

# ANÁLISE DO USO DE GAMIFICAÇÃO NA ENGENHARIA DE SOFTWARE

<sup>1</sup>Silvio Marcelino de OLIVEIRA

<sup>2</sup>Luiz Ricardo BEGOSSO

[smarcelino\\_1370@hotmail.com](mailto:smarcelino_1370@hotmail.com)

[begosso@femanet.com.br](mailto:begosso@femanet.com.br)

**RESUMO:** O aprendizado dos conceitos de engenharia de *software* é de suma importância, uma vez que o mercado de trabalho está cada vez mais exigente e necessitando de profissionais que possuam tais habilidades técnicas. O ensino prático desses conceitos se mostra a melhor escolha, uma vez que os estudantes só terem a devida experiência no momento em que estiverem na indústria de *Software* e muitas vezes, até lá, esses conceitos não se fixaram de forma satisfatória. O conceito de ‘gamificação’ tem trazido grandes vantagens no ensino em geral, podendo ser aplicado no ambiente educacional ou profissional. Esse método se destaca por engajar quem a utiliza aprendendo de forma amigável e divertida. Da mesma forma, ferramentas de ‘gamificação’ voltada ao ensino da disciplina de engenharia de *software* podem suprir essa necessidade, uma vez que ele difere das tradicionais aulas teóricas e leituras de materiais específicos. Desta forma, um caso de estudo foi levantado, com a finalidade de analisar se uma ferramenta de aprendizado simulado poderia ter resultados mais positivos que aulas convencionais.

**PALAVRAS-CHAVE:** gamificação; engenharia de software; objeto de aprendizagem; **SimSE**; simulador.

**ABSTRACT:** *The learning of the concepts of software engineering is of paramount importance, since the job market is becoming more demanding and requiring professionals who possess such technical skills. The practical teaching of these concepts is the best choice, since students only have the right experience when they are in the*

---

<sup>1</sup> Aluno de Bacharelado em Ciência da Computação e de Iniciação Científica.

<sup>2</sup> Professor Dr. Orientador.

*Software industry, and often until then, these concepts have not settled satisfactorily. The concept of 'gamification' has brought great advantages in teaching in general, and can be applied in the educational or professional environment. This method stands out for engaging those who use it by learning in a friendly and fun way. Likewise, 'gamification' tools aimed at teaching the discipline of software engineering can meet this need, since it differs from traditional theoretical lessons and specific material readings. In this way, a study case was raised, in order to analyze if a simulated learning tool could have more positive results than conventional classes.*

**KEYWORDS:** *gamification; Software Engineering; Learning object; SimSE; Simulator.*

## **1. INTRODUÇÃO**

O conhecimento em Engenharia de Software é peça fundamental na formação de um profissional na indústria de software e um método de ensino que vise aperfeiçoar as habilidades técnicas dos estudantes é fundamental para adequar esses estudantes ao mercado de trabalho. Muitas vezes, porém, o estudante precisa de um pouco mais do que apenas uma base teórica para construir seu conhecimento, pois se depender apenas disso, ele ficará defasado. Os conceitos fundamentais da Engenharia de Software e de gerenciamento de projetos devem ser abordados levando-se em conta a necessidade dos alunos em passarem por uma etapa de aplicação prática dos conceitos e avaliação dos experimentos realizados.

Objetos de Aprendizagem tem ganhado espaço no processo de ensino-aprendizagem, tendo sido adotado como importante recurso para a construção do conhecimento.

As novas gerações possuem uma maior proximidade e afinidade com a tecnologia. Dessa forma o uso da mesma para apresentação de conceitos teóricos através de uma linguagem fácil e objetiva que eles têm acesso é uma alternativa benéfica a educador e aos próprios estudantes. A partir desse fato se viu que a utilização de jogos digitais e ambientes de simulação para incrementar o conhecimento dos alunos se tornaram uma forte opção.

Dentro deste contexto, foi-se explorado a possibilidade do uso de jogos e ambientes de simulação no meio acadêmico, inclusive para o ensino dos conceitos de engenharia de software. O conceito utilizar elementos chaves próprios de jogos e aplica-los em um contexto genérico ficou conhecida como “gamificação”, um termo incorporado da língua inglesa, cujo original é “*gamification*”.

A ferramenta *SimSE*, *Simulation Software Engineer*, é um ambiente de simulação de Engenharia de Software educacional. Seu objetivo é demonstrar de forma prática os conceitos teóricos de Engenharia de Software dados aos alunos. O *SimSE* permite que os alunos pratiquem um processo de Engenharia de Software "virtual" em uma configuração gráfica e interativa em que seus avanços e erros são demonstrados de forma que o aluno possa analisá-los.

De acordo com PAIVA et al (2015), alguns exemplos de gamificação tem sido utilizado na área da Tecnologia da Informação, como por exemplo, *Ribbon Hero 2*, que é um aplicativo para a área administrativa de empresas, que possibilita aos usuários aprenderem ou treinarem determinadas habilidades. Em outro exemplo, a plataforma Duolingo adota a experiência de acumular pontos para medir o progresso de aprendizado de uma língua estrangeira pelo usuário.

Há atualmente empresas especializadas em gamificação que fornecem aplicativos que instalados no website de uma empresa ou no seu *back-office* permitem ‘gamificar’ toda a experiência, seja para clientes ou funcionários desta. Exemplos são a *Badgeville* e *Bunchball* (ARAÚJO, 2014).

De acordo com PAIVA et al (2015), a gamificação permite utilizar certos elementos, como pontuação ou premiação, para melhorar as habilidades e o engajamento, maximizar o aprendizado, e para promover mudanças no comportamento e na socialização dos indivíduos.

Segundo VIANNA (2013) empresas geralmente visam se adequarem aos diversos aspectos, como a comunicação com o cliente, aprendizado, recrutamento e formação. O autor ressalta que não se pode confundir gamificação com criação de jogos eletrônicos. A gamificação utiliza os métodos de jogos em contextos que não são relacionados a entretenimento, aplicando assim no contexto corporativo.

## **2. OBJETIVOS**

O presente projeto tem como objetivo analisar o uso de gamificação como metodologia de aplicação prática na disciplina de Engenharia de Software utilizando a ferramenta SimSE.

### **3. JUSTIFICATIVA**

Cada vez mais o aprendizado dos conceitos de Engenharia de Software tem se tornado necessário para se formar excelentes profissionais na área. Tais conceitos são elementos chaves e até mesmo diferenciais na contratação dos mesmos e contribui, também, no crescimento dentro da carreira. Atualmente, há uma grande preocupação por parte dos acadêmicos e educadores da área em encontrar algum método de ensino eficiente para fixar tais conceitos.

Um dos grandes obstáculos no ensino da Engenharia de Software se dá ao fato de seus conceitos serem em grande parte teóricos. Além disso, desenvolver um produto possui vários fatores, desde os voltados ao processo de desenvolvimento, como levantar requisitos, modelagem e desenvolvimento, como também no gerenciamento do projeto.

A geração atual possui dificuldade em compreender materiais densos, não conseguindo, muitas vezes, abstrair elementos chaves dos mesmos. Em contrapartida, com seu grande contato com redes sociais, jogos eletrônicos e outros tipos de conteúdo dinâmico, os jovens desenvolveram uma certa facilidade em aprender de forma prática. Porém é um grande desafio para os educadores desenvolverem materiais práticos para seus alunos.

Ao utilizar um ambiente virtual, simulando situações reais e problemas enfrentados na Engenharia de Software, é esperado que a absorção e aproveitamento do conhecimento apresentado seja maior entre os alunos.

O devido projeto teve como proposta realizar estudos sobre a aplicação da ferramenta *SimSE – Software Engineering Simulation Environment* no âmbito acadêmico com o objetivo de analisar seus pontos positivos e negativos.

### **4. METODOLOGIA**

A metodologia adotada para o desenvolvimento deste trabalho seguiu a implementação das seguintes etapas:

- Familiarização com a ferramenta de simulação SimSE – Modelo Incremental;
- Apresentação da ferramenta para alunos do 3º Ano do Curso de Bacharelado em Ciência da Computação;
- Manipulação da ferramenta pelos alunos;
- Aplicação da avaliação e questionário de satisfação aos alunos;

- Análise dos dados coletados;

## **5. ESTUDO DE CASO**

No mês de julho de 2017 foram ministradas quatro aulas para o terceiro ano do curso de Bacharelado em Ciência da Computação da Fundação Educacional do Município de Assis – FEMA. As aulas foram dadas para um grupo de 31 estudantes, a maioria do sexo masculino, com a faixa etária entre 18-25 anos. As duas primeiras aulas serviram como uma apresentação da ferramenta SimSE e um repasse de alguns conceitos de Engenharia de Software. As outras duas aulas serviram para os estudantes experimentarem e replicarem seus conhecimentos na ferramenta, além da aplicação da avaliação e questionário de satisfação. As aulas foram ministradas no laboratório de informática da FEMA e a duração foi de aproximadamente duas horas cada aula.

Para a apresentação da ferramenta foi utilizado um aparelho projetor Data Show e um computador. Para ser possível o download da ferramenta, bem como os alunos explorarem o site da universidade, foi disponibilizada também a internet. Para o manuseio por parte dos alunos, foram utilizados os computadores do laboratório de informática, sendo possível também a utilização de computador pessoal por aqueles que possuíam.

A avaliação e questionário foi feito através da ferramenta Google Forms e disponibilizado através de e-mail para os estudantes.

## **6. AVALIAÇÃO DOS RESULTADOS OBTIDOS**

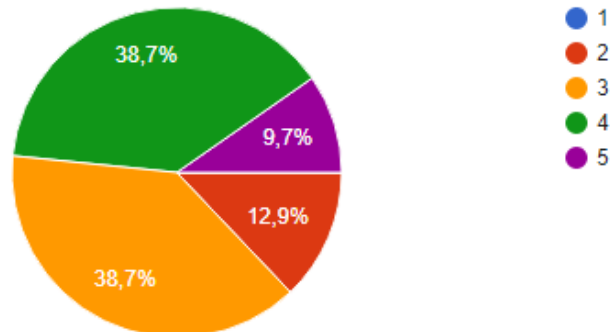
Após a interação dos alunos com a ferramenta de simulação SimSE, foi disponibilizado aos mesmos uma avaliação, contendo quatorze questões, para testar seus conhecimentos e um questionário de satisfação para recolher suas opiniões contendo nove questões.

### **6.1. Questionário**

Os gráficos a seguir representam as respostas voltadas ao questionário de satisfação sobre a experiência pessoal dos alunos com a ferramenta, conforme pode ser observado:

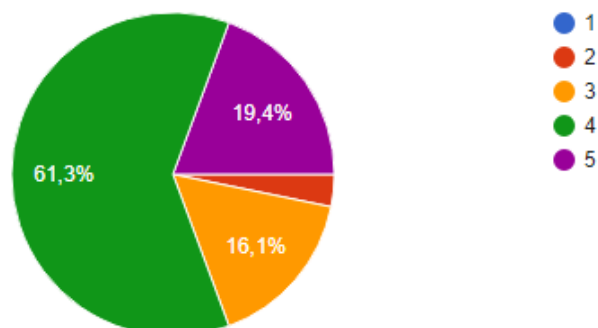
## Qual foi seu grau de dificuldade para utilizar a ferramenta SimSE?

Onde 1 significa Muita Dificuldade e 5 Nenhuma Dificuldade



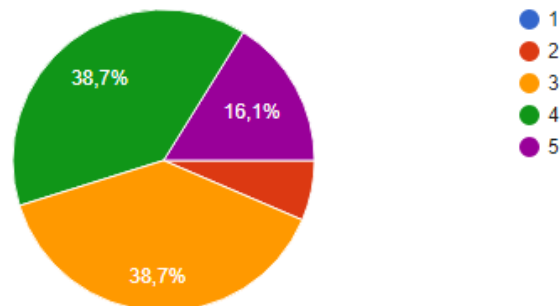
## Você acredita que uma ferramenta de simulação pode contribuir para o aprendizado dos conceitos de Gerenciamento de Projetos de Software?

Onde 1 significa Discordo Completamente e 5 Concordo Completamente



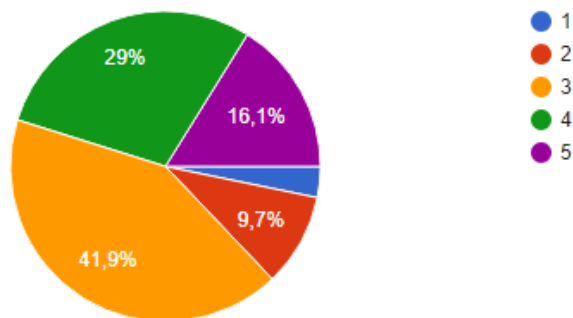
## A ferramenta SimSE é eficiente no ensino de Gerenciamento de Projetos de Software?

Onde 1 significa Discordo Completamente e 5 Concordo Completamente



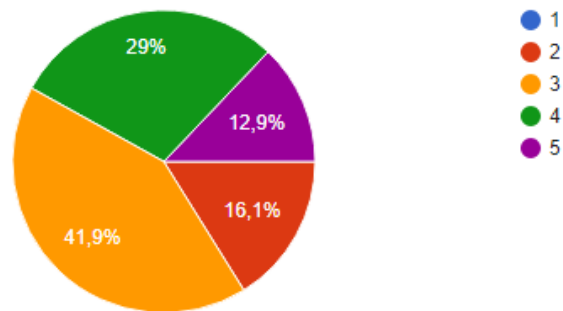
## Você teve facilidade em entender os conceitos de Gerenciamento de Projetos de Software com o uso da ferramenta SimSE?

Onde 1 significa Muita Dificuldade e 5 Nenhuma Dificuldade



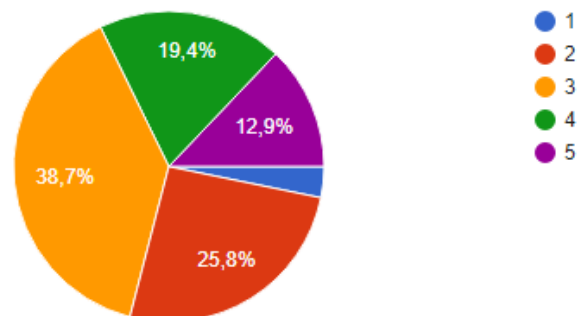
## Você acredita que uso de ferramentas como o SimSE pode completar seus conhecimentos de Gerenciamento de Projetos de Software?

Onde 1 significa Discordo Completamente e 5 Concordo Completamente



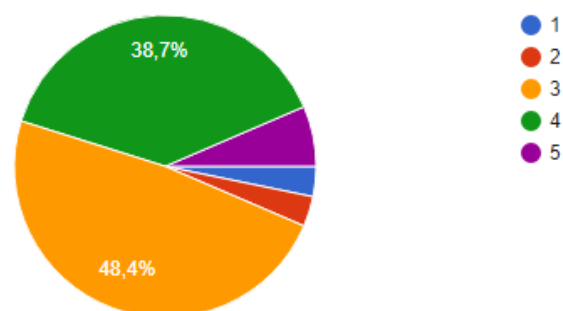
A ferramenta SimSE lhe agregou algum conhecimento adicional, que você não aprendeu em aula?

Onde 1 significa Muita Dificuldade e 5 Nenhuma Dificuldade



Para o entendimento de Gerenciamento de Projetos de Software, como você considera que foram os resultados usando o SimSE?

Onde 1 significa pouco útil para o entendimento e 5 significa muito útil para o entendimento



Abaixo está disponível duas questões dissertativas onde os alunos podiam dar sua opinião sobre a ferramenta, como se segue:



## Qual foi a sua maior dificuldade no aprendizado pela forma prática apresentada pelo SimSE?

Nenhuma (3)
Conceito (2)
Aumentar a satisfação do cliente e a precisão.
Conhecer os conceitos em questão
Descobrir a sequencia na qual o projeto deve ser projetado.
Entender de uma forma geral oque deve ser realizado ao customers
O projeto tinha muitas alterações feitas pelo cliente, porém, como o jogo não tem a comunicação entre analista e cliente, ficava um pouco difícil entender as mudanças.
Descobrir o que cada opção de tarefa faz.
As mudanças constantes do cliente
Entender a dinamica do trabalho em equipe.
Entender como funciona a lógica de tempo TICK e controlar e gerenciar os recursos apresentados.
as telas que informam os pontos do projeto não fica fixa, a tela de habilidade dos personagens também não fica fixa. Isso dificulta na escolha do que cada personagem deve fazer
Entender o significa dos campos
O idioma
A ordem das atividades e o tempo delas
Os termos apresentados em inglês
O idioma, ordem das atividades.
Não tive dificuldade.
como designar os funcionarios para suas funções
Entender o passo a passo, pois as porcentagens variam muito dependendo os funcionários que são selecionados.
decidir as ações de acordo com as várias estatísticas apresentadas
gerenciamento dos módulos
A ferramenta é meio confusa.
Divisão de tempo e tarefas
Distinguir qual era o objetivo final a ser concluído antes de entregar os módulos
A sequencia de processos a serem feitas no modelo
manipular a ferramenta
a mudança de pedido do cliente

## Na sua opinião, qual seria sua sugestão para tornar a ferramenta mais adequada?

Sem erro no 500

Gráficos melhores, mais personagens com mais características.

Poderia ter um local que explica os conceitos utilizados no SimSe

Ela poderia ser mais interativa , dar mais dicas sobre o que o modulo a ser desenvolvido pode melhorar e como ele pode melhorar. E não travar no tick 500

Portugues-br

A ferramenta possui alguns bugs que atrapalham a experiência, e não dá pra entender muito bem o que o cliente deseja mudar (se é relacionado a design, implementação, etc)

Uma tradução para pt-br para facilitar a aprendizagem de quem não possui entendimento da língua inglesa

Uma explicação das tarefas q o que cada uma faz e no que ela influencia no projeto.

Mudanças de layout e trazer mais informações ao usuário sobre oque está acontecendo.  
Também seria uma boa ideia implementar uma fase de tutorial, explicando e exemplificando cada etapa do desenvolvimento e as ferramentas disponíveis ao jogador.

Deveria ser mais intuitiva apenas

Um tutorial, fazer com que as abas sejam sempre visíveis junto as tabelas de detalhes. Correção de alguns bugs como TICK 500 trava o jogo.

as telas que informam os pontos do projeto ficarem fixas, a tela de habilidade dos personagens também ficarem fixa. Tradução para linguagem portuguesa. Tutorial para ferramenta em forma de demo indicando os passo principais.

Traduzir

Ser traduzido

Nenhuma

Avisos de erro

Uma tradução para o português e mais opções gráficas

Ter mais idiomas.

Nenhuma, pois a ferramenta é exelente

Poder observa mais os projetos aprofudamente

Tradução para Portugues
ter uma tradução
Um tutorial completo de como e quando usar as funcionalidades do aplicativo
maior contato com as estatísticas seguindo uma abordagem de ensinar o básico de cada raciocínio
ter mais dicas
A ferramenta deveria seguir uma sequência e não mostrar novamente módulos que já foram finalizados.
Maior entendimento da mesma e do uso de seus recursos
Um guia prático de como funciona o processo de finalização para entrega dos módulos
Poderia ter outras ideias de ensinamento.
dicas mais claras
não tenho sugestão, aparentemente a ferramenta é boa

Com a análise dos gráficos, é visto que os estudantes avaliaram a ferramenta de forma muito positiva. Por parte dos estudantes, a ferramenta pode sim contribuir para o aprendizado.

Em análise das questões dissertativas é possível afirmar que a ferramenta possui alguns pontos fracos, como o *layout* da ferramenta não ser otimizado para o jogo. Foi discutido que se torna necessário abrir e fechar várias janelas para poder analisar as informações e dar procedência ao jogo, o que dificulta sua fluidez. Houve também críticas à carência de informação por parte da ferramenta em sinalizar de alguma forma ao jogador se estão no caminho certo. A única maneira de obter tal informação seria no final do jogo, ou analisando os valores que a ferramenta oferece. Esse último, porém, se torna uma opção ainda mais complicada, uma vez que em um primeiro contato com a ferramenta é difícil compreender o que cada valor representa.

Grande parte dos pontos sugeridos pelos estudantes se resumem em uma melhor comunicação por parte da ferramenta. Atitudes como uma maior interação com o usuário e uma interface mais amigável, foram um desses pontos. A ferramenta possui uma breve explicação de como ela se comportará durante a jogatina, porém é um texto muito denso com muitos detalhes de mecânica do jogo e pouco voltado aos conceitos de Engenharia de Software. A falta de informação sobre como começar e o jogo não sinalizar quando uma atitude errada for tomada também foi apontado como uma possível melhoria. No site da faculdade da Califórnia existe alguns vídeos tutoriais sobre como usar a ferramenta, porém isso leva a outro ponto levantados pelos estudantes, que é a dificuldade com a língua inglesa.

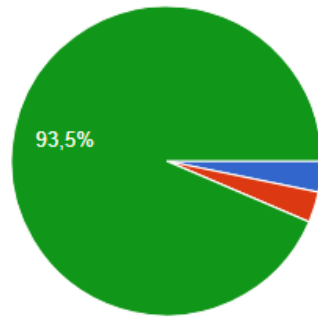
## 6.2. Avaliação

As questões a seguir fazem parte da avaliação para testar o conhecimento dos estudantes, seguida pelo gráfico gerado de acordo com suas respostas. As respostas corretas estão marcadas com um asterisco (\*):

- I) Quais são os conceitos bases do modelo incremental?
- a) No desenvolvimento de software, cada componente é trabalhado e concluído por vez, na ordem de maior prioridade, até a conclusão do produto.
  - b) É criado várias versões de cada modulo e apresentados ao cliente para o mesmo acompanhar o processo de desenvolvimento.
  - c) O desenvolvimento de software é dividido em módulos, funcionalidades bem definidas do programa, e gradativamente o software vai sendo desenvolvido baseado nesses módulos. (\*)
  - d) O modulo incremental serve apenas como uma metodologia para o desenvolvimento do software, visando a velocidade da equipe.



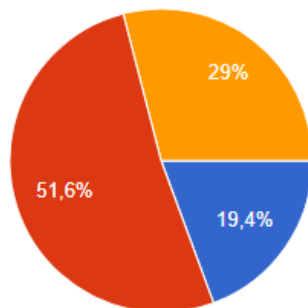
- II) Qual a importância do Gerente de Software no projeto?
- a) O gerente de software é a pessoa que vai conversar com o cliente, levantar seus requisitos e garantir que o software esteja da forma que ele pediu. É a pessoa que conhece o negócio.
  - b) O gerente de software tem como função contratar funcionários para o desenvolvimento e gerenciar suas necessidades como garantir ferramentas necessárias para a equipe. Ele não precisa conhecer o produto a fundo.
  - c) O gerente de software é a pessoa que contrata o serviço de uma equipe de desenvolvimento, sendo ele o intermediador entre cliente e desenvolvedores.
  - d) O gerente de software tem como objetivo delegar quem irá exercer cada função para melhor atender a demanda do software. Ele é responsável por fazer as coisas caminharem. (\*)



- O gerente de software é a pessoa que vai conversar com o cliente, levantar seus requisitos e garantir q...
- O gerente de software tem como função contratar funcionários para o desenvolvimento e gerenciar suas...
- O gerente de software é a pessoa que contrata o serviço de uma equi...
- O gerente de software tem como objetivo delegar quem irá exercer c...

## II) Qual a importância do Analista do Sistema?

- a) O analista deve, com bases nos requisitos já documentados, desenvolver o modelo conceitual do software, como esboço de tela e diagramas de classe, entidade-relacionamento e casos de uso.
- b) O papel do analista é recolher todas as informações sobre o que o cliente precisa, as funcionalidades que ele busca para seu produto. Com as informações recolhidas, ele as passa para os próximos integrantes.
- c) A função do analista é recolher as informações sobre o produto que o cliente necessita. Com base nelas ele deve montar toda a estrutura básica do produto, como banco de dados e desenvolvimento de tela.
- d) Nenhuma das anteriores. (\*)



- O analista deve, com bases nos requisitos já documentados, desenvolver o modelo conceitual do...
- O papel do analista é recolher todas as informações sobre o que o cliente precisa, as funcionalidades que ele...
- A função do analista é recolher as informações sobre o produto que o cliente necessita. Com base nelas...
- Nenhuma das anteriores.

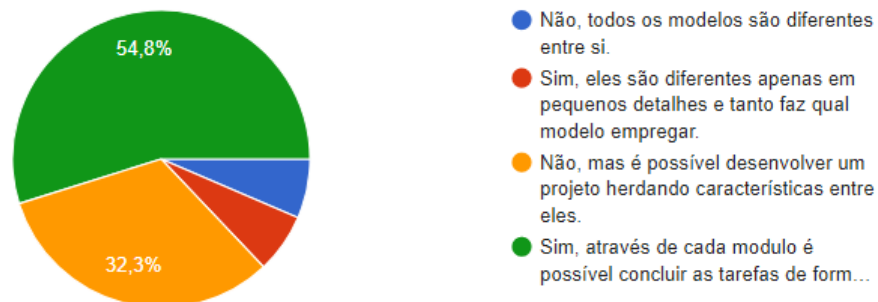
## III) Quantos são a quantidades de ciclos necessários para concluir um projeto utilizando o modelo incremental?

- a) Não se pode ultrapassar 4 ciclos.
- b) A quantidade de ciclos é indefinida e depende do projeto. (\*)
- c) Não é possível desenvolver um projeto com menos de 6 ciclos.
- d) Não é utilizado ciclos no modelo incremental.



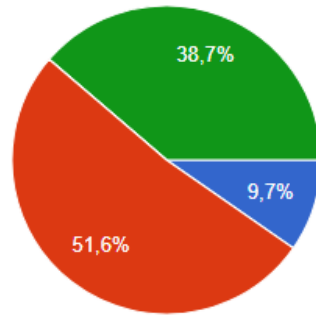
IV) Existe semelhança entre o modelo incremental com os demais modelos, como o modelo cascata e o prototipação?

- Não, todos os modelos são diferentes entre si.
- Sim, eles são diferentes apenas em pequenos detalhes e tanto faz qual modelo empregar.
- Não, mas é possível desenvolver um projeto herdando características entre eles.
- Sim, através de cada modulo é possível concluir as tarefas de forma semelhante a outros modelos. (\*)



V) O que é um modulo no modelo incremental?

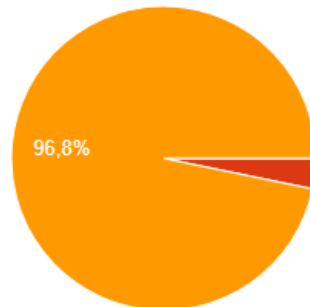
- É uma tarefa menor no desenvolvimento de software, como por exemplo, levantar requisitos, criar casos de uso, criar diagrama de caso de usos, etc.
- É um pedaço razoável do projeto, geralmente um modulo é uma funcionalidade bem definida do projeto, um programa. Poderia ser uma tela de cadastro de cliente, por exemplo. (\*)
- O conceito de modulo não é aplicado no modelo incremental.
- Um modulo é uma fase de desenvolvimento que pode repetir mais de uma vez para cada ciclo de desenvolvimento. Como por exemplo, a codificação do projeto é uma fase que se repetirá em todos os ciclos de desenvolvimento.



- É uma tarefa menor no desenvolvimento de software, como por exemplo, levantar requisitos, cri...
- É um pedaço razoável do projeto, geralmente um módulo é uma funcionalidade bem definida do proj...
- O conceito de módulo não é aplicado no modelo incremental.
- Um módulo é uma fase de desenvolvimento que pode repetir...

VI) Na ferramenta SimSE, o que significa a opção 'Evolve Code' ou 'Evoluir Código'?

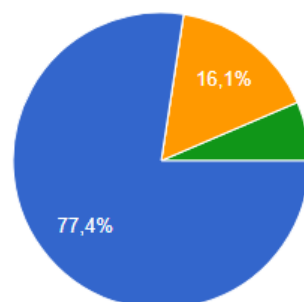
- a) Significa o ato de iniciar a codificação de um módulo.
- b) Significa, com base em progressos de codificação anteriores, unir o programa em um único, desta forma evoluindo o projeto de fato.
- c) Significa testar o projeto e fazer os devidos ajustes para que fique próximo ao que o cliente queria. (\*)
- d) Nenhuma das anteriores.



- Significa o ato de iniciar a codificação de um módulo.
- Significa, com base em progressos de codificação anteriores, unir o programa em um único, desta forma evoluindo o projeto de fato.
- Significa testar o projeto e fazer os devidos ajustes para que fique próximo ao que o cliente queria.
- Nenhuma das anteriores.

VII) Na ferramenta SimSE, quem é o funcionário mais bem qualificado para fazer integração dos módulos?

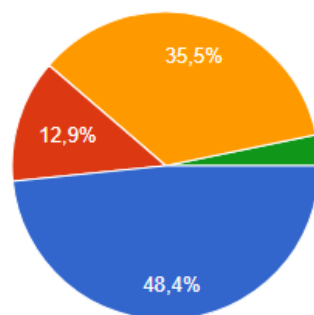
- a) Amy. (\*)
- b) Bob.
- c) Karl.
- d) Lola.



- Amy.
- Bob.
- Karl.
- Lola.

VIII) Na ferramenta SimSE, o que significa a opção 'Integrate' ou 'Integrar'?

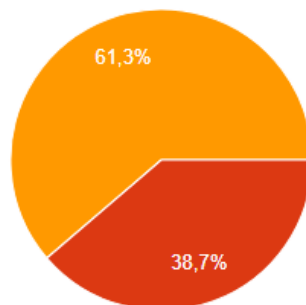
- a) Integrar significa juntar os módulos concluídos para o produto final, podendo esse já ser entregue para o cliente. (\*)
- b) Integrar significa fazer a mudança de fase, ou seja, sempre que você concluir alguma fase, como levantar requisito ou codificação, você deve integrar para concluir de fato aquela fase.
- c) Integrar significa juntar todos os módulos para o produto final, devendo então ser apenas a última etapa do projeto.
- d) Nenhuma das anteriores.



- Integrar significa juntar os módulos concluídos para o produto final, podendo esse já ser entregue par...
- Integrar significa fazer a mudança de fase, ou seja, sempre que você concluir alguma fase, como levanta...
- Integrar significa juntar todos os módulos para o produto final, devendo então ser apenas a última...
- Nenhuma das anteriores.

IX) Na ferramenta SimSE, em que momento eu faço o design do meu projeto?

- a) O design deve ser feito em paralelo com o levantamento de requisito.
- b) O design deve ser feito após o levantamento de requisito.
- c) Após o levantamento de requisito e demais análises, é recomendado fazer o design. O design pode ser feito também caso 'evoluir código' diminuir o progresso de design. (\*)
- d) Nenhuma das anteriores.

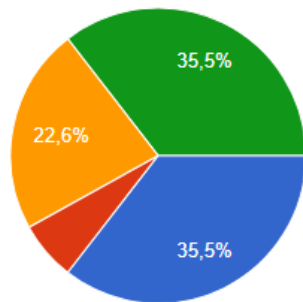


- O design deve ser feito em paralelo com o levantamento de requisito.
- O design deve ser feito após o levantamento de requisito.
- Após o levantamento de requisito e demais análises, é recomendado fazer o design. O design pode ser feito também caso 'evoluir código' diminuir o progresso de design.
- Nenhuma das anteriores.

X) Na ferramenta SimSE, o que significa o campo 'Changability' ou 'Facilidade de Mudança'?



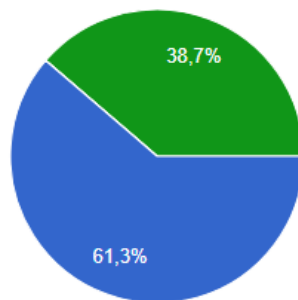
- a) É uma variável aleatória, não controlável, que representa a chance do cliente de enviar mais requisitos ao sistema.
- b) É uma variável bem definida, não controlável, que representa acontecimentos aleatórios que podem atrapalhar no desenvolvimento do software.
- c) É uma variável que aumenta à medida que o cliente não está satisfeito com o software e representa a chance do cliente de enviar mais requisitos ao sistema. (\*)
- d) É uma variável aleatória que define a chance do cliente de mudar tudo sobre o que ele quer no software.



- É uma variável aleatória, não controlável, que representa a chance do cliente de enviar mais requisitos...
- É uma variável bem definida, não controlável, que representa acontecimentos aleatórios que pod...
- É uma variável que aumenta à medida que o cliente não está satisf...
- É uma variável aleatória que define a chance do cliente de mudar tudo so...

XI) Na ferramenta SimSE, que atitude eu devo tomar caso o cliente não esteja satisfeito com o software.

- a) Utilizar opção Evoluir o código. (\*)
- b) Desfazer o passo anterior e continuar.
- c) Começar todo o desenvolvimento do zero.
- d) Fazer análise de risco e de dificuldade e corrigir esse problema gradativamente.



- Utilizar opção Evoluir o código.
- Desfazer o passo anterior e continuar.
- Começar todo o desenvolvimento do zero.
- Fazer análise de risco e de dificuldade e corrigir esse problema gradativamente.

Após as questões objetivas, foram realizadas duas questões dissertativas. Segue o exemplo de algumas das respostas dadas pelos alunos, respectivamente:

XII) O que vem antes, análise de dificuldade ou levantamento de requisito? Explique.

levantamento de requisito (2)

levantamento de requisitos, já que é necessário avaliar os requisitos para ver seus riscos ;D

Levantamento de Requisito, pois é necessário possuir os requisitos para conseguir realizar a análise de dificuldade

Levantamento de Requisitos , pois o quais seriam as análises se ele não teria nada para analisar.

Levantamento de requisitos, pois é necessário saber o que o cliente requisitou para o programa pra depois analisar a dificuldade de implementar aquela funcionalidade.

Levantamento de Requisitos. Pois precisa saber como o programa irá funcionar e o que ele irá ter para poder saber as dificuldades encontradas no sistema.

Levantamento de requisito, pois a partir ele é o primeiro passo no desenvolvimento de um projeto.

Levantamentos de requisitos, já que se é necessário saber o que irá se desenvolver para ter uma análise da sua dificuldade.

Levantamento de requisito, não tem como analisar dificuldade de algo inexistente .

Levantamento de Requisitos

XIII) Descreva qual processo você utilizou em sua simulação e que score você tirou (Exemplo, primeiro eu coloquei fulano para codificar, em seguida...).

Fui fazendo ao mesmo tempo, conforme a liberdade dos personagens e em suas áreas mais experientes. 30.

91,81. Primeiro fiz o levantamento de requisitos depois análise de risco, depois comecei a implementação.

Primeiramente, coloquei os funcionários para realizar o levantamento de requisito, logo já foi iniciado o design e com isso foi iniciado o desenvolvimento de código, o score que eu tirei foi de 45%

A principio coloquei o Karl e a Lola para levantar requisitos, após isso coloquei o Bob e Amy para levantar os riscos de análise e a análise de dificuldade , após isso o Bob para fazer o Design e então a Amy para desenvolvimento de Código.

Realizei uma performance aonde ninguém fica parado (55)

Fiz o levantamento de requisitos com o Karl e a Lola, após isso, fiz o design com o Bob e em paralelo a análise de risco com Karl e Lola novamente. Após terminados todos os módulos, comecei a implementação com a Amy e o Bob, fiz a integração como Karl e a Lola e por fim evolui o código com a Amy e o Bob.

1- Utilizei o Karl e a Lola para levantar requisitos. 2-Requisitos levantados utilizei o Bob para design enquanto Karl e Lola levantavam os riscos e análise de dificuldade. 3- Após o design coloquei a Amy para programar enquanto Karl e Lola levantavam requisitos do próximo módulo. E assim foi seguindo a sequência. Meu Score: 99.75;

Eu fiz um módulo de cada vez e coloquei primeiro os personagens com maior facilidade de levantar requisitos e

depois o desing e implementação.

Primeiramente eu coloquei toda a equipe no primeiro módulo para fazerem o levantamento de requisitos, análise de riscos e análise de dificuldade. Após isso Bob ficou responsável pelo design, junto da Amy. Karl e Lola começaram a documentação do próximo módulo.

Assim que o design era finalizado, Amy e Karl fariam a implementação e evolução do código, caso necessário junto de Bob ou Bob substituiria Karl.

A integração era realizada pelos Bob,Amy e Karl, caso possível, se não por Karl e mais um membro que não fosse a Lola. Enquanto os demais membros trabalhavam em outros módulos.

Após todos os módulos estarem completos, era feito o submit.

Eu usei todos trabalhando em tudo modulo por modulo , terminei em 470 horas e meu score foi 85...

Primeiramente realizei o Levantamento de Requisitos com Karl e Lola.

Fiz o Design com o Bob e a Análise de Risco com Amy.

Após a finalização das atividades.

Iniciei o desenvolvimento com a Amy, enviando Karl para Análise de Riscos, Bob para Designer e mantendo Lola sempre no levantamento de requisitos do módulo.

Mantive esse processo até que houvesse o requerimento de alguma alteração.

Através da análise geral, nota-se que não só os estudantes avaliaram a proposta de uma ferramenta de simulação de engenharia de software de forma positiva, como também obtiveram grande êxito no manuseio dela. Grande parte dos estudantes obtiveram pontuação final na faixa de 80 a 100 por cento além de nenhum estudante ter tirado nota menor do que 50 por cento. É visto que a maioria utilizou conceitos do modelo incremental, que foi o objeto de estudo da ferramenta.

Pode se analisar, também, que de acordo com o questionário a grande maioria acertou as questões disponibilizadas, comprovando a fixação do conhecimento.

Por fim, foi analisado em aula durante a aplicação da ferramenta, que a aceitação dos jovens estudantes e a cooperação em entender a ferramenta se mostrou um ponto muito favorável. Muitos dos alunos, após obter resultados não muito motivadores, não desistiram e começaram um novo jogo, agora com o conhecimento do erro em mente e continuaram até obter um resultado satisfatório.

## **7. CONCLUSÃO**

Esse projeto teve como objetivo a realização de um estudo sobre a plataforma de simulação de engenharia de software SimSE e analisar quais seriam as vantagens e desvantagens de utilizá-la em ambiente acadêmico, aprimorando o ensino teórico. Para tal, foi feita a análise aprofundada sobre a ferramenta para a apresentação para os estudantes do terceiro ano do curso de Bacharelado em Ciências da Computação da FEMA de modo a confirmar que o uso de gamificação na disciplina de engenharia de software pode trazer resultados melhores que o ensino tradicional.

Com o objetivo de avaliar o conhecimento que os alunos possam obter através da manipulação da ferramenta, foi-se realizado uma avaliação juntamente com um questionário de satisfação, sendo o primeiro objetivado em testar seus conhecimentos obtidos, enquanto o segundo para obter um retorno sobre o que acharam da ferramenta. Os resultados esperados era de que se os conceitos de engenharia de software fossem repassados de forma mais prática, utilizando as vantagens dadas pela metodologia baseada em gamificação, a absorção de conhecimento seria elevada, pois com uma maior interação com o conteúdo exposto, maior engajamento os alunos teriam.

De acordo com os resultados obtidos foi visto que de fato os estudantes se engajaram e gostaram da ferramenta. Mesmo com as limitações técnicas do ambiente, os alunos deram o melhor de si para tirarem resultados melhores. Fica claro também que os estudantes obtiveram bons resultados na avaliação, o que significa que de fato aprenderam com a ferramenta.

Em suma, é possível concluir que o uso da ferramenta SimSE em sala de aula é de grande ganho, tanto para os estudantes, quanto para o professor. Ela fornece uma nova abordagem do conteúdo capaz de reforçar os conhecimentos aprendidos em sala de aula de uma forma divertida e interativa, além de trazer uma experiência única, mais próxima do mercado de trabalho.

## **REFERÊNCIAS BIBLIOGRÁFICAS**

PAIVA, R.; BARBOSA, A.; BATISTA, E. PIMENTEL, D.; BITTENCOURT, I. **Badges and XP: An Observational Study About Learning**. Proceedings of the 2015 IEEE Frontiers in Education Conference. 326 - 333. EUA, 2015.

VIANNA, Ysmar. **Gamification, Inc : como reinventar empresas a partir de jogos**. 1ª Ed. Rio de Janeiro: MJV Press, 2013. ISBN 978-85-65424-09-7 (e-book)

ARAÚJO, Inês Cardoso, and Ana Amélia Carvalho. **Gamificação: uma oportunidade para envolver alunos na aprendizagem**. Atas do 2º Encontro sobre Jogos e Mobile Learning. Braga: CIED (2014).