

AULA 1 – CONCEITOS ESSENCIAIS DE DESENVOLVIMENTO ÁGIL

OBJETIVO DA AULA

Conhecer o modelo de desenvolvimento ágil e os conceitos essenciais que fazem parte desta filosofia de desenvolvimento de software.

APRESENTAÇÃO

O desenvolvimento ágil de software tornou-se uma prática muito popular há alguns anos, isso porque alguns modelos clássicos ou que estavam em voga na época demoravam muito para entregar um produto executável para o cliente.

Não era incomum alguns projetos, mesmo já iniciados há alguns meses, não terem ainda um protótipo executável, e ao invés disso, apenas pilhas de documentação e diagramas.

Com a velocidade como as coisas mudam na atualidade e grande possibilidade de mudança de prioridades e requisitos, os processos fracassavam muitas vezes por diversos motivos oriundos do modelo de desenvolvimento escolhido.

Nesta aula vamos abordar os principais conceitos da filosofia de desenvolvimento ágil.

1. MODELOS ÁGEIS

Há muito tempo já era questionado o processo de desenvolvimento clássico de software, pois eles eram normalmente longos e inflexíveis. Havia uma demora grande entre o momento em que o projeto era definido e o cliente finalmente poder testar e usar o seu produto.

Esses questionamentos iniciaram já nos anos 1980, mas foi apenas no final dos anos 1990 que a ideia tomou forma com o desenvolvimento da noção de abordagens ágeis como a Metodologia de Desenvolvimento de Sistemas Dinâmicos (DSDM), cunhado em 1997, a *Extreme Programming* (XP) de 1999 e o *Scrum* em 2001.

O modelo de desenvolvimento ágil é uma filosofia cujo objetivo é produzir mais rapidamente projetos úteis, isto é, elaborar produtos de maneira mais rápida, realizando entregas constantes e executáveis para os clientes.

Essas entregas são incrementos que trazem novas funcionalidades a cada entrega, trazendo com isso uma série de benefícios tanto para o cliente quanto para a equipe de desenvolvimento:

- **Maior interação desenvolvedor-cliente:** como são realizadas várias pequenas entregas, ao invés de uma grande entrega única, o processo de validação e aceite ocorre de maneira constante, o que reduz o risco de falhas;

- Priorização definida pelo cliente: como os requisitos são entregues a cada nova entrega, o cliente pode negociar com a equipe de desenvolvimento qual é ou quais são as prioridades de entrega para a nova versão;
- Flexibilidade na mudança de requisitos: esta filosofia faz com que seja muito mais rápida e menos impactante a mudança de determinado requisito.

Com a absurda velocidade das mudanças nos negócios, impulsionadas pela alta competitividade e o suporte indispensável da tecnologia da informação, o conceito de desenvolvimento ágil trouxe diversos benefícios, como a rapidez no processo como um todo e a qualidade do produto final.

Há a necessidade, porém, de um ambiente que encoraje esse tipo de processo, além de lideranças que apoiem essa filosofia. O ambiente deve ser organizado, inspecionado constantemente e munido de objetos que facilitem a prática desses modelos, como quadros e canetas *pilot*, para que o andamento do projeto e prazos sejam percebidos por toda a equipe.

Conheça a comunidade de desenvolvimento ágil no Brasil. Disponível em: www.desenvolvimentoagil.com.br. Acesso em: 15 nov. 2022.

LINK



Os modelos ágeis normalmente utilizam como base o modelo iterativo e incremental, além de incluir no processo de desenvolvimento todos os interessados no projeto, utilizando reuniões diárias e semanais de acompanhamento do desenvolvimento do produto.

Esse tipo de modelo de desenvolvimento considera algumas características que compartilham entre si.

- Os processos de especificação, projeto e implementação são intercalados;
- A documentação do projeto é minimizada, trazendo apenas aquilo que for considerado fundamental pela equipe;
- Prefere-se a comunicação com o cliente do que a elaboração de documentação;
- São geradas diversas versões do mesmo sistema, cada uma trazendo um ou mais incrementos prontos para serem testados e usados;
- Há a participação ativa de todos os *stakeholders* do projeto;
- Os próprios usuários validam cada nova versão do sistema;

Cada novo ciclo dura, em média, entre duas e três semanas.

Após toda a discussão sobre a efetividade dos modelos clássicos, abarrotados de documentos que, muitas vezes, nem seriam utilizados, foi instituído o chamado *Manifesto Ágil*, elaborado por desenvolvedores que visavam desburocratizar o processo.

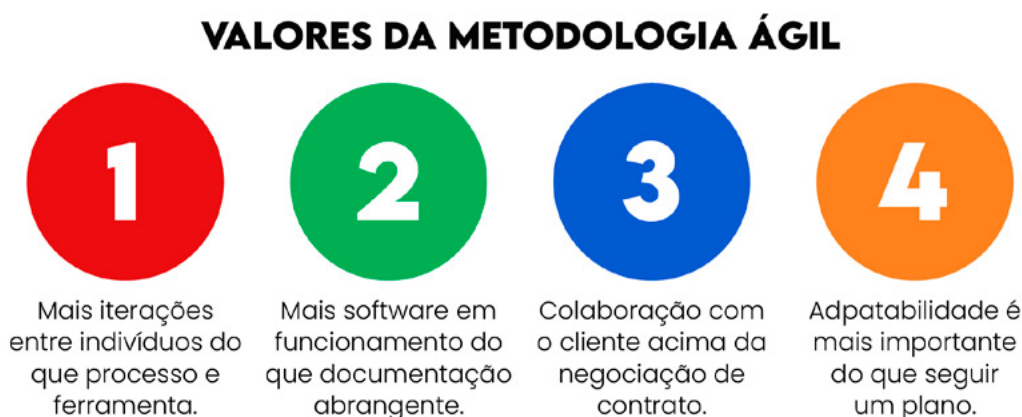
Neste manifesto, uma série de condições foram estabelecidas para que um projeto de software fosse encaixado nessa categoria. Vamos discutir mais profundamente este assunto a seguir, mas basicamente o manifesto ágil afirma que:

Estamos descobrindo maneiras melhores de desenvolver softwares, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Através deste trabalho, passamos a valorizar:

- Indivíduos e interações mais que processos e ferramentas;
- Software em funcionamento mais que documentação abrangente;
- Colaboração com o cliente mais que negociação de contratos; e
- Responder a mudanças mais que seguir um plano.

Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda.

FIGURA 1 | **Valores da Metodologia Ágil**



Elaboração própria.

A Figura 1 mostra de maneira gráfica os valores utilizados nas metodologias ágeis.

2. TIPOS DE METODOLOGIAS ÁGEIS

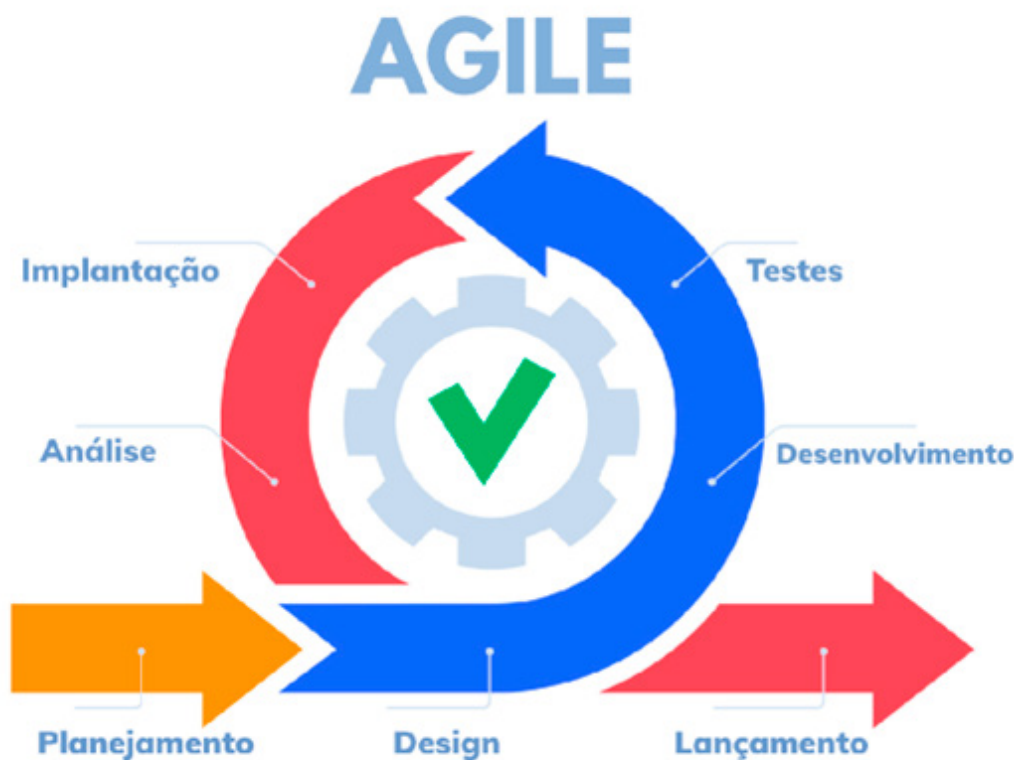
Dentre os vários tipos de modelos temos como mais populares os modelos *Extreme Programming* (XP) e o *Scrum*. Mas existem também outros modelos considerados como ágeis:

- FDD – Desenvolvimento dirigido à funcionalidade (1997);
- *Dynamic Systems Development Method* (1997);
- Desenvolvimento de Software Adaptativo (2000);
- *Crystal* (2001); e

- Desenvolvimento Dirigido a Características (2002)

Devido ao sucesso deste movimento, alguns modelos já existentes acabaram criando adaptações de seus modelos para a metodologia ágil, como foi o caso do modelo RUP (*Rational Unified Process*), que criou instâncias ágeis de seu modelo.

FIGURA 2 | Ciclo de Vida Básico das Metodologias Ágeis



Fonte: blog.cronapp.io/governanca-de-dados/ – Acesso em: 14 nov. 2022.

A Figura 2 mostra um exemplo do ciclo de vida básico dos métodos de desenvolvimento ágeis.

Os métodos ágeis são muito utilizados atualmente e têm sido bem-sucedidos principalmente para desenvolvimento de produtos de complexidade e tamanhos pequenos ou médios. No entanto, esse sucesso depende também de um amplo envolvimento e comprometimento do cliente com todo o processo de desenvolvimento.

CONSIDERAÇÕES FINAIS

Nesta aula tratamos da conceituação dos modelos ágeis, apresentando o cenário de desenvolvimento que motivou aos desenvolvedores a propor este tipo de filosofia, publicando logo a seguir o chamado manifesto ágil, que determinava quais pensamentos conduziria a abordagem.

Discutimos as características principais de um modelo ágil, que normalmente é compartilhado entre os modelos mais conhecidos no mercado como o *Extreme Programming* e o *Scrum*.

As metodologias ágeis são alternativas ao modelo tradicional de desenvolvimento de software e são utilizadas com o objetivo de agilizar o trabalho dos desenvolvedores, minimizando a elaboração de documentação desnecessária, e gerando melhoria contínua para os processos que envolvem a produção de software.

Vale a pena destacar que, embora seja muito utilizada em projetos de software em todo o mundo, esta forma de desenvolver softwares não inviabiliza outros modelos e metodologias de desenvolvimento.

MATERIAIS COMPLEMENTARES

Vídeo: Decifrando o Agile 1 – Por que usar métodos ágeis? Disponível em: https://www.youtube.com/watch?v=ds_Fydzsu08. Acesso em: 14 nov. 2022.

Vídeo: Decifrando o Agile 2 – Entenda o que são os métodos ágeis em 5 minutos. Disponível em: <https://www.youtube.com/watch?v=efZlpew90Nk>. Acesso em: 14 nov. 2022.

Link: Modelos Ágeis: Conceitos e Princípios. Disponível em: <https://www.devmedia.com.br/modelos-de-processos-ageis-conceitos-e-principios/30059>. Acesso em: 14 nov. 2022.

REFERÊNCIAS

KENNETH C. LAUDON; JANE P. LAUDON. *Sistemas de informação gerenciais*. 11ª edição. Editora Pearson, 2014.

PRESSMAN, R. G. *Engenharia de Software*. 9ª ed. Rio de Janeiro: McGraw-Hill, 2021.

SOMMERVILLE, I. *Engenharia de Software*. 10ª ed. São Paulo: Pearson Addison. Wesley, 2019.