



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

LUIS HENRIQUE BUZZO FRANCO

**SIMSCRUMF: APOIANDO O ENSINO DE SCRUM ATRAVÉS DE UM
AMBIENTE SIMULADO GAMIFICADO**

**Assis/SP
2021**



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

LUIS HENRIQUE BUZZO FRANCO

**SIMSCRUMF: APOIANDO O ENSINO DE SCRUM ATRAVÉS DE UM
AMBIENTE SIMULADO GAMIFICADO**

Projeto de pesquisa apresentado ao curso de Bacharelado em Ciência da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

**Orientando(a): Luis Henrique Buzzo Franco
Orientador(a): Prof. Dr. Luiz Ricardo Begosso**

**Assis/SP
2021**

FICHA CATALOGRÁFICA

F825s FRANCO, Luis Henrique Buzzo

Simscrumf: apoiando o ensino de scrum através de um ambiente simulado gamificado / Luis Henrique Buzzo Franco. – Assis, 2021.

54p.

Trabalho de conclusão do curso (Ciência da Computação). – Fundação Educacional do Município de Assis-FEMA

Orientador: Dr. Luiz Ricardo Begosso

1.Engenharia de Software 2.Gamificação 3.Scrum

CDD005.12

SIMSCRUMF: APOIANDO O ENSINO DE SCRUM ATRAVÉS DE UM AMBIENTE SIMULADO GAMIFICADO

LUIS HENRIQUE BUZZO FRANCO

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: _____
Prof. Dr. Luiz Ricardo Begosso

Examinador: _____
Prof. Dr. Luiz Carlos Begosso

RESUMO

Compreender corretamente os conceitos de Engenharia de Software é algo extremamente importante para alunos de qualquer curso relacionado a tecnologia da informação, pois trata-se de uma área que abrange todas as etapas do desenvolvimento de softwares, e tais conhecimentos tem sido cada vez mais requisitados pelo mercado de trabalho. A geração atual demonstra grande afinidade com a tecnologia, fator que torna viável o desenvolvimento de aplicações que possuem elementos de gamificação, voltadas ao ambiente acadêmico, com o intuito de promover um aprendizado mais divertido e interativo. O SimScrumF é uma ferramenta gamificada, com suporte a múltiplos idiomas, que almeja ensinar os conceitos básicos de Scrum, permitindo com que o aluno vivencie, de forma simulada, grande parte das etapas do planejamento, gestão e desenvolvimento de um software, regido pela metodologia. O presente projeto tem como objetivo dar continuidade a implementação do SimScrumF, adicionando mais conteúdos a plataforma, como elementos de gamificação e interações com o usuário, objetivando tornar o aprendizado mais divertido.

Palavras-chave: Engenharia de Software, Gamificação, Scrum.

ABSTRACT

Correctly understanding the concepts of Software Engineering is extremely important for students of any course related to information technology, as it is an area that covers all stages of software development, and such knowledge has been increasingly required by the job market. The current generation demonstrates great affinity with technologies, a factor that makes it feasible to develop applications that have gamification elements, aimed at the academic environment, to promote a more fun and interactive learning. SimScrumF is a gamified tool, with support for multiple languages, which aims to teach the basic concepts of Scrum, allowing the student to experience, in a simulated way, a large part of the stages of planning, management and development of software, governed by the methodology. This project aims to continue the implementation of SimScrumF, adding more content to the platform, such as elements of gamification and interactions with the user, aiming to make learning more fun.

Keywords: Software Engineering, Gamification, Scrum.

LISTA DE ILUSTRAÇÕES

Figura 1. Menu lateral da aplicação. Nesse momento, o jogador está controlando as ações da personagem Maria Rocha, a <i>Product Owner</i> da equipe.	26
Figura 2. Perfil do jogador.....	27
Figura 3. Detalhes do produto que está sendo desenvolvido.	28
Figura 4. Ranking de jogadores online.	28
Figura 5. Base de Conhecimento.	29
Figura 6. Cabeçalho da Aplicação.	30
Figura 7. Conversa entre a Product Owner e o Gerente.....	31
Figura 8. Jogador desbloqueando uma conquista.	32
Figura 9. Priorização do <i>Product Backlog</i>	32
Figura 10. Desenvolvimento da Sprint.....	33
Figura 11. Desafio de Associação.	34
Figura 12. Desafio de Ordenação.....	35
Figura 13. Fluxo de etapas das três primeiras fases do SimScrumF.....	36
Figura 14. Imagem original do layout do item “Página Inicial”, do Sprint Backlog.	40
Figura 15. Imagem original do layout do item “Página Sobre”, do Sprint Backlog.	40
Figura 16. Imagem original do layout do item “Página Contato”, do Sprint Backlog.	41
Figura 17. Imagem embaralhada do layout do item “Página Inicial”, do Sprint Backlog, onde o jogador deve organizar novamente ao estado original.	41
Figura 18. Desafio de palavras cruzadas.....	42
Figura 19. Desafio de Digitação.....	43
Figura 20. Tela final da aplicação.	43
Figura 21. Fluxo de etapas implementadas na quarta fase.	44

LISTA DE TABELAS

Tabela 1. Cronograma para a realização do projeto.....	14
Tabela 2. Quantidade de alunos por curso e série que participaram do processo de validação do SimScrumF.....	45
Tabela 3. Perguntas e respostas relacionadas a opinião dos estudantes sobre o SimScrumF.....	47
Tabela 4. Perguntas e respostas relacionadas aos conceitos do Scrum.....	48

SUMÁRIO

1. INTRODUÇÃO	10
1.1. OBJETIVOS	11
1.2. JUSTIFICATIVA	11
1.3. MOTIVAÇÃO.....	12
1.4. PERSPECTIVA DE CONTRIBUIÇÃO.....	12
1.5. METODOLOGIA DE PESQUISA	12
1.6. RECURSOS NECESSÁRIOS	12
1.7. ESTRUTURA DO TRABALHO.....	13
1.8. CRONOGRAMA.....	13
2. REVISÃO BIBLIOGRÁFICA	15
2.1. GAMIFICAÇÃO	15
2.2. SCRUM	16
3. TRABALHOS CORRELATOS	20
3.1. SIMSE	20
3.2. ENSINANDO SCRUM COM MINECRAFT	21
3.3. SCRUMIA.....	22
4. ESTADO ATUAL DO PROJETO	24
4.1. TEMÁTICA	24
4.2. NOME DE PERSONAGENS E FIGURAS	25
4.3. FUNCIONAMENTO DA FERRAMENTA.....	25
5. TECNOLOGIAS UTILIZADAS	37
5.1. JAVA	37
5.2. SPRING.....	37
5.3. TYPESCRIPT	38
5.4. ANGULAR	38
5.5. POSTGRESQL.....	38
6. IMPLEMENTAÇÃO DE NOVOS RECURSOS	39
7. APLICAÇÃO DA FERRAMENTA E ANÁLISE DOS RESULTADOS OBTIDOS	45
8. CONCLUSÃO	49

REFERÊNCIAS..... 51

1. INTRODUÇÃO

A engenharia de software é uma área da computação voltada para a especificação, desenvolvimento e manutenção de softwares, abordando metodologias, tecnologias e base de métodos e ferramentas adotadas na prática, que guiam o desenvolvimento de softwares. Compreender bem tais conceitos é algo essencial para estudantes de cursos superiores relacionados à tecnologia da informação, tendo em vista que o mercado de trabalho está se tornando cada vez mais exigente, necessitando de profissionais com estes conhecimentos para ocupar cargos em várias áreas relacionadas.

Metodologias ágeis são conceitos contemplados pela engenharia de software, que visam aumentar a eficiência dos processos de planejamento, gestão e desenvolvimento de softwares. Martins (2019) ressalta que a cultura ágil tem ganhado cada vez mais espaço em vários outros setores, além dos relacionados a TI, devido a competitividade entre empresas, pois nessa situação há a necessidade de haver mais agilidade na tomada de decisões estratégicas. O autor salienta que o célere avanço de startups colaborou para a adoção de métodos ágeis em grandes companhias, citando como exemplo a Serasa Experian, empresa que possui atualmente cerca de 2500 funcionários no Brasil e passou pelo processo de adaptação à cultura ágil, e agora 100% de sua estrutura de TI atua sob um modelo ágil.

Existem vários métodos ágeis, como o Lean, Smart, *Extreme Programming* (XP), entre outros. O objeto de estudo deste trabalho será o Scrum. Silva et al. (2009) o definem como sendo um método ágil para o gerenciamento de projetos, onde sua principal função é gerenciar projetos de desenvolvimento de software, mas que também pode ser aplicado em outros contextos em que pessoas precisem trabalhar em conjunto, em detrimento de um objetivo comum. Schwaber e Sutherland (2017) ressaltam que o Scrum consiste em times associados a eventos, artefatos, papéis e regras, onde cada um desses componentes tem um propósito específico e essencial para garantir o sucesso na aplicação do Scrum.

O Scrum é ensinado em cursos de graduação relacionados à tecnologia da informação devido a sua relevância no mercado de trabalho. Para contribuir com a fixação dos conceitos aprendidos em sala de aula sobre o assunto, é importante que os alunos passem por uma experiência prática dos princípios abordados pelo método ágil. Para isso, é necessário que os estudantes vivenciem a experiência de atuar no desenvolvimento de um

projeto regido pelo Scrum, porém, fatores como as limitações da sala de aula podem dificultar esse processo.

Devido à grande afinidade da geração atual com a tecnologia, torna-se viável o desenvolvimento de ferramentas gamificadas para apoiar o ensino das mais diversas áreas, como a engenharia de software. Fardo (2013) define a gamificação como sendo a aplicação de características de jogos em ambientes que não necessariamente são jogos, tais como recompensas, estratégias e mecânicas, com a finalidade de promover o engajamento do usuário na atividade que está sendo executada. Ao gamificar um processo de ensino, espera-se que o usuário se sinta mais motivado a estudar, pois os jogos possuem elementos que promovem o engajamento do jogador, de maneira divertida.

1.1. OBJETIVOS

O presente trabalho tem como objetivo dar continuidade ao desenvolvimento da ferramenta SimScrumF (BEGOSSO e FRANCO, 2019) (BEGOSSO et al., 2019a) (BEGOSSO et al., 2019b), uma plataforma gamificada com suporte a múltiplos idiomas, cujo propósito é apoiar o ensino de conceitos básicos do Scrum. Nessa nova versão, pretende-se agregar mais níveis, explicações práticas e elementos de gamificação, com a intenção de torná-la mais eficaz e atrativa aos estudantes e validar sua eficácia como ferramenta de apoio ao ensino de Scrum, através da aplicação da ferramenta para alunos dos cursos de computação.

1.2. JUSTIFICATIVA

Compreender corretamente os conceitos do Scrum é algo importante para os alunos, tendo em vista sua grande utilização no mercado de trabalho. Embora os professores sejam didáticos e possuam bons materiais de estudo, é importante que o estudante passe por uma experiência prática para fixar os conceitos lecionados, atuando no desenvolvimento de um projeto regido pelo Scrum, porém as limitações da sala de aula, tais como a ausência de tempo e recursos, podem dificultar esse processo.

Ao criar um ambiente gamificado para apoiar o ensino da metodologia, é possível simular os conceitos do Scrum. Devido à afinidade da geração atual com jogos, é de se esperar que, ao gamificar um processo de ensino, o aprendizado se torne mais divertido, motivador e eficiente.

1.3. MOTIVAÇÃO

Gamificar um processo de ensino tem demonstrado ser algo promissor. O SimScrumF tem obtido bons resultados quando aplicado em sala de aula, porém é necessário dar continuidade no desenvolvimento da plataforma, para abranger mais conceitos do Scrum e diferentes maneiras de interação com o usuário, almejando torná-lo um aliado no ensino da metodologia para alunos da área de computação.

1.4. PERSPECTIVA DE CONTRIBUIÇÃO

Espera-se que este trabalho possa apoiar o ensino do Scrum em ambientes acadêmicos, permitindo que os alunos tenham a possibilidade de participar de boa parte das etapas do planejamento e desenvolvimento de software regido pela metodologia, através de um ambiente simulado e gamificado.

1.5. METODOLOGIA DE PESQUISA

A metodologia adotada para este trabalho consistirá nas seguintes etapas: (a) Levantamento Bibliográfico: estudo exploratório sobre os conceitos de gamificação e Scrum; (b) Trabalhos Correlatos: pesquisa sobre outros trabalhos que possuem objetivos semelhantes a esse; (c) Situação Atual do Projeto: descrição das funcionalidades que já foram implementadas na versão anterior do projeto; (d) Propostas de Melhorias: levantamento dos recursos e conceitos que precisam ser agregados ao SimScrumF nessa versão; (e) Implementação: desenvolvimento das funcionalidades descritas no item anterior; (f) Aplicação: aplicação da ferramenta e de um questionário para estudantes de computação, em sala de aula; (g) Análise dos Resultados Obtidos: análise estatística do questionário respondido pelos estudantes; (h) Conclusão: considerações finais sobre o trabalho.

1.6. RECURSOS NECESSÁRIOS

Para a realização do trabalho será necessário apenas um computador, que será utilizado para desenvolver os novos recursos do SimScrumF e atuar como servidor, permitindo que os alunos acessem a ferramenta, através de uma rede LAN, que será estabelecida em um

dos laboratórios de informática da instituição. Para dar continuidade ao desenvolvimento, será utilizado a linguagem de programação Java, com o *framework* Spring, e TypeScript, juntamente com o *framework* Angular, bem como suas bibliotecas, que devem estar instalados no dispositivo.

1.7. ESTRUTURA DO TRABALHO

O presente projeto está dividido da seguinte maneira:

- **Capítulo 1 – Introdução:** Contextualização do tema do trabalho, objetivos, justificativa e perspectiva de contribuição.
- **Capítulo 2 – Revisão Bibliográfica:** Revisão bibliográfica sobre os conceitos de gamificação e Scrum.
- **Capítulo 3 – Trabalhos Correlatos** Estudo sobre trabalhos correlatos ao SimScrumF.
- **Capítulo 4 – Estado Atual do Projeto:** Neste capítulo, são apresentados os recursos que já estão implementados no SimScrumF.
- **Capítulo 5 – Tecnologias Utilizadas:** Capítulo que aborda as tecnologias que serão utilizadas para a implementação.
- **Capítulo 6 – Implementação de Novos Recursos:** Capítulo que descreve detalhadamente o processo de implementação dos conteúdos que foram adicionados à ferramenta, com capturas de tela e exemplos.
- **Capítulo 7 – Aplicação da Ferramenta e Análise dos Resultados Obtidos:** Após finalizar a implementação dos novos recursos, o SimScrumF foi aplicado para alunos de computação da instituição, juntamente com um questionário, com o intuito de verificar a eficácia do projeto como ferramenta de apoio ao ensino de Scrum. Este capítulo descreve o processo de aplicação e a análise dos resultados obtidos.
- **Capítulo 8 – Conclusão:** Capítulo que contempla as considerações finais do trabalho.
- **Referências:** Referências bibliográficas utilizadas no trabalho.

1.8. CRONOGRAMA

Para atingir as metas de desenvolvimento da pesquisa, elaborou-se o cronograma representado na Tabela 1.

Atividade Prevista	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago
Capítulo 1: Introdução										
Capítulo 2: Revisão Bibliográfica										
Capítulo 3: Trabalhos Correlatos										
Capítulo 4: Estado Atual do Projeto										
Capítulo 5: Tecnologias Utilizadas										
Capítulo 6: Implementação de Novos Recursos										
Capítulo 7 – Aplicação da Ferramenta e Análise dos Resultados Obtidos										
Capítulo 8 – Conclusão										

Tabela 1. Cronograma para a realização do projeto.

2. REVISÃO BIBLIOGRÁFICA

2.1. GAMIFICAÇÃO

A sociedade tem evoluído de forma muito rápida e tem tornado a tecnologia e os games cada vez mais parte de seu cotidiano, e isso tem despertado o interesse de pesquisadores a ponto de proporem a adição de elementos de jogos no ensino, realidade que tem sido aplicada muitas vezes em ambientes empresariais, pois os games possuem a característica de desenvolver os dois tipos de aprendizagem: como conhecimento e como habilidade (LEFFA, 2014).

Para Goshevski, Valjanoska e Hatziapostolou (2017), a motivação é um fator essencial para ajudar pessoas na realização de suas metas, aliada a isso, a gamificação tem como objetivo promover um maior envolvimento das pessoas em diferentes atividades de suas vidas.

Carvalho (2016) diz que a gamificação trata da aplicação de dinâmicas e mecânicas de jogos em ambientes fora do contexto de jogos, com o propósito de engajar pessoas, melhorar o aprendizado e resolver problemas, motivando comportamentos e ações. O autor salienta que através da aplicação dos conceitos de gamificação, é possível atingir resultados promissores, tais como o estímulo de uma competição saudável, ensino prático e a possibilidade de medição de desempenho dos usuários.

De acordo com Deterding et al. (2011), o pensamento de utilizar elementos de jogos em contexto que não são jogos com o intuito de aumentar a retenção e atividade do usuário ganhou força rapidamente nas áreas de design de interação e marketing digital, após o sucesso de um serviço baseado em localização chamado Foursquare. Os autores enfatizam que a gamificação tem sido aplicada em várias áreas, como produtividade, saúde, educação e finanças, onde os fornecedores de software tem incorporado tais conceitos através de sistemas de recompensa e reputação com pontos, níveis, quadros de classificação e emblemas.

A gamificação pode ser uma grande aliada para a área acadêmica, podendo colaborar de maneira significativa com o envolvimento e aprendizados dos estudantes, desde que seja aplicada corretamente. Para Fardo (2013), é importante que o educador conheça bem os conceitos da gamificação antes de aplicá-los em sala de aula, pois caso sejam aplicados de maneira incorreta ou equivocada, é possível com que problemas existentes na educação

atual sejam reforçados, tais como a maior valorização de notas obtidas do que o aprendizado em si. O autor ressalta que:

Entendemos que a sua utilização nesse contexto requer uma boa dose de compreensão por parte dos professores, e que esse é particularmente um dos grandes desafios em nossa realidade. Porém, a ideia é que a gamificação possa ser vista como mais um caminho em busca das soluções que a educação no século XXI demanda (FARDO, 2013. p. 7).

Este projeto tem a proposta de utilizar a gamificação como forma de engajar os estudantes no aprendizado dos conceitos de Scrum, incorporando elementos como níveis, desafios, rankings, entre outros.

2.2. SCRUM

Schwaber e Sutherland (2017) definem o Scrum como sendo um *framework* que pode ser utilizado para resolver problemas adaptativos complexos, de forma produtiva. Bernardo (2015) ressalta que o Scrum tem o objetivo de entregar resultados rapidamente com menor custo, almejando fornecer serviços e produtos que se adequem as necessidades do cliente. Embora esta metodologia ágil tenha sido inicialmente desenvolvida para a gestão e planejamento de projetos de software, suas premissas podem ser aplicadas em todas as áreas de gestão que necessitem de mais transparência e organização.

Para garantir seu funcionamento, o Scrum incorpora alguns conceitos, como a atribuição de papéis à membros da equipe, eventos e artefatos. A equipe Scrum é composta por três papéis, sendo eles: *Scrum Master*, *Product Owner* e Equipe de Desenvolvimento.

Coutinho (2020) define o *Scrum Master* como sendo o membro da Equipe Scrum que geralmente possui o maior conhecimento sobre a *framework*, e devido a isso, é de sua responsabilidade potencializar o trabalho do resto da equipe, garantindo que todos compreendam e apliquem corretamente os princípios do Scrum. O autor salienta que as principais características desse profissional são as *soft skills*, ou seja, habilidades de comunicação, política e facilitação. Outra importante função desse membro da Equipe Scrum é assegurar-se de que os impedimentos que surgem durante as *Sprints* sejam removidos, garantindo que a Equipe de Desenvolvimento não tenha seu desempenho comprometido.

O *Product Owner* deve ser o intermediário entre o cliente e o restante da equipe Scrum, possuindo total conhecimento sobre o produto que está sendo desenvolvido, e devido a isso, é sua responsabilidade definir, priorizar e manter o *Product Backlog*, de acordo com as necessidades do cliente. Schwaber e Sutherland (2017) ressaltam que o *Product Owner* deve garantir que o *Product Backlog* seja visível, claro e transparente para todos os outros membros do time.

A Equipe de Desenvolvimento é multidisciplinar, formada por profissionais das mais diversas áreas de atuação, responsáveis por desenvolver o produto em questão. No contexto de desenvolvimento de software, a equipe geralmente é composta por programadores, analistas, designers de interface, entre outros. Para Schwaber e Sutherland (2017), a Equipe de Desenvolvimento deve ser estruturada e capacitada para gerenciar seu próprio trabalho, resultando em uma maior sinergia entre seus membros, fator que promove a eficiência do time em geral.

Artefatos são importantes conceitos do Scrum para o desenvolvimento do projeto, pois proveem transparência e facilitam a inspeção do que está sendo feito, sendo eles: *Product Backlog*, *Sprint Backlog* e Incremento.

O *Product Backlog* trata-se de uma lista ordenada de tudo o que é necessário ser adicionado ao produto, detalhando todos os requisitos, funcionalidades, características, requisitos e melhorias, sendo de responsabilidade do *Product Owner* mantê-lo e ordená-lo (SCHWABER E SUTHERLAND, 2017). O *Product Backlog* é incremental, ou seja, é possível que elementos sejam adicionados, removidos ou repriorizados de acordo com o decorrer do desenvolvimento do produto.

Silva et al. (2009) definem o *Sprint Backlog* como sendo uma interpretação de um dos itens do *Product Backlog*, contendo uma lista de tarefas técnicas que devem ser desempenhadas durante a iteração especificada. Schwaber e Sutherland (2017) salientam que sempre que uma nova tarefa é necessária, a Equipe de Desenvolvimento deve adicioná-lo ao *Sprint Backlog*, devendo também atualizá-lo quando uma atribuição for completada, alterando a estimativa de trabalho restante.

Para Duarte (2020), o Incremento é um artefato gerado pela Equipe de Desenvolvimento ao longo da Sprint, que passou por todas as etapas de desenvolvimento e validação necessárias, de tal forma que possa ser considerado como pronto. O autor ressalta que no contexto de desenvolvimento de software, o Incremento é tratado como um software

desenvolvido e em plenas condições técnicas de ser introduzido em um ambiente de produção.

O Scrum também conceitualiza alguns eventos, que são fundamentais para garantir o bom andamento do projeto, com o objetivo de manter os valores implicados pela metodologia, sendo eles: transparência, inspeção e adaptação (ZULIANI, 2015). O autor enfatiza que os eventos são no formato *Time-Boxed*, ou seja, seu tempo de duração máximo é predefinido, sendo eles: *Sprint*, *Sprint Planning*, *Sprint Review*, *Sprint Retrospective* e *Daily Scrum*.

A *Sprint* é considerada o principal evento do Scrum, tratando-se de um ciclo de desenvolvimento, que possui tempo predeterminado, geralmente variando entre duas a quatro semanas (ZULIANI, 2015). O autor evidencia que o objetivo deste evento é transformar um ou mais itens do *Product Backlog* em uma parte de um software pronto para ser utilizado, ou seja, um Incremento, dentro do período definido para a *Sprint* em questão.

A *Sprint Planning* é uma reunião em que será decidido quais itens do *Product Backlog* farão parte da próxima *Sprint*, onde todos os membros da Equipe Scrum participam, ocorrendo uma troca de conhecimentos entre o *Product Owner* e a *Equipe de Desenvolvimento*, porque enquanto um possui total conhecimento sobre os requisitos do produto, os outros possuem o conhecimento técnico necessário para desenvolvê-lo.

Quando uma *Sprint* é finalizada, ocorre a *Sprint Review*, uma reunião cujo objetivo é inspecionar o que foi desenvolvido, onde participam a Equipe Scrum e os clientes, sendo um momento em que todos colaboram para decidir o que deve ser feito a seguir, pois durante a revisão surgem informações que podem ser importantes para o próximo *Sprint Planning* (SCHWABER E SUTHERLAND, 2017).

Após o término da *Sprint Review*, ocorre a última reunião do ciclo de uma *Sprint*, denominada *Sprint Retrospective*, onde todos os membros da Equipe Scrum participam. Seu propósito é proporcionar um momento de reflexão entre o time, onde cada um deve inspecionar a si mesmo e planejar melhorias para serem aplicadas na próxima *Sprint* (SCHWABER E SUTHERLAND, 2017).

O *Daily Scrum* (ou *Daily Meeting*) é uma reunião diária, cujo propósito é permitir que a Equipe de Desenvolvimento realize a adaptação e inspeção sobre o que é necessário para atingir as metas estabelecidas na *Sprint* (MACHADO, 2019). O autor explica que durante a discussão, é importante que cada membro do time compartilhe qual atividade está

desempenhando no momento, e se está precisando de ajuda ou possui alguma dependência que esteja impedindo ou atrapalhando seu progresso.

3. TRABALHOS CORRELATOS

3.1. SIMSE

O SimSE (BAKER et al., 2010) é uma ferramenta computacional gamificada, interativa e com ambiente gráfico, desenvolvida por pesquisadores da Universidade da Califórnia, Irvine, cujo propósito é ensinar os processos de engenharia de software, focado nos conceitos de gerenciamento de projetos. Durante sua utilização, o jogador passa por uma experiência simulada de atuar como o gerente de projetos, coordenando uma equipe de desenvolvedores, com a possibilidade de tomar decisões importantes, como a contratação e demissão de membros da equipe, gerir as atividades dos funcionários, e monitorar seu progresso.

O cenário presente no SimSE exhibe um escritório virtual, com objetos típicos desse tipo de ambiente, como cadeiras, mesas, lixeiras, salas de reunião, entre outros. Sua interface exhibe elementos que permitem que o jogador monitore a situação dos funcionários, informando alguns dados sobre eles, tais como seu nível de produtividade e a tarefa que está desempenhando no momento, exibindo também informações como o nível de satisfação dos clientes e os detalhes do projeto, como o orçamento e o prazo (BAKER et al., 2010). Os autores ressaltam que os funcionários da equipe reportam suas atividades ao jogador através de balões de falas, no formato de *pop-up*, que são exibidas sobre suas cabeças.

BAKER et al. (2010) relatam que a motivação para desenvolver o projeto SimSE foi a lacuna existente nos métodos tradicionais de ensino relacionadas a engenharia de software, onde os alunos são apresentados a vários conceitos teóricos, porém possuem poucas oportunidades de praticá-los em um processo real de desenvolvimento de software. Entretanto, para utilizar a ferramenta é necessário que o estudante possua conhecimento na língua inglesa, e este fator pode dificultar sua aplicação em universidades brasileiras.

3.2. ENSINANDO SCRUM COM MINECRAFT

Minecraft é um jogo que permite com que os jogadores tenham acesso a um mundo aberto, onde é possível fazer construções livremente, utilizando vários tipos de blocos, possuindo diversos modos de jogo (MARTINS, 2015).

Schäfer (2016) conduziu um projeto de ensino, na faculdade Ostbayerische Technische Hochschule Amberg-Weiden, cujo propósito era apresentar o Scrum, de forma prática, a cinquenta e seis estudantes do segundo ano de bacharelado em Ciência da Computação aplicada e Engenharia Elétrica, que ocorreu durante um semestre, com duração semanal de três horas. O professor assumiu o papel de *Product Owner*, definindo que o objetivo da equipe seria construir um modelo 3D do campus da universidade no Minecraft, onde os estudantes, que já possuíam conhecimento prévio do jogo, deveriam criar salas, laboratórios, entre outras construções da instituição.

Após uma introdução geral aos conceitos da metodologia ágil Scrum, foi disponibilizado aos estudantes recursos para que pudessem realizar as reuniões, como adesivos, canetas e pôsteres, bem como o acesso virtual ao livro *The Scrum Guide*, e recursos da plataforma *Moodle Learning Management System*, como chats e fóruns (SCHÄFER, 2016). O autor explica que, inicialmente, a planta do campus foi dividida em oito quadrantes, que ficaram disponíveis em um repositório no Moodle, e os estudantes foram divididos em oito grupos, onde cada um deles elegeu seu *Scrum Master* e ficaram responsáveis por uma parte do quadrante.

Schäfer (2016) explica que o período de criação da construção principal foi constituído em quatro *Sprints*, onde cada uma delas teve duração de uma semana, sendo que em todas elas havia a realização de *Sprint Plannings*, *Daily Meetings*, *Sprint Reviews* e *Sprint Retrospectives*. O autor salienta que, durante as *Sprints*, as equipes produziram pôsteres sobre o *Product Backlog*, impedimentos, entre outros assuntos, que foram armazenados no Moodle.

Após a conclusão do objetivo proposto, foi conduzido uma validação, oral e escrita, com o objetivo de receber um feedback dos estudantes (SCHÄFER, 2016). O autor explica que não foi possível ter uma conclusão precisa dos resultados, devido a algumas opiniões divergentes dos alunos, onde três deles queixaram-se de muita carga de trabalho, dois mencionaram que prefeririam um projeto real de desenvolvimento de software regido pelo Scrum ao invés de uma abordagem com Minecraft, enquanto muitos outros declararam que

a realização do projeto foi viável e divertida. O autor salienta que, oralmente, os estudantes mencionaram explicitamente que a comunicação e distribuição de trabalho funcionaram bem, e que todos os participantes ficaram satisfeitos com o resultado geral.

Agregar a utilização do jogo Minecraft em atividades dinâmicas voltadas para o ensino de Scrum pode ser algo promissor, pois torna o processo de ensino mais divertido e envolvente, porém como o foco do jogo não é o ensino de Scrum, é possível que o aprendizado dos estudantes seja prejudicado.

3.3. SCRUMIA

Borgatto, Savi e Wangenheim (2013) desenvolveram um projeto chamado SCRUMIA, um jogo manual, que utiliza apenas lápis e papel, para ensinar os conceitos da metodologia ágil Scrum. Os autores explicam que o objetivo do jogo é fortalecer a compreensão dos conceitos de Scrum, exercitando a aplicação de seus processos, almejando que os estudantes sejam capazes de recordarem o que foi lecionado, bem como adquirir a capacidade de distinguir os artefatos, papéis e eventos, tornando-os capazes de executar o processo do Scrum, criando seus respectivos artefatos, para gerenciar projetos de software em situações concretas. Os autores enfatizam que o jogo foi planejado para ser utilizado como estratégia complementar para o ensino de gerenciamento ágil em cursos de graduação, e que antes de jogarem, o ideal seria que os alunos fossem expostos previamente aos conceitos básicos do Scrum.

O jogo baseia-se no planejamento e execução de uma *Sprint* hipotética, onde os alunos são divididos em grupos de seis estudantes, de forma que cada um assuma uma função específica, sendo elas: *Scrum Master*, *Product Owner*, Equipe de Desenvolvimento (função ocupada por três alunos) e Auditor (BORGATTO, SAVI E WANGENHEIM, 2013). Os autores salientam que o jogo é estruturado em diversas etapas, que devem ser executadas sequencialmente por todos os grupos, de forma simultânea, com duração total de cerca de sessenta minutos.

Com o intuito de fornecer os contextos necessários para a definição dos requisitos do usuário e tornar o jogo mais atraente, os autores criaram uma narrativa fictícia, e a partir dela, os alunos podem iniciar o jogo, que decorre através de etapas que abordam os conceitos do Scrum, iniciando na formulação do *Product Backlog*, seguindo para a realização da *Sprint Planning*, desenvolvimento intercalado com *Daily Meetings*, *Sprint*

Review e entrega, finalizando com uma conversa que tem como objetivo fazer com que os estudantes reflitam sobre o uso do Scrum para gerenciar projetos de forma geral, comparando-o com outras abordagens tradicionais de gerenciamento de projetos. O desenvolvimento é simbolizado com a criação de aviões, barcos e chapéus de papel, e outras atividades similares, relacionadas a história fictícia criada.

Embora a aplicação em sala de aula de um jogo manual para o ensino de Scrum seja interessante e descomplicada, devido a afinidade da nova geração com a tecnologia, é possível que os estudantes se sintam mais estimulados a jogar um jogo virtual, focado no ensino dos conceitos da metodologia através de uma simulação, permitindo que os alunos passem por uma experiência simulada de grande parte das etapas de planejamento, gestão e desenvolvimento de um projeto de software, sem as limitações da sala de aula.

4. ESTADO ATUAL DO PROJETO

4.1. TEMÁTICA

O SimScrumF é uma ferramenta gamificada que tem o objetivo de apoiar o ensino dos conceitos de Scrum, permitindo que o aluno possa interagir com o processo de desenvolvimento de software numa forma que dificilmente seria possível pelos métodos tradicionais de ensino em sala de aula. Com o objetivo de engajar o aluno, o SimScrumF é dividido em várias fases nas quais o estudante recebe conceitos e é testado através de diversos desafios, tais como associação, ordenação ou palavras-cruzadas.

O SimScrumF possui a temática de um CRPG (computer role-playing game), aplicando parte de suas características-base com o intuito de promover o engajamento do estudante com o aprendizado que lhe é apresentado na aplicação, sendo elas:

História: Durante o decorrer do uso da plataforma, o jogador tem a oportunidade de vivenciar virtualmente uma sequência de ações que permitem que ele passe por boa parte das etapas da gestão e desenvolvimento de software regido pelo Scrum, desde o levantamento de requisitos até a entrega incremental de cada módulo do sistema.

Cenários: Ao longo da história, o estudante passa por cenários relacionados a ação que está acontecendo naquele momento, como a mesa de reuniões da equipe de desenvolvimento e o escritório do cliente.

Personagens: Conforme a necessidade da etapa atual do ciclo do Scrum, o jogador controla as ações dos funcionários de uma equipe Scrum, como o *Product Owner*, *Scrum Master* e Equipe de Desenvolvimento, de forma que suas atribuições e responsabilidades são explicadas durante as ações requisitadas.

Interface: O SimScrumF possui uma interface intuitiva, que permite que o estudante consiga acessar, de maneira simplificada, todos os recursos que a aplicação possui, como seu perfil, progresso individual e o ranking de jogadores online.

4.2. NOME DE PERSONAGENS E FIGURAS

O nome dos personagens que são apresentados durante o decorrer da utilização da plataforma são todos fictícios, não fazendo nenhuma referência direta ou indireta a qualquer pessoa, entidade ou instituição. Todas as imagens e figuras utilizadas no projeto foram retiradas de repositórios gratuitos na internet, sem a necessidade de dar créditos a direitos autorais.

4.3. FUNCIONAMENTO DA FERRAMENTA

Quando o usuário acessa o SimScrumF em seu navegador, lhe é apresentado uma tela de seleção de idiomas, onde é possível selecionar em qual língua a aplicação irá ser exibida, atualmente há suporte para o português do Brasil e inglês.

Após selecionar o idioma, é apresentado uma tela para acessar a ferramenta, onde o aluno deve informar seu nome e o nome de sua equipe, para identificá-lo durante o decorrer do jogo. O nome da equipe é tratado como o nome da empresa fictícia do jogador durante o jogo.

Ao informar os dados e confirmar, o estudante é apresentado ao tutorial da ferramenta, com o propósito de ensiná-lo a utilizar todos os recursos nela disponíveis. A tela inicia-se vazia, com uma caixa de texto dando as boas-vindas ao usuário, e conforme ele confirma a leitura do texto, o conteúdo da interface aparece aos poucos, enfatizando o que está sendo explicado.

O menu lateral da aplicação é composto por alguns itens, conforme ilustra a Figura 1. No topo, é exibido a imagem, nome e cargo do membro da equipe Scrum que o jogador está controlando no momento, bem como algumas opções, sendo elas:



Figura 1. Menu lateral da aplicação. Nesse momento, o jogador está controlando as ações da personagem Maria Rocha, a *Product Owner* da equipe.

Meu Perfil: Ao clicar nesta opção, uma tela contendo todas as informações do jogador são exibidas, como nome, nome da equipe, nível, pontos de experiência, pontuação, progresso no nível atual e conquistas, conforme exibido na Figura 2. A barra de progresso é preenchida conforme o estudante recebe pontos de experiência. O SimScrumF possui o conceito de conquistas, contendo três conquistas para cada membro da Equipe Scrum (Product Owner, Scrum Master, Equipe de Desenvolvimento) e para o próprio jogador, totalizando doze. Caso o aluno queira saber o que deve fazer para liberar determinada conquista, deve passar o mouse no ícone de troféu, que está apagado, pois a conquista ainda não foi liberada, tendo sua cor alterada apenas quando o requisito for atingido.

Para liberar a primeira conquista como *Product Owner*, o jogador deve concluir sua primeira conversa com o cliente, com o intuito de levantar os requisitos da aplicação que será desenvolvida. A segunda conquista requer que o estudante realize a priorização de um *Product Backlog*, e a terceira, e mais difícil, será desbloqueada quando três *Product Backlogs*, de produtos diferentes, forem priorizados.

As conquistas do *Scrum Master* são relacionadas a sua capacidade de resolução de problemas e a realização dos eventos do Scrum. A primeira conquista será liberada quando o jogador resolver um impedimento que está atrapalhando o empenho da Equipe de

Desenvolvimento, a segunda quando ele conduzir duas reuniões de *Sprint Planning* e a terceira quando houver duas reuniões de *Sprint Review*.

Os objetivos para desbloquear as conquistas da Equipe de Desenvolvimento são mais simples. Cada conquista é desbloqueada na medida que os itens do *Product Backlog* são desenvolvidos, uma para cada módulo do sistema, ou seja, todas são liberadas quando o jogador conclui três Sprints.

Sempre que o estudante conclui algum evento ou artefato do Scrum ou adquire uma nova conquista, ele é recompensado com pontos de experiência, que ao atingirem certa quantidade, o jogador ganha mais um nível em seu perfil, e seus pontos de experiência são zerados. Para desbloquear as três conquistas do jogador, ele deve atingir o nível dois, três e quatro, respectivamente.

A pontuação do jogador é incrementada sempre que pontos de experiência são adquiridos, dividindo esse valor por dez, e somando o resultado nos pontos. Ao contrário dos pontos de experiência, que são zerados sempre que o jogador sobe de nível, sua pontuação sempre irá aumentar, e é tida como critério para ordenar os jogadores no *ranking*.

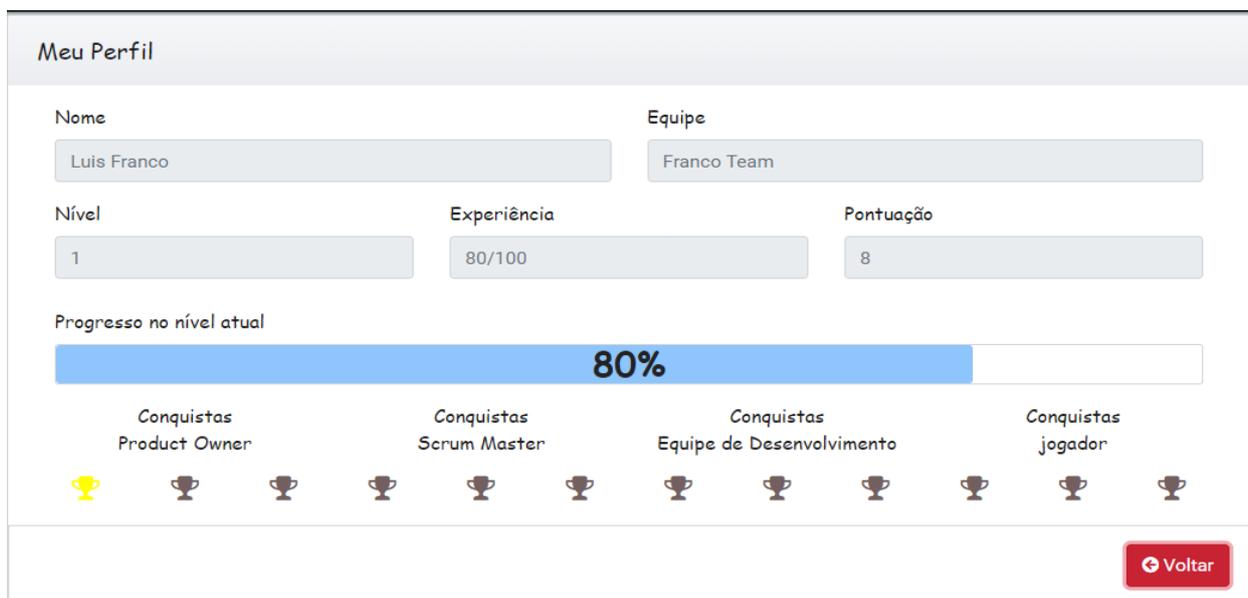


Figura 2. Perfil do jogador.

Sobre: Exibe uma tela com informações sobre a ferramenta, como as tecnologias utilizadas, versão e o nome do autor.

Produto: Tem o propósito de permitir com que o estudante visualize a situação atual do software que está sendo desenvolvido, exibindo os itens do *Product Backlog* que ainda não

foram desenvolvidos, gráfico que ilustra a quantidade de incrementos que foram entregues ao cliente, e a opção de entregar um incremento ao cliente quando a *Sprint* for finalizada, como ilustra a Figura 3.



Figura 3. Detalhes do produto que está sendo desenvolvido.

Amigos Online: Exibe um ranking de todos os jogadores que estão utilizando a ferramenta naquele momento, contendo o nome da equipe, nível, experiência e pontuação de cada estudante, ordenados pela pontuação. Os três primeiros jogadores recebem troféus e possuem um plano de fundo com cores diferentes, estimulando a competitividade entre os usuários, conforme ilustrado na Figura 4.

Amigos Online

Nome	Nível	Experiência	Pontuação
🏆 Franco Team	2	10	11
🏆 Time do Zé	1	80	8
🏆 Time da Maria	1	70	7
João Team	1	50	5

Atualizar Lista Voltar

Figura 4. Ranking de jogadores online.

Iniciar Desenvolvimento: Este botão fica desabilitado até que seja o momento da Equipe de Desenvolvimento iniciar o processo de programação do sistema, ou seja, só estará disponível após a finalização de todas as etapas de levantamento de requisitos e planejamento da *Sprint*. Quando acionado, a ferramenta redireciona o jogador para a respectiva página que contém as ações que devem ser realizadas para prosseguir com o desenvolvimento do sistema.

Bloco de Anotações: Representado por um lápis, o bloco de anotações é uma janela flutuante contendo uma área que o estudante pode realizar anotações durante o decorrer da utilização da ferramenta.

Base de Conhecimento: Representado por uma lâmpada, esta página exhibe ao jogador um painel que contém todos os conceitos relacionados à metodologia que foram apresentados até o momento, e os que ainda não foram aprendidos, representados por pontos de interrogação, que serão revelados conforme o estudante progredir no uso da ferramenta, lendo as respectivas explicações. Ao clicar em um dos conceitos, é exibido um resumo sobre ele no lado direito, como ilustrado na Figura 5.

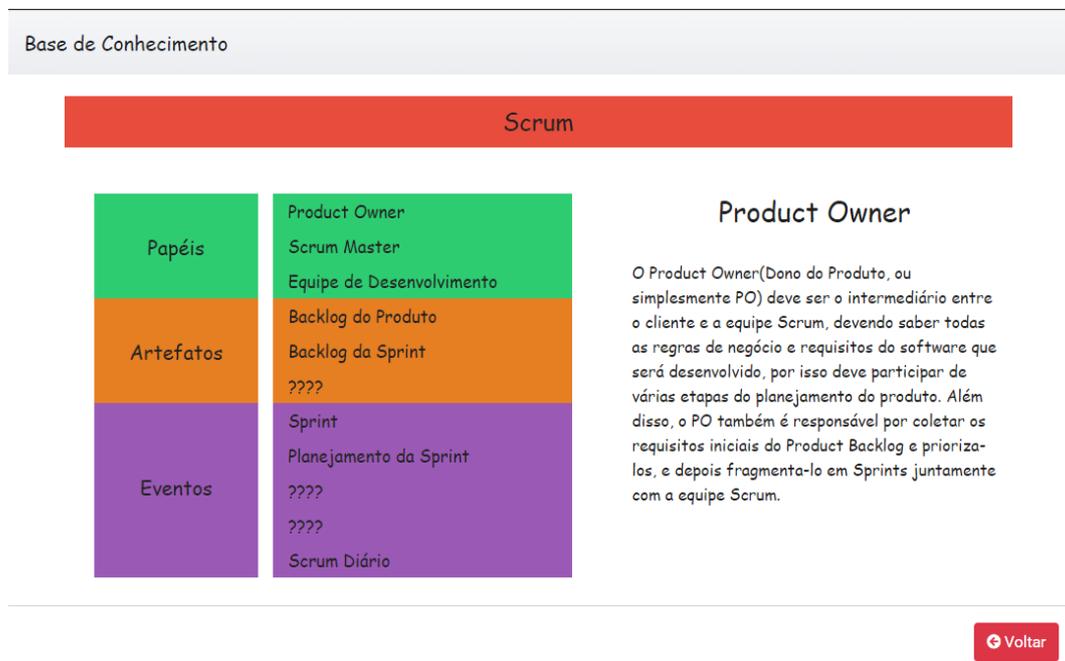


Figura 5. Base de Conhecimento.

No topo da interface do SimScrumF, há um cabeçalho que exhibe um resumo do perfil do jogador, contendo seu nome, nível, pontos de experiência e progresso no nível atual. No

canto direito, são apresentadas quatro imagens, que representam o *Product Owner*, *Scrum Master*, *Equipe de Desenvolvimento* e o próprio jogador, com uma borda colorida e um número sob a imagem. Sempre que uma nova conquista é desbloqueada, esse número é alterado, juntamente com a cor da borda, que se inicia na cor azul, alterando para cor bronze, prata e ouro. Por exemplo, quando o estudante libera sua primeira conquista como *Product Owner*, a imagem referente a este papel tem a cor de sua borda alterada para bronze, exibindo o total de conquistas em sua frente, como mostra a Figura 6.

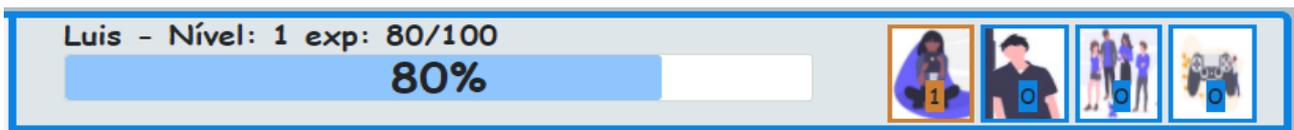


Figura 6. Cabeçalho da Aplicação.

O SimScrumF possui três fases, onde em cada uma delas um novo software é produzido e entregue ao cliente, tendo seu processo de desenvolvimento regido pelo Scrum. Ao concluir o tutorial de como utilizar a ferramenta, o aluno é redirecionado para a primeira fase, que é focada na explicação dos conceitos da metodologia, sendo que durante todo o processo serão exibidas caixas de texto contendo explicações sobre a etapa que está acontecendo no momento, para que o aluno efetue a leitura e possa fixar o referido conceito.

Diversos personagens fictícios foram criados para atuar como atores do processo de desenvolvimento de software. O cliente é nomeado Luiz Silva, Maria Rocha como *Product Owner* e Ricardo Roberto como *Scrum Master*.

A primeira fase inicia-se em uma reunião, no escritório de Luiz Silva, o gerente de uma empresa que está contratando os serviços de desenvolvimento de softwares da equipe do jogador, que nesse momento controla a personagem Maria Rocha, *Product Owner* da equipe. Essa conversa tem o propósito de compreender quais são as necessidades do cliente, e o que ele precisa que seja implementado, com o intuito de formular o *Product Backlog*. Nesse momento, o cenário é alterado para o escritório, e toda a conversa é exibida no inferior da página, como ilustrado na Figura 7.

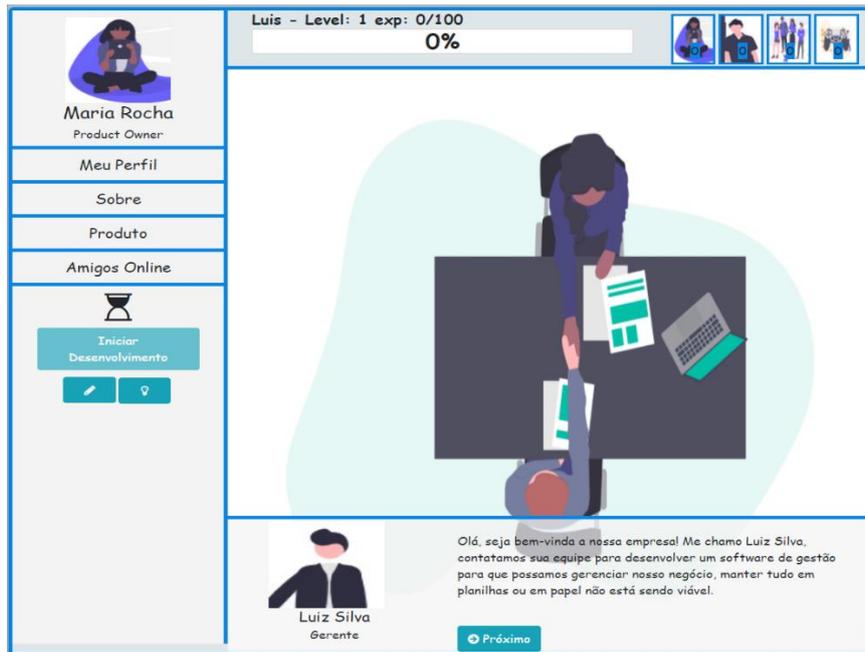


Figura 7. Conversa entre a Product Owner e o Gerente.

Durante a conversa, o estudante jogador será questionado pela primeira vez, com o objetivo de avaliar se ele compreendeu os conceitos apresentados. Dessa forma, uma janela é aberta pedindo que o jogador sugira um sistema que possa agregar funcionalidades úteis para a empresa, de acordo com as necessidades que foram discutidas durante a reunião, sendo exibidas três opções, onde apenas uma delas faz sentido naquele contexto. Caso a opção correta seja selecionada, o jogador será recompensado com cem pontos de experiência, caso contrário, poderá selecionar outra vez, porém receberá menos pontos quando acertar. Quando a reunião é finalizada, o jogador recebe sua primeira conquista como *Product Owner*, conforme mostra a Figura 8.



Figura 8. Jogador desbloqueando uma conquista.

Em seguida, o jogador é redirecionado para a página de priorização do *Product Backlog*, sendo exibida uma explicação sobre sua importância. Durante a conversa entre o gerente e o *Product Owner* na etapa anterior, foi possível identificar quais são os itens que compõem o *Product Backlog*, que são exibidos em formato de lista, e nesse momento, o estudante jogador deverá priorizá-la de acordo com as necessidades do cliente, como ilustra a Figura 9. Caso deseje, o estudante poderá visualizar a conversa novamente para tirar possíveis dúvidas e compreender os requisitos do cliente.

Priorizar itens do Product Backlog

Selecione a ordem de desenvolvimento do produto

Item do Product Backlog	Prioridade
Gerenciamento de Funcionários	1 2 3
Gerenciamento de Estoque	1 2 3
Gerenciamento de Clientes	1 2 3

Product Backlog Priorizado:

- 1º - Sistema de Gerenciar Funcionários
- 2º - Sistema de Gerenciar Estoque
- 3º - Sistema de Gerenciar Clientes

Feito

Lembre-se: Priorize os sistemas de acordo com as necessidades da empresa, pois será a ordem em que as funcionalidades serão desenvolvidas e entregues. Se precisar, clique no botão abaixo para ler as partes mais importantes de sua conversa com o gerente novamente.

Visualizar Conversa

Figura 9. Priorização do *Product Backlog*.

Na próxima etapa, o jogador é introduzido a explicações teóricas sobre a *Sprint*, *Sprint Planning* e *Sprint Backlog*, que após finalizadas, habilitam o botão “Iniciar Desenvolvimento”, no menu. Quando acionado, o jogador assume o papel da Equipe de Desenvolvimento, sendo redirecionado para uma página que contém todas as tarefas que devem ser desenvolvidas na *Sprint*, divididas em quatro colunas, sendo elas: “A Fazer”, “Fazendo”, “Feito” e “Problemas”, representando todas as etapas do desenvolvimento, conforme ilustra a Figura 10. A coluna “A Fazer” contém todos os itens do *Sprint Backlog*, que para serem desenvolvidos, o jogador deve clicar no botão “Iniciar”, ação que irá transferir o item selecionado para a coluna “Fazendo”, e iniciará o carregamento de uma barra de progresso, com efeito de animação de execução durante o passar do tempo, e que quando é totalmente preenchida, transfere o item para a coluna “Feito”.



Figura 10. Desenvolvimento da Sprint.

Durante o desenvolvimento de um dos itens do *Sprint Backlog*, a Equipe de Desenvolvimento reporta um impedimento, e o item que está sendo desenvolvido é migrado para a coluna “Problemas”. Neste ponto, o estudante jogador deverá assumir o papel de *Scrum Master*, e precisará clicar no botão “Solucionar Problema”, que aparece logo abaixo do item, que exibe uma tela, informando que a equipe precisa de mais algumas informações que são importantes para dar continuidade no desenvolvimento. Após isso, uma caixa de texto é exibida, mostrando quais são as ações que um *Scrum Master* deve ter para lidar com essa situação, e então, o item é levado novamente até a coluna “Em Progresso”, até ser finalizado.

Quando todos os itens são finalizados, o botão “Finalizar Sprint” aparece, e quando é acionado, exibe uma explicação sobre a *Sprint Review* e *Sprint Retrospective*, instruindo o

jogador a clicar no item “Produto”, no menu, para entregar o incremento que foi desenvolvido ao cliente, através do botão “Entregar Incremento”, como mostra a Figura 3. Quando acionado, o jogador assume o papel do *Product Owner da equipe*, que está novamente no escritório do gerente, informando que o incremento foi desenvolvido. Caso o jogador tenha priorizado corretamente o *Product Backlog*, o cliente irá dizer que está satisfeito com o que foi desenvolvido, e o estudante receberá um bônus de pontos de experiência. Como o *Product Backlog* é composto de três itens, o processo precisa ser repetido três vezes, até que o desenvolvimento seja finalizado.

Após todos os itens do *Product Backlog* terem sido desenvolvidos e entregues ao cliente, o jogador finaliza a primeira fase do jogo, desbloqueando todos os conceitos da “Base do Conhecimento”. O foco dessa primeira etapa é ensinar os conceitos básicos do Scrum. Antes de iniciar a segunda fase, o jogador é redirecionado a um desafio de associação, onde são exibidas quatro colunas. A primeira delas contém uma lista com todas as responsabilidades dos membros da Equipe Scrum, e as outras três referem-se aos papéis em si, sendo que o jogador deverá arrastar as responsabilidades até a sua coluna correspondente, conforme ilustra a Figura 11. Ao clicar no botão “Finalizar”, ocorre uma verificação, para validar se as atribuições foram associadas aos papéis corretamente, e caso tudo esteja correto, o jogador é recompensado com pontos de experiência, seguindo para o próximo desafio. Se ao menos um item estiver incorreto, é exibido uma tela de diálogo, informando o erro, destacando os que estão organizados corretamente em verde, e oferecendo a possibilidade de tentar novamente, recompensando-o com menos pontos de experiência quando acertar.

Desafio de Associação

Associe as responsabilidades aos seus devidos papéis arrastando-as até seu respectivo local.

Finalizar

✓ 2/7

Responsabilidades	Product Owner	Scrum Master	Equipe de Desenvolvimento
	Deve ser um líder-servidor, portanto não deve ter a autoridade de ordenar que a Equipe de Desenvolvimento faça algo, mas sim indicá-los o melhor caminho a seguir.	Deve ser o intermediário entre o cliente e a equipe Scrum, devendo saber todas as regras de negócio e requisitos do software que será desenvolvido	Sempre que uma Sprint é definida pela equipe Scrum, devem distribuir as atividades entre si, de acordo com suas habilidades e áreas de atuação.
	Sua principal função é garantir que o Scrum seja executado da	Deve impedir que problemas externos atrapalhem a equipe.	

Figura 11. Desafio de Associação.

A próxima etapa consiste em um desafio de ordenação, onde é apresentado, em formato de lista, todas as etapas do processo do ciclo Scrum, desde a definição do *Product Backlog*, até a entrega de um incremento, como ilustra a Figura 12. Estes itens são exibidos em ordem aleatória, sendo que o jogador deve ordená-los e clicar no botão “Finalizar”. Caso a ordem apresentada esteja correta, o estudante será redirecionado para o início da próxima fase, caso contrário, os que estão corretos ficam destacados em verde, e o usuário pode ordená-los novamente, recebendo menos pontos de experiência quando acertar.



Figura 12. Desafio de Ordenação.

As duas próximas fases são parecidas com a primeira, focadas em apoiar a fixação dos conceitos lecionados, mantendo os procedimentos de levantamento de requisitos, priorização e desenvolvimento da primeira fase, ou seja, em cada uma delas, o estudante passa por boa parte das etapas do desenvolvimento de software regido com o Scrum, acrescentando explicações práticas da *Sprint Planning*, *Sprint Review* e *Sprint Retrospective*, onde o jogador assume o papel de *Scrum Master*, conduzindo essas reuniões, além de agregar alguns desafios, auxiliando na fixação dos conceitos. Quando o desafio de ordenação é finalizado, a segunda fase é iniciada, onde o jogador é redirecionado novamente para o escritório do cliente, assumindo o papel de *Product Owner* da equipe, tendo outra reunião para levantar os requisitos da aplicação que será desenvolvida. O mesmo procedimento é aplicado na fase três, após a conclusão da anterior.

Quando a terceira e última fase é concluída, antes de finalizar o jogo, o jogador deve concluir mais dois desafios de associação, que possuem a mesma interface gráfica do que a exibida na Figura 11. No primeiro, são exibidas quatro colunas, a primeira delas, denominada “Itens”, contém uma lista, em ordem aleatória, de todos os papéis, artefatos e eventos do Scrum, e o jogador deve arrastá-los para suas respectivas colunas, sendo elas “Papéis”, “Eventos” e “Artefatos”. Já no último desafio, o jogador deverá mostrar que aprendeu os conceitos relacionados aos artefatos do Scrum, onde são exibidas quatro colunas, a primeira delas, chamada “Itens”, contém uma lista de conceitos, em ordem aleatória, de cada um dos artefatos, e o jogador deve arrastá-los para suas respectivas colunas, sendo elas “*Sprint Backlog*”, “*Product Backlog*” e “Incremento”. A Figura 13 ilustra o fluxo de etapas das três primeiras fases do SimScrumF.

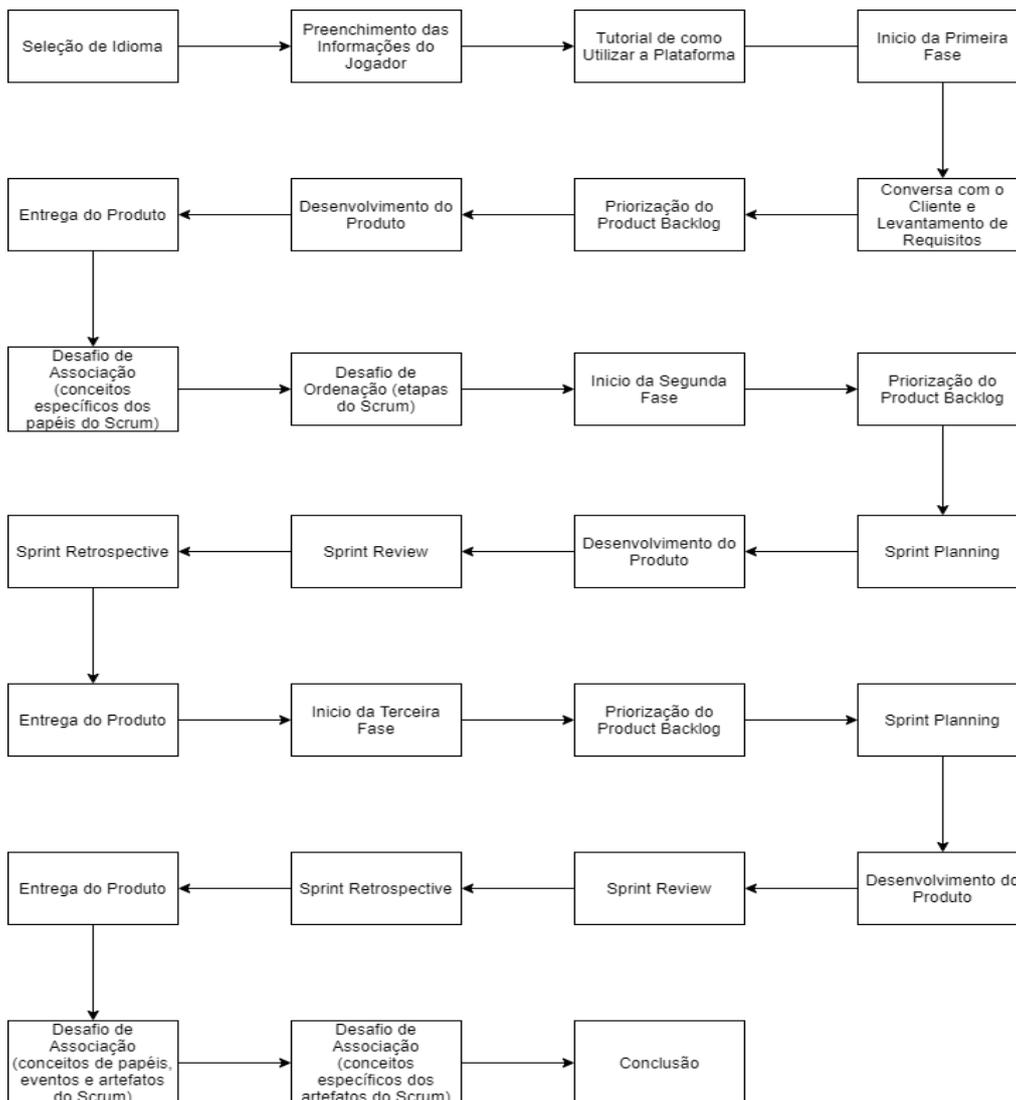


Figura 13. Fluxo de etapas das três primeiras fases do SimScrumF.

5. TECNOLOGIAS UTILIZADAS

5.1. JAVA

A história da linguagem de programação Java iniciou-se no ano de 1991, na empresa Sun Hill Road, composta por uma equipe de engenheiros, liderados por James Gosling, Sun Fellow e Patrick Naughton, filiada a Sun, que atualmente pertence a Oracle (PALMEIRA, 2012). O autor explica que o grupo iniciou um projeto chamado *Project Green*, voltado para a criação de tecnologias modernas de software, para empresas eletrônicas de consumo, com o intuito de permitir a comunicação entre os dispositivos, e para isso, foi desenvolvido o sistema operacional *GreenOS*, responsável por executar a linguagem de programação *Oak*, criada pelo chefe de projeto, James Gosling, que quando foi realizar o procedimento para registrar a linguagem, foi constatado que o nome já havia sido registrado, então, o nome Java foi escolhido.

Java é uma linguagem de programação de alta performance, possui um compilador, orientada a objetos e independente de plataformas, pois pode ser executada em um emulador chamado *Java Virtual Machine* (ou Máquina Virtual Java), dessa maneira, podendo ser utilizada em qualquer sistema operacional que suporte essa máquina virtual (PALMEIRA, 2012). Nesse projeto, o Java, na versão oito, foi utilizado para desenvolver o *back-end* da aplicação, ou seja, o processamento que é executado no servidor, responsável por acessar o banco de dados e processar informações, juntamente com o *framework Spring*.

5.2. SPRING

O Spring Framework é um framework com suporte para Java, utilizando como padrões de projeto os conceitos de inversão de controle e injeção de dependências, constituído por diversos módulos que facilitam o desenvolvimento, tornando o trabalho mais produtivo (CAVALCANTE, 2020). Neste projeto, o Spring Framework foi utilizado para facilitar o desenvolvimento do *back-end*, através da utilização dos módulos Spring Data JPA e Spring Boot WEB, que são responsáveis por gerenciar o acesso ao banco de dados, e permitir a criação de uma API para prover dados ao *front-end* da aplicação, respectivamente.

5.3. TYPESCRIPT

Desenvolvido pela Microsoft, o TypeScript surgiu como sendo um *superset* para a linguagem Javascript, agregando funcionalidades que não são disponíveis nativamente nela, ou que necessitam grande esforço para serem utilizadas, como os conceitos de orientação a objetos e tipagem de dados (DIONISIO, 2016). O autor salienta que a utilização de TypeScript em projetos onde há o intenso uso de Javascript pode ser benéfica, pois ele permite que boas técnicas e práticas de programação sejam utilizadas, tornando o código mais estruturado. Neste projeto, o TypeScript, na versão três, foi utilizado para desenvolver o *front-end* da aplicação, ou seja, a interface do usuário, juntamente com o framework Angular.

5.4. ANGULAR

Criado pela Google, o Angular é um framework voltado para a construção de interface de aplicações, utilizando HTML, CSS, e Javascript, possuindo alguns elementos interessantes que ajudam neste processo, como o conceito de componentização, *templates*, módulos, diretivas, ferramentas de infraestruturas que automatizam tarefas, entre outros recursos (AFONSO, 2018). Este projeto utiliza o Angular, na versão oito, para facilitar o desenvolvimento da interface da ferramenta, voltado para a criação de páginas dinâmicas, tornando o processo de desenvolvimento mais simples e produtivo.

5.5. POSTGRESQL

Souza (2020) define o PostgreSQL como sendo uma ferramenta focada no gerenciamento de bancos de dados relacionados, permitindo a implementação da linguagem SQL em estruturas, onde um de seus pontos principais é a adequação em padrões de conformidade, auxiliando no processo de construção de banco de dados otimizados, armazenando informações de forma segura, lidando bem com altos volumes de solicitações e grandes volumes de processamento. Este projeto utiliza o PostgreSQL, na versão dez, para gerenciar a base de dados da aplicação, armazenando de forma relacional todas as informações utilizadas por ela.

6. IMPLEMENTAÇÃO DE NOVOS RECURSOS

Embora o SimScrumF esteja funcionando corretamente e possua vários recursos, ainda há espaço para a implementação de melhorias. Nessa nova versão, foi implementado uma nova fase, totalizando quatro, que como nas outras, o aluno passa por grande parte dos processos de gerenciamento e desenvolvimento de um software regido pelo Scrum, desde o levantamento de requisitos até a entrega de todos os Incrementos, com mais conteúdos interativos, tornando a experiência do aluno mais divertida e motivadora.

Quando o aluno inicia a quarta fase, ele assume o papel de *Product Owner* da equipe, participando de uma reunião com o gerente, com o intuito de compreender os requisitos do cliente, onde a interface é semelhante a demonstrada na Figura 7. O gerente explica que é necessário desenvolver uma página web para anunciar um de seus produtos. A partir dessa conversa, é possível concluir que é um pedido simples, e que apenas uma *Sprint* será necessária para desenvolver o projeto proposto. O motivo dessa fase ser mais curta que as anteriores, é que ela possui novas interações com o jogador durante o desenvolvimento da *Sprint*, e caso o processo precisasse ser repetido diversas vezes, é possível que o estudante se sinta entediado, pois além de haver três fases anteriores, os novos elementos de gamificação se tornariam repetitivos.

Quando a conversa é finalizada, uma caixa de diálogo é exibida, explicando que, como o *Product Backlog* possui apenas um item, não há a necessidade de realizar sua priorização. Após isso, é exibido uma questão de múltipla escolha, questionando o jogador sobre qual é a próxima etapa a seguir, sendo “Sprint Planning” a resposta correta. Nessa etapa, o estudante é redirecionado para a *Sprint Planning*, assumindo o papel de *Scrum Master* da equipe, liderando a reunião, em que são discutidos quais serão os itens que irão compor o *Sprint Backlog*, sendo eles: uma página inicial, um formulário de contato e uma página que contém informações sobre a empresa.

Após essa etapa ser finalizada, o jogador assume o papel da Equipe de Desenvolvimento, para dar andamento ao processo de implementação do produto, porém, diferente das fases anteriores, onde o jogador clica em um botão e aguarda o processo ser finalizado, nessa nova fase, de acordo com o item do *Sprint Backlog* selecionado, é exibido o layout da página que será desenvolvida, durante vinte segundos, conforme ilustrado pelas Figuras 14, 15 e 16, sendo tarefa do jogador memorizar a imagem. Em seguida, a figura do layout é

embaralhada, semelhante a um jogo de quebra-cabeças, onde o estudante deve organizar as peças corretamente para continuar, conforme ilustra a Figura 17. Esse processo deverá ser executado três vezes, uma para cada item do *Sprint Backlog*, finalizando o desenvolvimento da *Sprint*. Sempre que um item é finalizado, o jogador é direcionado para o *Daily Meeting*, conceito que ainda não havia sido abordado de forma prática nas fases anteriores da ferramenta, havendo uma conversa entre toda a Equipe Scrum sobre as atividades que estão sendo desempenhadas.

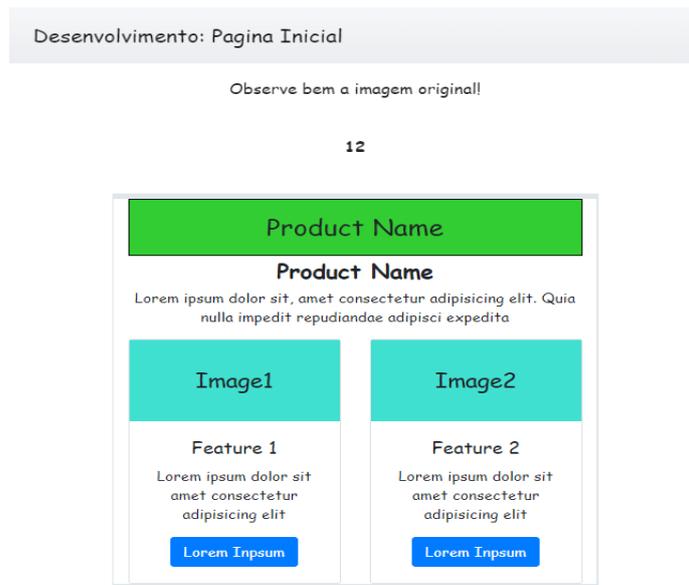


Figura 14. Imagem original do layout do item “Página Inicial”, do Sprint Backlog.

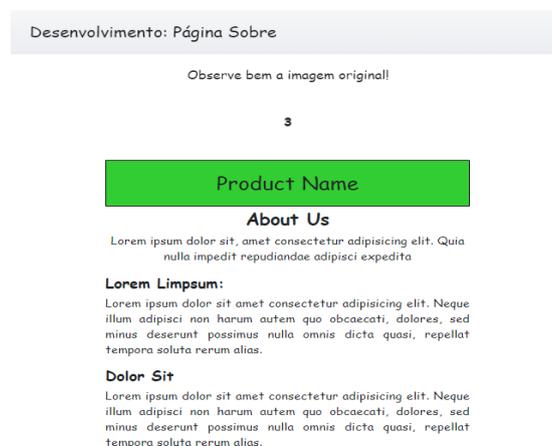


Figura 15. Imagem original do layout do item “Página Sobre”, do Sprint Backlog.

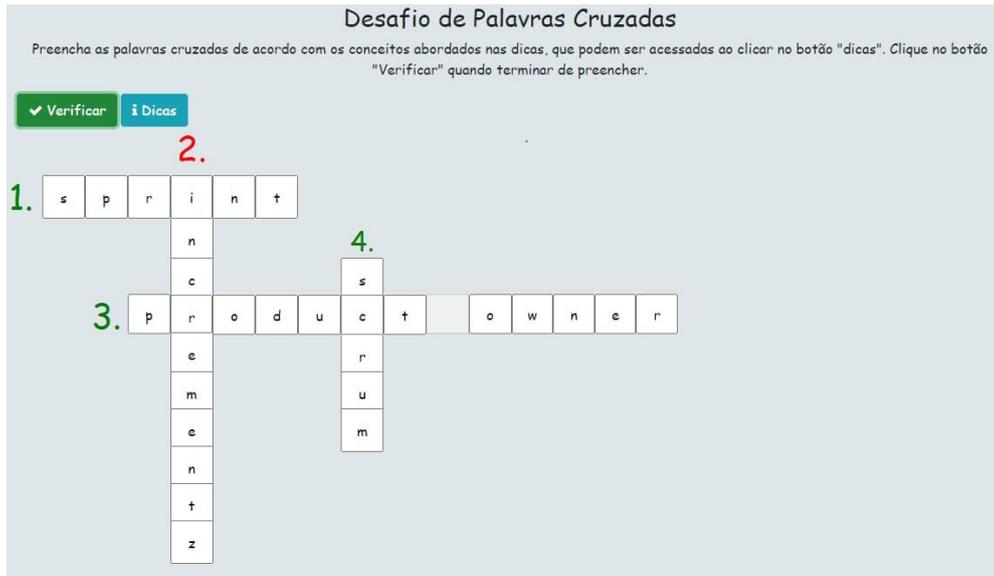


Figura 18. Desafio de palavras cruzadas.

Para acessar as dicas relacionadas a cada palavra, o jogador deve clicar no botão “Dicas”, que exibe uma caixa de diálogo contendo informações sobre cada uma das palavras. Quando o jogador terminar de digitar, o mesmo deve acionar o botão “Verificar”, que valida se tudo foi preenchido corretamente. As palavras que estiverem corretas ficam com seu número indicador verde, enquanto as incorretas ficam vermelhas. Caso o jogador acerte tudo na primeira tentativa, lhe são concedidos cem pontos de experiência, sendo retirados dez pontos para cada tentativa falha, com o valor mínimo de trinta pontos. As respostas corretas são: “Sprint”, “Incremento”, “Product Owner” e “Scrum”, respectivamente.

Ao concluir o desafio de palavras cruzadas, o aluno é redirecionado para um desafio de digitação, com o intuito de promover a fixação das definições aprendidas. Nessa etapa, o nome de um conceito do Scrum é exibido, juntamente com uma caixa de texto, onde o jogador deve preencher digitando o conhecimento que aprendeu relacionado a aquela definição, como mostra a Figura 19. Quando o botão “Revelar Resposta Correta” é acionado, é exibida a definição correta do conceito em questão, para que o estudante efetue a comparação entre ambos. Após efetuar a comparação, o jogador deve acionar o botão “Próximo Conceito” para prosseguir, devendo repetir o processo para cada um dos conceitos da metodologia.

Desafio de Digitação

A proposta deste desafio é que você digite tudo o que aprendeu sobre os conceitos do Scrum na caixa de texto abaixo, e depois, comparar com o conceito correto.

Product Owner

Digite aqui 

Confira a resposta correta:

O Product Owner (Dono do Produto, ou simplesmente PO) deve ser o intermediário entre o cliente e a equipe Scrum, devendo saber todas as regras de negócio e requisitos do software que será desenvolvido, por isso deve participar de várias etapas do planejamento do produto. Além disso, o PO também é responsável por coletar os requisitos iniciais do Product Backlog e prioriza-los, e depois fragmenta-lo em Sprints juntamente com a equipe Scrum.

[Revelar Resposta Correta](#) [Próximo Conceito](#)

Figura 19. Desafio de Digitação.

Depois de concluir o desafio de digitação, o jogador é redirecionado para a página final, como mostra a Figura 20, que contém algumas informações sobre a ferramenta, bem como a opção de jogar novamente. O diagrama representado na Figura 21 ilustra o fluxo de etapas implementadas na quarta fase.



SIMSCRUMF: APOIANDO O ENSINO DE SCRUM ATRAVÉS DE UM AMBIENTE SIMULADO GAMIFICADO PROJETOS DE SOFTWARE

Orientando: Luis Henrique Buzzo Franco
Orientador: Prof. Dr. Luiz Ricardo Begosso

[Jogar Novamente!](#)

Plataforma WEB desenvolvida utilizando Angular e Spring Framework com o intuito de ensinar os conceitos da metodologia ágil SCRUM através de gamificação.

Figura 20. Tela final da aplicação.

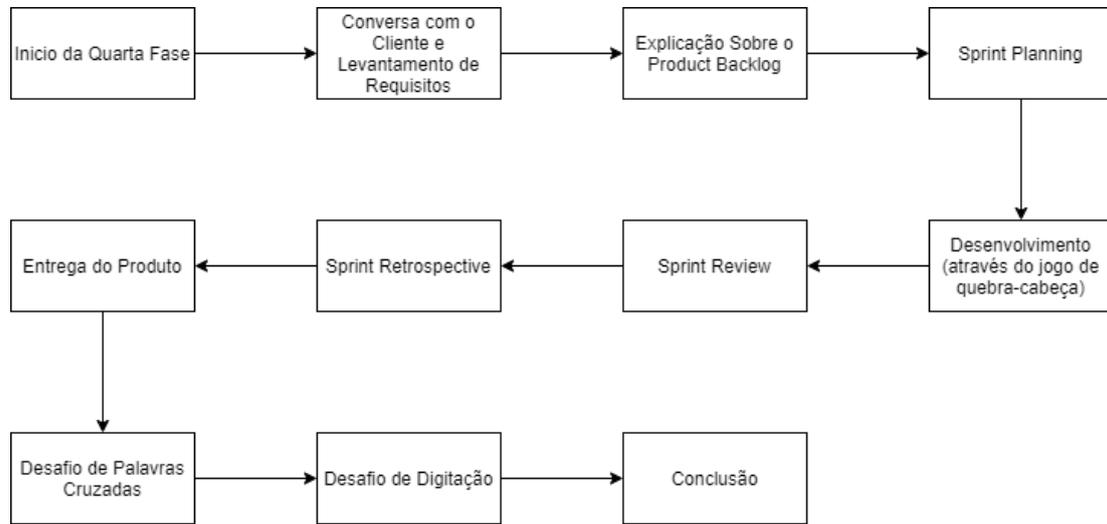


Figura 21. Fluxo de etapas implementadas na quarta fase.

7. APLICAÇÃO DA FERRAMENTA E ANÁLISE DOS RESULTADOS OBTIDOS

Com o propósito de efetuar a validação da eficácia do SimScrumF como ferramenta de apoio ao ensino de Scrum, foi conduzida uma aula introdutória sobre a metodologia, aplicação da ferramenta e de um questionário, para alunos do curso de Análise de Desenvolvimento de Sistemas e Ciência da Computação da FEMA – Fundação Educacional do Município de Assis, que são estagiários do setor CEPEIN – Centro de Pesquisas em Informática, da mesma instituição. Treze estudantes participaram do procedimento, todos do sexo masculino, com idade entre dezoito e vinte e oito anos. A Tabela 2 mostra a quantidade de alunos que cursavam cada um dos cursos, bem como suas séries.

Curso e Série	Quantidade de Estudantes
1º Ano de Análise e Desenvolvimento de Sistemas	4
2º Ano de Análise e Desenvolvimento de Sistemas	3
1º Ano de Ciência da Computação	2
2º Ano de Ciência da Computação	3
3º Ano de Ciência da Computação	1
Total	13

Tabela 2. Quantidade de alunos por curso e série que participaram do processo de validação do SimScrumF.

De acordo com informações providas pela instituição, neste momento, ao todo, 94 alunos estão matriculados no curso de Análise e Desenvolvimento de Sistemas, bem como 82 estudantes no curso de Ciência da Computação. A quantidade de alunos que participaram desse processo de avaliação do SimScrumF representa aproximadamente 7% do total de alunos devidamente matriculados nos cursos relacionados a informática oferecidos pela FEMA. Devido as dificuldades impostas pela pandemia do vírus COVID-19, não foi possível incluir mais alunos nesta etapa de avaliação.

Todo o procedimento aconteceu de forma online e remota, através de uma chamada de voz em grupo, utilizando a plataforma Discord. Após todos os participantes ingressarem na chamada, a aula sobre Scrum foi iniciada, com o propósito de proporcionar um conhecimento teórico para os alunos, simulando uma aula tradicional, com duração de 50 minutos.

Quando a aula foi finalizada, os estudantes acessaram o SimScrumF através de seus navegadores, e fizeram a utilização da ferramenta, com o intuito de participar de uma experiência prática e simulada dos processos do Scrum. O tempo médio que os alunos levaram para finalizar todas as etapas propostas pelo SimScrumF foi de uma hora e vinte e seis minutos.

Assim que todos terminaram de utilizar a ferramenta, os estudantes responderam um questionário aplicado através da plataforma Google Forms, de forma anônima, composto por quatorze questões, onde as cinco primeiras são relacionadas a opinião do jogador sobre a ferramenta, com respostas de múltipla escolha, sendo as possíveis respostas: “Discordo Totalmente”, “Discordo Parcialmente” (respostas negativas), “Não Concordo nem Discordo” (resposta neutra), “Concordo Parcialmente” e “Concordo Totalmente” (respostas positivas). As perguntas e respostas estão tabuladas na Tabela 3.

Pergunta	Respostas Positivas	Respostas Neutras	Respostas Negativas
1. As explicações sobre os conceitos do Scrum durante o decorrer da utilização da aplicação estão claras e objetivas?	13 (100%)	0 (0%)	0 (0%)
2. A interface da aplicação está intuitiva?	12 (92,3%)	1 (7.7%)	0 (0%)
3. Você acredita que a aplicação tem potencial para auxiliar o aprendizado dos conceitos da metodologia Ágil Scrum?	13 (100%)	0 (0%)	0 (0%)

4. Você acredita que a aplicação pode complementar o aprendizado dos conceitos da metodologia Ágil Scrum quando aplicada aos alunos após uma aula sobre o tema em sala de aula, de forma complementar?	13 (100%)	0 (0%)	0 (0%)
5. Os elementos de gamificação presentes na aplicação colaboram para o melhor engajamento do usuário e para a compreensão da metodologia Ágil Scrum?	13 (100%)	0 (0%)	0 (0%)

Tabela 3. Perguntas e respostas relacionadas a opinião dos estudantes sobre o SimScrumF.

Analisando a Tabela 3, é possível concluir que excelentes resultados foram obtidos, mostrando que os alunos acharam a interface do SimScrumF intuitiva, gostaram dos elementos de gamificação presentes e acreditam em seu potencial como ferramenta para apoiar o ensino de Scrum, pois todas as questões tiveram o percentual de respostas positivas superiores a noventa e dois por cento.

As próximas oito questões do questionário foram planejadas para verificar se os estudantes conseguiram compreender corretamente os conceitos do Scrum. Cada pergunta possui cinco alternativas, onde apenas uma é a correta. A relação entre perguntas e acertos está tabulada na Tabela 4.

Pergunta	Quantidade de Acertos	Quantidade de Erros
6. Quais são os papéis (cargos) fundamentais que compõem o Scrum?	13 (100%)	0 (0%)
7. Assinale a alternativa que contém a ordem correta dos acontecimentos	13 (100%)	0 (0%)
8. O que é o artefato Product Backlog?	12 (92,3%)	1 (7,7%)

9. Qual é a função da Equipe de Desenvolvimento?	12 (92,3%)	1 (7,7%)
10. Qual é a função do Scrum Master?	10 (76,9%)	3 (23,1%)
11. Qual é a função do Product Owner?	13 (100%)	0 (0%)
12. A metodologia Ágil Scrum utiliza vários conceitos, trazendo reuniões, cargos e afins para auxiliar a gestão do produto, tais como Product Owner, Sprint Planning, Incremento, entre outros. Esses conceitos são distribuídos em 3 categorias, sendo elas:	11 (84,6%)	2 (15,4%)
13. O que é o Daily Meeting?	13 (100%)	0 (0%)

Tabela 4. Perguntas e respostas relacionadas aos conceitos do Scrum.

Após analisar os dados presentes na Tabela 4, é possível concluir que foram obtidos bons resultados, pois todas as questões obtiveram uma taxa de acertos superior a 75%, bem como uma média de acertos de aproximadamente 93%, mostrando que o SimScrumF tem potencial de auxiliar o processo de aprendizado da metodologia ágil Scrum quando utilizado em conjunto com uma aula tradicional.

A última questão teve o intuito de coletar a opinião dos alunos sobre a ferramenta e sugestões para uma próxima versão. No geral, os estudantes elogiaram a aplicação, enfatizando sua capacidade de ensinar os conceitos do Scrum, e sugeriram a adição de novas palavras no desafio de palavras cruzadas, bem como algumas melhorias no funcionamento do desafio de quebra-cabeças.

8. CONCLUSÃO

O presente trabalho teve como objetivo dar continuidade ao desenvolvimento da plataforma SimScrumF, uma aplicação feita com o intuito de apoiar o ensino da metodologia ágil Scrum em sala de aula, adicionando novos conteúdos interativos para a ferramenta, almejando tornar o processo de aprendizado mais divertido e eficiente.

Com o propósito de fornecer embasamento teórico para as novas implementações, foi realizada uma revisão bibliográfica sobre os conceitos de gamificação e Scrum. Em sequência, como o SimScrumF já possui uma versão anterior, foi feita uma descrição completa da plataforma, explicando sua temática e funcionamento de seus recursos que já haviam sido implementados em trabalhos anteriores. Após isso, foi descrito as tecnologias utilizadas para o desenvolvimento do SimScrumF, expondo seus propósitos e suas aplicabilidades na ferramenta.

Foi desenvolvida uma nova fase, totalizando quatro, com o propósito de proporcionar aos alunos uma oportunidade de ter uma experiência prática, de forma simulada, de parte dos processos de desenvolvimento de software de uma equipe regida pelo Scrum. Nessa fase, foi adicionado uma explicação prática sobre o *Daily Meeting*, conteúdo abordado apenas de forma teórica na versão anterior, bem como a adição de novos elementos de gamificação que tornam a utilização do SimScrumF mais divertida, como o desafio de quebra-cabeças e palavras cruzadas, bem como elementos que promovem a fixação dos conceitos da metodologia, como o desafio de digitação.

Almejando conduzir uma avaliação do SimScrumF como ferramenta de apoio ao ensino da metodologia ágil Scrum, foi dirigida uma aula teórica sobre o assunto para treze alunos de cursos de informática da FEMA, seguido pela aplicação do SimScrumF e finalizando com um questionário, que teve o objetivo de coletar a opinião dos estudantes sobre a ferramenta, e avaliar o conhecimento adquirido.

Após analisar os dados obtidos através do questionário, devido ao alto percentual de acertos obtidos e respostas positivas, foi possível concluir que o SimScrumF pode ser eficiente no processo de apoiar o ensino de Scrum quando utilizado em conjunto a uma aula tradicional, e que os elementos de gamificação presentes na ferramenta, bem como suas explicações, estão satisfatórias.

Como trabalho futuro, no intuito de dar andamento ao projeto, pretende-se realizar novamente a aplicação do processo de avaliação do SimScrumF, envolvendo mais estudantes da instituição, com o propósito de efetuar uma análise estatística mais detalhada e precisa, bem como incrementar o capítulo de trabalhos correlatos, adicionando mais referências bibliográficas e descrevendo outros projetos com objetivos semelhantes ao SimScrumF. É necessário analisar a viabilidade da criação de uma página web que hospede o projeto e ensine os usuários a acessá-lo de forma online.

REFERÊNCIAS

- AFONSO, Alexandre. **O que é Angular?** Algaworks. 2018. Disponível em: <<https://blog.algaworks.com/o-que-e-angular/>>. Acesso em: 8 mar. 2021.
- BAKER, Alex; BEVERLY, Chan; FOG, Terry Mingstun; HOEK, André van der; LEE, Calvin; LEE, Ethan; NAVARRO, Emily; ZHU, Barbara. **SimSE: An Educational, Game-Based Software Engineering Simulation Environment**. Universidade da Califórnia. Disponível em <<https://www.ics.uci.edu/~emilyo/SimSE/details.html>>. 2010. Acesso em: 15 fev. 2020.
- BEGOSSO, Luiz Carlos; BEGOSSO, Luiz Ricardo; CUNHA, Douglas Sanches Da; FRANCO, Luis Henrique Buzzo. The Use of Gamification to Support the Process of Teaching Scrum. In: **24th Annual ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE'19)**, 2019a. Aberdeen, Escócia. p 297-297.
- BEGOSSO, Luiz Carlos; BEGOSSO, Luiz Ricardo; CUNHA, Douglas Sanches Da; FRANCO, Luis Henrique Buzzo. SimScrumF: a game for supporting the process of teaching Scrum. In: **ICICM 2019: Proceedings of the 9th International Conference on Information Communication and Management**, 2019b. Praga, República Tcheca. p 27-31.
- BEGOSSO, Luiz Ricardo; FRANCO, Luis Henrique Buzzo. Implementação De Um Game Para o Gerenciamento De Projetos De Software. **Intelecto**, v.2, n.2, dezembro, 2019, p.6-19.
- BERNARDO, Kleber. **Como funciona o Scrum?**. Cultura Ágil. 2015. Disponível em: <<https://www.culturaagil.com.br/como-funciona-o-scrum>>. Acesso em: 16 fev. 2021.
- CARVALHO, Rafael. **O que é gamificação e como ela funciona?**. Edools. 2016. Disponível em: <<https://www.edools.com/o-que-e-gamificacao/>>. Acesso em: 10 dez. 2020.
- CAVALCANTE, Pablo Henrique Aguiar. **Spring Framework: o que é, seus módulos e exemplos!**. Geekhunter. 2020. Disponível em: <<https://blog.geekhunter.com.br/spring-framework/>>. Acesso em: 10 mar. 2021.
- COUTINHO, Thiago. **Quem é o Scrum Master e o que faz esse profissional?**. Voitto. 2020. Disponível em: <<https://www.voitto.com.br/blog/artigo/scrum-master>>. Acesso em: 16 fev. 2021.

DETERDING, Sebastian; DIXON, Dan; KHALED, Rilla; NACKE, Lennart. From game design elements to gamefulness: defining "gamification". MindTrek '11. In: **Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments**. 2011. p. 9-15.

DIONISIO, Edson José. **Introdução ao TypeScript**. Devmedia. 2016. Disponível em: < <https://www.devmedia.com.br/introducao-ao-typescript/36729>>. Acesso em: 8 mar. 2021.

DUARTE, Luiz. **A Definição de Pronto e o Incremento**. Escritório de Projetos. 2020. Disponível em: < <https://escritoriodeprojetos.com.br/definicao-de-pronto-e-incremento>>. Acesso em: 16 fev. 2021.

FARDO, Marcelo Luis. A Gamificação Aplicada em Ambientes de Aprendizagem. **Renote**, v.9, n.1, julho, 2013, p.1-9.

GOSHEVSKI, Dimitar; VELJANOSKA, Joana; HATZIAPOSTOLOU, Thanos. A Review of Gamification Platforms for Higher Education. 2017. In: 8TH BALKAN CONFERENCE IN INFORMATICS, 8, 2017, Skopje, Macedonia. **BCI '17: Proceedings of the 8th Balkan Conference in Informatics**, 8, setembro, 2017, p. 1-6.

LEFFA, V.L. Gamificação Adaptativa Para o Ensino de Línguas. In: CONGRESSO IBEROAMERICANO DE CIÊNCIA, TECNOLOGIA, INOVAÇÃO E EDUCAÇÃO. 2014, Buenos Aires, Argentina. **Anais do Congresso Iberoamericano de Ciência, Tecnologia, Inovação e Educação**, novembro, 2014, p.1-14.

MACHADO, Filipe. **O que é a Daily Scrum?**. Agile School. 2019. Disponível em: < <https://agileschool.com.br/o-que-e-a-daily-scrum/>>. Acesso em: 17 fev. 2020.

MARTINS, Danylo. **Métodos ágeis avançam nas empresas**. O Valor. 2019. Disponível em: < <https://valor.globo.com/carreira/noticia/2019/11/26/metodos-ageis-avancam-nas-empresas.ghtml>>. Acesso em: 10 out. 2020.

MARTINS, Pedro Henrique. **O que é Minecraft? Conheça um dos jogos mais populares do mundo**. Blasting News. 2015. Disponível em: < <https://br.blastingnews.com/tecnologia/2015/10/o-que-e-minecraft-conheca-um-dos-jogos-mais-populares-do-mundo-00585487.html> >. Acesso em: 10 fev. 2020.

PALMEIRA, Thiago Vinicius Varallo. **Java: história e principais conceitos**. Devmedia. 2012. Disponível em: < <https://www.devmedia.com.br/java-historia-e-principais-conceitos/25178>>. Acesso em: 10 mar. 2021.

SCHWABER, Ken; SUTHERLAND, Jeff. **The Scrum Guide™ The Definitive Guide to Scrum: The Rules of the Game**. Scrum Guides. 2017. Disponível em: <<https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>>. Acesso em: 10 out. 2020.

SCHÄFER, Ulrich. Teaching Scrum with Minecraft. In: **Proc. of European Conference on Software Engineering Education (ECSEE)**. Seeon, Alemanha. 2016, p. 191-195.

SOUZA, Ivan De. **PostgreSQL: saiba o que é, para que serve e como instalar**. Rockcontent. 2020. Disponível em: <<https://rockcontent.com/br/blog/postgresql/>>. Acesso em: 8 mar. 2021.

SILVA, Diego R. M. da; NETO, Gert U. Müller; CLEMENTE, Thárcylla R. N.; SILVA, Thiago R. M. da. **A utilização do Scrum para desenvolvimento de projetos organizacionais: Aplicação na gerência de uma Empresa Júnior**. Faculdade de Ciência e Tecnologia de Caruaru (FACITEC) - Universidade de Pernambuco – UPE. Disponível em: <<https://www.cin.ufpe.br/~gumn/files/4o-ebdqpg.pdf>>. 2009. Acesso em: 12 out. 2020.

ZULIANI, Emerson. **Eventos do Scrum - A Sprint**. 2015. Disponível em: <<https://emersonzuliani.com.br/eventos-do-scrum-a-sprint/>>. Acesso em: 17 fev. 2020.