificou a falta de vivência prática durante o processo de ensino como um desafio adicional. O jogo proposto atende a essa lacuna, proporcionando um ambiente simulado que replica elementos-chave do Scrum, como o quadro Kanban, reuniões diárias e sprints. Essa abordagem visa melhorar a compreensão prática do Scrum e preparar os alunos para os desafios do desenvolvimento ágil de software.

O jogo apresenta uma estrutura clara, com papéis definidos (Product Owner, Scrum Master, Membros do Time) e elementos como tabuleiro, cartas de habilidades e emblemas, proporcionando uma experiência autêntica de desenvolvimento ágil. As vantagens e desafios associados a cada função foram cuidadosamente integrados, promovendo o entendimento não apenas do Scrum em si, mas também das dinâmicas de colaboração e responsabilidade dentro de uma equipe ágil. Ao finalizar cada rodada do jogo, os jogadores não enfrentam uma competição tradicional com vencedores ou perdedores, mas sim o desafio de concluir todas as tarefas e acumular os emblemas necessários. Essa abordagem reforça o foco na colaboração e na consecução de objetivos comuns, refletindo os princípios fundamentais do Scrum.

Em síntese, o "ScrumBoard Challenge" se apresenta como uma valiosa contribuição para a educação em Engenharia de Software, oferecendo uma solução inovadora para tornar o ensino do Scrum mais envolvente e prático. Através desse jogo, os estudantes têm a oportunidade não apenas de compreender os conceitos teóricos do Scrum, mas também de aplicá-los em um contexto simulado, preparando-os de maneira mais eficaz para os desafios do desenvolvimento ágil de software.

Referências

- Schwaber, K., & Sutherland, J. (2017). The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game. Scrum.org.*
- Cohn, M. (2010). Succeeding with Agile: Software Development Using Scrum. Addison-Wesley.*
- Ambler, S. W. (2010). Introduction to Disciplined Agile Delivery: A Small Agile Team's Journey from Scrum to Continuous Delivery. IBM Press.*
- Beck, K., Beedle, M., Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). Manifesto for Agile Software Development. Agile Alliance.*
- Caulfield, C., Xia, J., Veal, D., e Maj, S. P. (2011). A systematic survey of games used for software engineering education. In Modern Applied Science, 5(6), 28-43*
- *Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide™*. Scrum.org.*
[Link](<https://www.scrum.org/resources/scrum-guide>)
- *Gee, J. P. (2003). "What Video Games Have to Teach Us About Learning and Literacy." Computers in Entertainment (CIE), 1(1), 20-20.*
- Royce, W. W. (1970). Managing the Development of Large Software Systems. Proceedings of IEEE WESCON.*
- Boehm, B. W. (1988). A Spiral Model of Software Development and Enhancement. ACM SIGSOFT Software Engineering Notes.*
- Sommerville, I. (2011). Software Engineering. Addison-Wesley.*

Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education.

Anderson, T., & Dron, J. (2011). Three generations of distance education pedagogy. *The International Review of Research in Open and Distributed Learning*.

Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: defining "gamification". *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*.

Egenfeldt-Nielsen, S. (2006). Overview of research on the educational use of video games. *Digital Kompetanse*.

Gómez, M. A., Toro, C., Fuentes, J., Riveiro, M., & Piattini, M. (2011). Educational game design for online education. *Computers in Human Behavior*.

Hunicke, R., LeBlanc, M., & Zubek, R. (2004). MDA: A formal approach to game design and game research. *Proceedings of the AAAI Workshop on Challenges in Game AI and Simulation*.

Shapiro, R. B. (2014). *The Gameful World: Approaches, Issues, Applications*. MIT Press.