

Metodologias Ágeis

Framework Ágeis

Scrum

Kanban

Git e Github

Profª. Heloisa Moura

CONTEÚDO DA AULA

- Introduzir conceitos de metodologias ágeis
- Framework Scrum e Kanban
- Git e Github

Metodologia tradicional/Metodologias Ágeis

Metodologia tradicional

Como funciona a Metodologia tradicional:

- A empresa pensa e desenha como será o projeto;
- Planeja todas as atividades, como designer, programação e comunicação;
- Executa todas as tarefas;
- Testa;
- Implanta o projeto pronto.



Metodologia tradicional/Metodologias Ágeis

Metodologia tradicional

E para ter sucesso nessa metodologia, não basta só o projeto está pronto, é preciso entregar no prazo, dentro do orçamento, com a qualidade esperada. Atrasos e estouro de custos são bem comuns e entendidos. A qualidade do produto pronto pode ser duas coisas:

- Não ter erros e funcionar bem;
- Ter todas as funcionalidades que imaginou no início.

Podemos perceber que na metodologia tradicional é esperado que se execute o que planejou, dentro do prazo e do custo. Por isso que mudanças aqui no decorrer dos ciclos é vista com muita preocupação pelos Gerentes de projeto.

Metodologia tradicional/Metodologias Agéis

Metodologia tradicional

Exemplo de um projeto que a utilização da metodologia tradicional é comum é a construção de prédios e casas, onde não se pode ter mudanças no meio da construção.

Agora pense nas empresas de tecnologia ou qualquer outra empresa que tenha projetos complexos, que muitas vezes precisam de mudanças no decorrer dos ciclos.

Isso é uma características dessas empresas, que precisam estar sempre atentas as necessidades dos clientes que mudam quase diariamente, a tecnologia que muda direto e os concorrentes bem mais agressivos.

- 1 Clientes
- 2 Tecnologia
- 3 Concorrentes

Metodologia tradicional/Metodologias Ágeis

Metodologia tradicional

Com esse cenário, não se pode planejar um projeto com início por exemplo em janeiro e término em agosto, sem se preocupar com tudo que aconteceu no decorrer desse período. Então imagina!!!

A empresa precisa ter liberdade para mexer nos ciclos do projeto, sem causar transtornos em prazos, custos e qualidade.

COMO FAZER???

Metodologia tradicional/Metodologias Ágeis

Metodologias Ágeis

Ai surgiu a metodologia Ágil, que se baseia em 4 valores e 12 princípios, que surgiram através de uma reunião que aconteceu em 2001 entre 17 representantes . E deram o nome de manifesto ágil. Esses valores e princípios podem ser encontrados em: agilemanifesto.org

Nessa metodologia agilidade não vem de rapidez e sim de quebrar o produto final em pedaços menores. Ao invés de:

- Desenhar;
- Planejar;
- Executar tudo no detalhe como na metodologia tradicional.

Metodologia tradicional/Metodologias Ágeis

Metodologias Ágeis

Na Metodologia Ágil o projeto começa com uma visão bem ampla, e desenha de uma forma mais genérica o produto, quebrando em versões.

Imagina que a empresa quer implantar o produto final daqui a 6 meses. Então lança uma versão inicial, com o mínimo de funcionalidades para testar junto aos usuários. E essa versão pode estar pronta daqui a 2 meses. Uma segunda versão com correções e melhorias em 4 meses e a versão final com tudo que o produto precisa no tempo final de 6 meses.

Note que por causa da quebra em versões se pode fazer mudanças ao longo do projeto e o que importa é que o planejado, executado em detalhe é a versão e não o projeto todo. As preocupações com prazos, custos e qualidade são as mesmas. Mas se consegue controlar mais de perto e gerenciar as mudanças durante o projeto.

Metodologia tradicional/Metodologias Ágeis

Metodologias Ágeis

E daí a opção das empresas de tecnologia utilizarem a metodologia ágil ou combinar as duas. Extraindo o melhor de cada uma. Dependendo da complexidade do projeto em questão.

E dentro da metodologia ágil, existe os métodos ágeis, que são frameworks que ajudam as empresas a implantarem o manifesto ágil em seus projetos.

Daí a pergunta!!!

Mas não seria só seguir os 4 valores e os 12 princípios do manifesto?

Não dessa forma, pois não conseguiríamos colocar em prática, e ter agilidade no dia a dia. E os frameworks são mais específicos, ou seja, vão definir alguns papéis, eventos, artefatos mais tangíveis e algumas regras.

Frameworks

Scrum e kanban

Scrum

O que é o Scrum:

É um framework para organizar e gerenciar trabalhos complexos, tal como projetos de desenvolvimento de software. É um conjunto de valores, princípios e práticas que fornecem a base para que a empresa adicione suas práticas particulares de gestão e que sejam relevantes para a realidade da empresa.

Como funciona o Scrum

Video explicativo

Frameworks

Scrum e kanban

Kanban

O que é o Kanban:

É um framework de gestão ágil que auxilia no controle do projeto, tarefas, demandas, etc. É praticamente um mapa daquilo que está acontecendo na empresa, ou equipe. Ao bater o olho no quadro Kanban, se enxerga a evolução das tarefas, em que ponto estão as coisas. É um quadro que representa todo o trabalho que está em andamento e se todo mundo colaborar, estará sempre atualizado e facilitará a vida de todos.

O método permite a melhoria no fluxo de tarefas, deixa o projeto mais visual, colaborativo, facilita a identificação de problemas, deixa o time mais unido, o trabalho mais fluído e como consequência faz com que o produto final tenha maior qualidade.

Frameworks

Scrum e kanban

Kanban

UMA BREVE HISTÓRIA DO KANBAN

O Kanban nasceu dentro da Toyota tendo Taiichi Ohno como seu criador, seu significado é “cartão visual”. A ideia na época era fazer o controle de estoque, de produção, algo mais voltado para a manufatura. Mais tarde o método foi popularizado pelo David Anderson para área de Desenvolvimento de Software.

O Kanban apesar de ter sido criado pela Toyota lá nos anos 60, é um dos métodos mais modernos, práticos e ágeis para a empresa organizar, gerenciar e liderar os seus projetos.

E PODE SER USADO PARA QUALQUER TIPO DE PROJETO!

Frameworks

Scrum e kanban

Kanban

Como funciona o kanban:

- O Kanban cabe em qualquer área, qualquer tipo de projeto e equipe;
- Se você está fazendo algum trabalho que tenha um passo a passo e uma coisa no fim, como a fabricação de pães, montagem de lanches, projetos de marketing, qualquer coisa que tenha início meio e fim pode usar kanban!
- A principal ideia do Kanban é entender o fluxo de trabalho;
- Parta do que você tem hoje, olhe como funciona o passo a passo do trabalho do seu time. Mapeie por quais fases a sua demanda passa até que ela finalize.

Frameworks

Scrum e kanban

Kanban

Como funciona o kanban:

- Pegue um quadro, ou mesmo a parede, escreva em uma ponta “para fazer” e na outra “concluído” e preencha o meio entre essas fases com o que acontece para que a demanda saia do primeiro e chegue no último.
- Chame a equipe! Às vezes você sozinho não irá enxergar todas as fases de trabalho.
- Entenda esse fluxo junto com a equipe, afinal, nem sempre todos estão cientes do que acontece em todas as etapas, isso será um ganho enorme para sua equipe.

Frameworks

Scrum e kanban

Kanban

- **Como começar**
- Se a empresa usa Scrum por exemplo, é só pegar as prioridades do backlog e colocar na primeira coluna do quadro Kanban, que pode ser o “a fazer”.
- Depois comece puxando trabalho semanalmente e vá analisando se isso faz sentido pra equipe, até achar o *time* que combina com o processo de trabalho.
- Pode inclusive trabalhar com os dois ao mesmo tempo! Nesse pode por exemplo usar o Kanban durante a daily, a sprint planning irá alimentar o quadro... Scrum e Kanban funcionam muito bem juntos!

Frameworks

Scrum e kanban

Kanban

- Como começa
- Se a empresa usar o quadro Kanban
- Depois começa o *time* que com
- Pode inclusive ter a daily, a sprint



r na primeira coluna

pra equipe, até achar

sar o Kanban durante
o bem juntos!

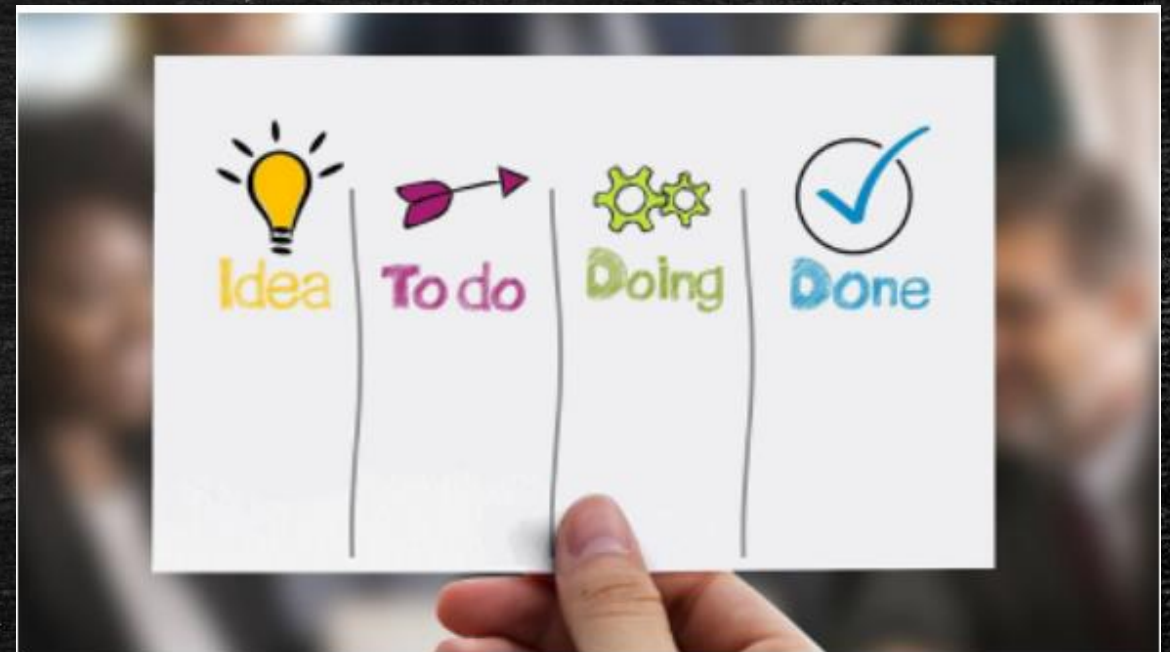
Frameworks

Scrum e kanban

Kanban

Os passos do Kanban:

- Visualize seu fluxo
- Limite o trabalho
- Meça e gerencie seu fluxo
- Tenha mecanismos de feedback
- Tenha liberdade de feedback
- Mantenha as políticas explícitas!



Git e Github

Git

O que é o Git:

É um sistema de controle de versão open-source, ou seja, gratuito. Ele é **utilizado para a criação de um histórico de alterações em código-fonte de projetos de desenvolvimento de software**. Foi desenvolvido por Linus Torvalds, o criador do sistema operacional Linux. Ajuda no gerenciamento e organização dos códigos-fonte.

A importância maior de se trabalhar com um sistema de controle de versões é poder trabalhar em equipe, e essa equipe poder compartilhar as versões dos códigos entre si.

Por meio de sua utilização, podemos saber quais foram as alterações realizadas, quem fez cada uma das alterações e baixar essas mudanças em nossa máquina. Se necessário, revertê-las para uma versão anterior.

O Git é um sistema de controle de versão distribuído. Isso que dizer que a cópia de cada versão que está sendo desenvolvida ou foi desenvolvida com o passar do tempo estará em todas as estações de trabalho que estiverem trabalhando naquele sistema e não só no repositório. Isso permite uma velocidade maior de trabalho, porque você, não precisa ficar acessando muito seu repositório.

Git e Github

Git

São vários conceitos dentro do Git que nos permitem manipular o código-fonte dentro de suas inúmeras versões.

Os principais deles são:

- ***Repositório***

Local físico onde os arquivos e suas cópias ficarão armazenados. O repositório pode ser local ou remoto, podendo salvar não apenas arquivos de texto, mas também imagens, áudios e outros elementos relacionados ao projeto.

- ***Branch***

Branches são os ramos, cópias do código original que podem ser manipuladas de forma livre pela pessoa que trabalha em programação, sem afetar as funcionalidades em produção no código-fonte. Isso permite que todas as alterações sejam realizadas de forma segura, sem que erros ocorram na cópia principal do projeto.

Git e Github

Git

- ***Merge***

Após a finalização de um trabalho em um Branch, é necessário realizar o Merge, fundir a cópia e seus arquivos modificados com o ramo principal do projeto. Isso acontece apenas no repositório local para que ajustes possam ser feitos.

- ***Push Request***

O Push Request é o envio das modificações após o Merge para o repositório central, para que todas as outras pessoas que atuam no desenvolvimento possam atualizar suas cópias e revisar o código criado, verificando conflitos com seus próprios trabalhos.

Git e Github

Git

- ***Pull Request***

O Pull Request é utilizado quando outra pessoas que atua no desenvolvimento muda o ramo principal no repositório central, puxando as modificações realizadas para a sua máquina, fundindo a nova versão com o seu código local.

- ***Fork***

Trata-se do comando de cópia de um repositório remoto para a máquina local, realizado sempre que vamos começar a trabalhar em um projeto que já existe. Também é usado para pegar um código público para posterior modificação e utilização em um programa interno

Git e Github

Github

O que é o Github:

É uma plataforma totalmente online onde você pode criar repositórios e hospedar neles seus projetos, colaborar com softwares open source, seguir outros(as) programadores(as) e interagir com códigos de terceiros. Para criar uma conta no Github acesse:

<https://github.com/join>