Atividade disciplina – Engenharia de Software

Aluno: Cezarino Martins da Hora

Data: 07/10/2021

**1. Grandes projetos podem ser gerenciados de forma ágil? Como é possível? É confiável?**

Na maioria das vezes, apenas projetos que estão conectados diretamente com os clientes são elegíveis para usar metodologias ágeis de desenvolvimento. Projetos com requisitos bem definidos e que requerem grandes investimentos devem usar metodologias que se caracterizam por fases bem definidas de desenvolvimento. Criar muitas dependências entre projetos com desenvolvimento ágil aumenta, dramaticamente, a complexidade de gerenciamento, assim como grandes equipes. O uso equivocado de metodologias de desenvolvimento pode acarretar no fracasso dos projetos.

As metodologias de desenvolvimento ágeis foram concebidas para entregar resultados rápidos em cenários complexos que exigem alto grau de adaptabilidade, buscando produtividade e criatividade.

O PMI.org (Project Management Institute) desenvolveu um conjunto de práticas para lidar com metodologias ágeis para grandes projetos, afirmando ser possível executar projetos ágeis em grande escala. A ideia não é aumentar os níveis de complexidade dos projetos ágeis, mas sim trazer o desenvolvimento ágil para o mundo do gerenciamento estruturado de projetos.

Aqui vem a reflexão. Na prática, será que alcançaremos os objetivos de negócios que exigem flexibilidade e dinamismo para competir no mercado introduzindo metodologias para gerenciar várias equipes de desenvolvimento ágil, priorizando tarefas, gerenciando os backlogs de forma integrada, entre outras restrições? Será que conseguiremos mais produtividade, criatividade e alinhamento com o mercado tirando a independência das equipes?

**2. Gerenciamento ágil para qualquer tipo de projeto Construção de edifícios, aviões, robôs Como é possível?**

Os sistemas críticos são sistemas sociotécnicos ou técnicos, dos quais as pessoas dependem.

Atualmente são três os tipos de sistemas críticos, tais como: sistema crítico de segurança, sistema crítico de missão e sistema crítico de negócio.

Os sistemas considerados críticos precisam ser de total confiança, mas para atingir esse requisito é preciso: disponibilidade, confiabilidade, segurança e proteção.

Para especificar um sistema crítico, é preciso compreender os riscos e gerar requisitos de confiabilidade para lidar com eles. No entanto, é preciso saber identificar os riscos e aplicar a cada tipo de sistema crítico.

Em minha opinião os sistemas críticos não devem ser desenvolvidos por meio das metodologias clássicas de desenvolvimento pois as metodologias de desenvolvimento ágeis foram concebidas para entregar resultados rápidos em cenários complexos que exigem alto grau de adaptabilidade, buscando produtividade e criatividade, o que não é o caso da maioria dos sistemas críticos.

**4. Descreva com suas palavras resumidamente as reuniões no Scrum:**

**Sprint Planning Meeting**

Durante o Sprint Planning Meeting, o [**Product Owner**](http://www.desenvolvimentoagil.com.br/scrum/product_owner) descreve as funcionalidades de maior prioridade para a equipe. A equipe faz perguntas durante a reunião de modo que seja capaz de quebrar as funcionalidades em tarefas técnicas, após a reunião. Essas tarefas irão dar origem ao [**Sprint Backlog**](http://www.desenvolvimentoagil.com.br/scrum/sprint_backlog).

**Sprint - Reuniões Diárias - Daily Scrum Meeting**

Durante o Daily Scrum, cada membro da equipe provê respostas para cada uma destas três perguntas:

* O que você fez ontem?
* O que você fará hoje?
* Há algum impedimento no seu caminho?

**Sprint - Revisão - Sprint Review**

Durante o Sprint Review, o projeto é avaliado em relação aos objetivos do Sprint, determinados durante o [**Sprint Planning Meeting**](http://www.desenvolvimentoagil.com.br/scrum/sprint_planning_meeting). Idealmente, a equipe completou cada um dos itens do [**Product Backlog**](http://www.desenvolvimentoagil.com.br/scrum/product_backlog) trazidos para fazer parte do Sprint, mas o importante mesmo é que a equipe atinja o objetivo geral do Sprint.

**Sprint - Retrospectiva- Sprint Retrospective**

O Sprint Retrospective ocorre ao final de um Sprint e serve para identificar o que funcionou bem, o que pode ser melhorado e que ações serão tomadas para melhorar.

**5. Descreva com suas palavras resumidamente os papéis no Scrum:**

**Product Owner (PO)**

O Product Owner é a pessoa que define os itens que compõem o [**Product Backlog**](http://www.desenvolvimentoagil.com.br/scrum/product_backlog) e os prioriza nas [**Sprint Planning Meetings**](http://www.desenvolvimentoagil.com.br/scrum/sprint_planning_meeting).

**Scrum Master**

O Scrum Master procura assegurar que a equipe respeite e siga os valores e as práticas do [**Scrum**](http://www.desenvolvimentoagil.com.br/scrum). Ele também protege a equipe assegurando que ela não se comprometa excessivamente com relação àquilo que é capaz de realizar durante um Sprint.

O Scrum Master atua como facilitador do [**Daily Scrum**](http://www.desenvolvimentoagil.com.br/scrum/daily_scrum) e torna-se responsável por remover quaisquer obstáculos que sejam levantados pela equipe durante essas reuniões.

O papel de Scrum Master é tipicamente exercido por um gerente de projeto ou um líder técnico, mas em princípio pode ser qualquer pessoa da equipe.

**Time**

O Scrum Team é a equipe de desenvolvimento. Nela, não existe necessariamente uma divisão funcional através de papéis tradicionais, tais como programador, designer, analista de testes ou arquiteto. Todos no projeto trabalham juntos para completar o conjunto de trabalho com o qual se comprometeram conjuntamente para um Sprint.

Um Scrum Team típico tem de 6 a 10 pessoas, embora haja relatos de projetos [**Scrum**](http://www.desenvolvimentoagil.com.br/scrum) com equipes maiores.

**6. Faça uma pesquisa conforme o dia de seu nascimento:**

Se for par: Pesquise como é utilizado a ferramenta **Burndown Chart** bem como sites e sistemas para facilitar o seu uso.

R: 29

Se for ímpar:

Pesquise como é utilizado a ferramenta **Planning Poker** bem como sites e sistemas para facilitar o seu uso.

**O que é Planning Poker?**

O Planning Poker é uma estratégia muito aplicada em projetos que utilizam [metodologias ágeis](https://blog.betrybe.com/carreira/metodologias-ageis/), especialmente o [Scrum](https://blog.betrybe.com/carreira/metodologia-scrum/), cujo objetivo é **estimar o esforço necessário para o desenvolvimento das funcionalidades de uma aplicação**.

Apesar de ter sido definida por James Grenning em 2002, a técnica ganhou popularidade nos projetos após ser publicada no livro “Agile Estimating and Planning”, de Mike Cohn, e hoje é uma das ferramentas mais interessantes quando se trata de fazer estimativas.

Na prática, a técnica se baseia em um jogo, no qual cada pessoa que faz parte da [equipe de desenvolvimento](https://blog.betrybe.com/carreira/squad/) indica uma carta com a sua estimativa. Nessa carta, há um valor que representa o esforço que será gasto para desenvolver um determinado item do [backlog](https://blog.betrybe.com/carreira/o-que-e-backlog/).

A partir disso, o time realiza discussões, afim de avaliar os diferentes pontos de vista e encontrar um consenso entre eles.

### **O que significa estimar?**

Antes de prosseguir com o assunto, é importante esclarecer o que é fazer uma estimativa. De forma geral, podemos dizer que estimar é fazer uma “conta aproximada”. No caso do Planning Poker, essa conta busca [**avaliar a complexidade**](https://blog.betrybe.com/carreira/pensamento-sistemico/) **de cada tarefa do backlog, considerando o tempo que será gasto, o esforço que deverá ser empregado e o risco envolvido**.

Essa estimativa ajuda no planejamento do time, permitindo que o trabalho seja feito sem grandes imprevistos e dentro das expectativas. Para isso, é importante que o processo seja feito pelas pessoas que trabalham na equipe de desenvolvimento, pois elas têm experiência prática e conseguem mensurar melhor o quanto cada item demanda de trabalho.

## Quais os benefícios de usar o Planning Poker?

Utilizar o Planning Poker traz vários benefícios para o projeto. A [colaboração](https://blog.betrybe.com/soft-skills/colaboracao/) entre a equipe, por exemplo, é ampliada, visto que todas as [pessoas envolvidas no desenvolvimento do software](https://blog.betrybe.com/carreira/desenvolvedor-de-software/) precisam trocar ideias e opiniões até que a estimativa seja feita. Veja mais vantagens que a técnica traz nos tópicos a seguir!

### **Favorece o entendimento das tarefas**

Uma vez que as estimativas do Planning Poker são baseadas no consenso da equipe, um dos seus principais benefícios é **favorecer a discussão e**[**comunicação**](https://blog.betrybe.com/soft-skills/comunicacao/)**entre o time**. Com isso, as pessoas envolvidas no projeto conseguem compreender melhor as tarefas que estão no backlog.

Dessa forma, possíveis dúvidas sobre o desenvolvimento podem ser esclarecidas, o que reduz a possibilidade de que sejam inseridos [erros na aplicação](https://blog.betrybe.com/tecnologia/o-que-e-bug/). Além disso, a equipe [melhora seu desempenho](https://blog.betrybe.com/carreira/gestao-do-tempo-dicas-essencias/), visto que quanto mais descomplicada a tarefa está, mais fácil será para [desenvolver códigos](https://blog.betrybe.com/desenvolvimento-web/hello-world-ola-mundo/) e funções necessárias.

### **Ajuda a identificar a qualidade das user stories**

Outra consequência interessante das discussões favorecidas pelo Planning Poker é a identificação de user stories que apresentam uma estimativa muito alta. Em geral, isso indica que a história não está “quebrada” o suficiente e precisa ser dividida em várias tarefas ou ser mais detalhada.

Assim, é possível avaliar a qualidade das histórias que foram definidas e, a partir disso, a pessoa que está atuando como **Product Owner pode fazer os refinamentos necessários para garantir a adequação de todos os itens**.

### **Aumenta a confiabilidade das estimativas**

Em outros métodos de estimativa, é possível uma pessoa mais experiente acabe influenciando as demais durante o processo. Contudo, no Planning Poker essa possibilidade é reduzida, pois toda a equipe deve mostrar suas cartas ao mesmo tempo e cada um precisa explicar a complexidade que viu na tarefa, o que gera opiniões mais sinceras.

Dessa forma, as estimativas se tornam mais confiáveis, pois **diferentes pontos de vista são analisadas, favorecendo uma**[**tomada de decisão**](https://blog.betrybe.com/soft-skills/tomada-de-decisao/)**coerente** com a pluralidade de visões da equipe e baseada em um acordo geral.

Parte superior do formulário

## Como funciona o Planning Poker?

Como dissemos mais acima, o Planning Poker é uma espécie de jogo de cartas usado para estimar o tempo que cada tarefa precisará para ser cumprida. Para isso, **cada carta do jogo recebe o número de uma sequência de valores**.

Em equipes Scrum, normalmente a sequência é baseada numa parte da sequência de Fibonacci modificada, que é a seguinte: **0, ½, 1, 2, 3, 5, 8, 13, 20, 40 e 100**. Além desses números, o baralho também contém os símbolos “∞”, “?” e uma xícara de café. Obviamente, cada um deles tem um significado, que será explicado logo abaixo. Observe:

* **0**: é utilizado para itens que já foram finalizados ou que são tão simples que não precisam ser estimados;
* **½**: usado quando a user story é muito simples de ser implementada;
* **1, 2 e 3**: se referem a itens ainda pequenos, mas que apresentam uma complexidade um pouco maior e, por isso, não se encaixam nos valores anteriores;
* **5, 8 e 9**: usados para estimar itens médios, que requerem mais esforço para ser implementados;
* **20, 40 e 100**: são usados quando os itens são grandes e podem indicar a necessidade de refinamento da user story;
* **∞**: é usado quando a complexidade da tarefa é tão grande que não há como fazer uma estimativa para ela;
* **xícara de café**: esse símbolo é usado para pedir uma pausa. Normalmente, isso ocorre quando a equipe está cansada ou quando alguém acha necessário fazer uma reflexão maior antes da decisão ser tomada.

Explicado o significado de cada carta, é importante lembrar que **os valores da sequência apresentada não representam as horas de trabalho que serão gastas em cada item**. Eles são usados apenas para indicar a complexidade das tarefas do product backlog.

## O passo a passo para aplicar o Planning Poker no seu projeto!

Agora que você já entende como o baralho do Planning Poker funciona, nada melhor do que ver como esse jogo pode ser aplicado em um projeto, certo? Abaixo, explicamos como esse processo funciona, separando-o em 3 etapas básicas. Confira!

### **Estimativa individual**

Iniciada a reunião do Team Scrum, a pessoa que atua como Product Owner seleciona um item descrito no product backlog, descrevendo-o para toda a equipe. Em seguida, o item é discutido entre as partes e cada integrante do time de desenvolvimento faz a sua estimativa individual, escolhendo a carta do baralho que julga ser a mais adequada.

Já com as escolhas feitas, é chegado o momento de revelar as cartas. **Se todas as pessoas tiverem feito a mesma escolha, a estimativa do item é finalizada**. Caso contrário, segue-se para o próximo passo.

### **Exposição de opiniões**

Se no primeiro momento não há um consenso entre a equipe, é preciso abrir uma discussão entre o grupo para que as opiniões sejam expostas. Nessa etapa, **integrantes que indicaram o maior e o menor valor devem defender seus pontos de vista, explicando o motivo pelo qual escolheram as cartas em questão**.

Com essa nova reflexão, a equipe volta à fase anterior para que cada pessoas faça uma nova escolha de estimativa. Assim, as cartas são mostradas novamente e, se foram todas iguais, a estimativa é encerrada. Se não, a equipe segue para a próxima etapa.

### **Decisão baseada no consenso**

Em geral, após a primeira rodada, a tendência é que as escolhas dos valores das cartas sejam mais próximos uns dos outros, o que facilita a estimativa. Por isso, nesse momento, a pessoa que assume a função de Scrum Master pode fazer uma escolha entre:

* **escolher o maior valor**, caso sejam próximos o suficiente;
* **calcular a média** entre as estimativas de cada carta mostrada;
* ou **fazer novas rodadas de estimativas individuais** até que a equipe toda escolha a mesma carta.

É importante lembrar que o Planning Poker valoriza o consenso, sendo interessante envolver e analisar a opinião da equipe para fazer essa escolha.

Agora você já sabe o que é Planning Poker e como essa ferramenta pode ser usada para auxiliar no processo de criação de um software. Essa é uma técnica que certamente vai ajudar a [garantir o sucesso](https://blog.betrybe.com/carreira/guia-carreira-de-sucesso/) do seu projeto, por isso vale a pena conhecê-la bem antes de colocá-la em prática!