

Engenharia de Software

José Cascais Brás

Tópico 2



UNIVERSIDADE
LUSÓFONA

Engenharia de Software

AGENDA

1

AGILE MANIFESTO

2

METODOS AGILE

3

TÉCNICAS DE DESENVOLVIMENTO AGILE

4

GESTÃO DE PROJETOS AGILE

5

ESCALONAMENTO DE PROJETOS EM AGILE (SAFe)

AGILE

AGILE MANIFESTO

Contexto

O Agile Manifesto foi um documento criado em 2001 por um conjunto de 17 signatários que estabelece os valores e princípios fundamentais das metodologias ágeis.

Valores

O Agile Manifesto estabelece 4 valores orientadores

Princípios

O Agile Manifesto estabelece ainda 12 princípios da metodologia agile.

AGILE

VALORES DO MANIFESTO AGILE

O Agile Manifesto estabelece quatro valores orientadores:

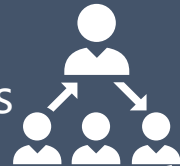
Os indivíduos e suas interações são mais importantes que os procedimentos e ferramentas;



Software a funcionar é mais importante que ter a documentação completa;



A colaboração com o cliente é mais importante que a negociação de contratos



A capacidade de responder a alterações é mais importante que seguir um plano



Estes valores não pretendem negar o valor dos procedimentos, ferramentas, documentação, contratos e planos.

Em vez disso, pretendem estabelecer uma escala de prioridades que introduza flexibilidade nos projetos.

AGILE

AGILE MANIFESTO

O AGILE MANIFESTO ESTABELECE 12 PRINCÍPIOS DA METODOLOGIA AGILE

1	Garantir a satisfação do cliente, através da entrega rápida e continua de software funcional	Software funcional é a principal medida de progresso do projeto	7
2	Até mesmo mudanças tardias ao âmbito do projeto são bem-vindas	Os processos ágeis promovem o desenvolvimento sustentável. Os patrocinadores, programadores e utilizadores devem ser capazes de manter um ritmo constante indefinidamente	8
3	Software funcional é entregue frequentemente	A atenção contínua à excelência técnica e ao bom design aumenta a agilidade	9
4	Cooperação constante entre as pessoas que entendem do 'negócio' e os programadores	Simplicidade – a arte de maximizar a quantidade de trabalho não feito – é essencial	10
5	Os projetos devem ser criados em torno de indivíduos motivados. Dê-lhes o ambiente e o apoio que necessitam, e confie-os para começar o trabalho feito	As melhores arquiteturas, requisitos e projetos emergem de equipas auto-organizadas	11
6	O método mais eficiente e eficaz de transmitir informações para e dentro de uma equipa de desenvolvimento é a conversa cara-a-cara	Em intervalos regulares, a equipa reflete sobre como tornar-se mais eficaz, ajustando o seu comportamento em conformidade	12

MÉTODOS AGILE

Antes

Insatisfação com as despesas gerais relacionadas com desenvolvimento de software métodos de design das décadas de 1980 e 1990 levaram à criação de métodos ágeis.



Agora

- › Concentração no código em vez do design
- › Abordagem iterativa para o desenvolvimento de software
- › Orientado à entrega de software funcional rapidamente e evoluir
- › Permite rapidamente atender a requisitos em constante mudança.

Um dos objetivos dos métodos ágeis é reduzir despesas gerais no processo de desenvolvimento de software (por exemplo, limitando a documentação) e ser capaz de responder rapidamente às mudanças de requisitos sem retrabalho excessivo.

AGILE

Estamos a descobrir melhores maneiras de desenvolver software, ao fazer isso e ajudamos outros a fazê-lo.

Por meio desta forma de trabalhar, conseguimos valorizar:

Responder à mudança em vez de seguir um plano

Colaboração do cliente em vez de negociação do contrato



Indivíduos e interações em vez de processos e ferramentas

Software que trabalha em vez de documentação completa.

AGILE

PRINCIPIO	DESCRIÇÃO
Envolvimento do cliente	Os clientes devem estar intimamente envolvidos em todo o processo de desenvolvimento. Sua função é fornecer e priorizar novos requisitos de sistema e avaliar as iterações do sistema.
Entregas incrementais	O software é desenvolvido em incrementos com o cliente especificando os requisitos a serem incluídos em cada incremento
Pessoas não processos	As habilidades da equipa de desenvolvimento devem ser reconhecidas e exploradas. Aos elementos da equipa deve ser permitido desenvolver as suas próprias maneiras de trabalhar sem processos prescritivos.
Abraçar a mudança	Esperar que os requisitos do sistema mudem e, assim, projetar o sistema para acomodar essas mudanças
Manter a simplicidade	Foco na simplicidade tanto do software a ser desenvolvido e do processo de desenvolvimento. Sempre que possível, trabalhar ativamente para eliminar a complexidade do sistema.

ORGANIZAÇÕES AGILE

“É necessário um tipo especial de liderança para conduzir uma empresa por um meio de incerteza e disfunções rotineiras. Se queremos deseja liderar um negócio próspero, devemos ser capazes de encarar um desafio e saber que o peso não está apenas nos ombros de uma pessoa. Precisamos de agilidade sustentável. É sobre ser capaz de confiar que as equipes da nossa empresa têm o que é preciso para assumirmos a liderança e nos adaptarmos rapidamente para entregar o que seus clientes precisam.”

Source: <https://www.scrumalliance.org/agile-organizations>

AGILE

Strategic Leaders Must Be Consistent and Agile at the Same Time



FROM "THE BEST STRATEGIC LEADERS BALANCE AGILITY AND CONSISTENCY," BY JOHN COLEMAN, JANUARY 2017

© HBR.ORG

AGILE

TRABALHO BEM DEFINIDO VS. TRABALHO DE ALTA INCERTEZA



- › O trabalho do projeto varia de trabalho definível a trabalho de alta incerteza.
- › Projetos de trabalho definíveis são caracterizados por procedimentos claros que tiveram sucesso em projetos semelhantes no passado.
- › A produção de um carro, aparelho elétrico ou casa após a conclusão do projeto são exemplos de trabalho definível.
- › O domínio de produção e os processos envolvidos são geralmente bem compreendidos e normalmente existem baixos níveis de incerteza e risco de execução.

AGILE

TRABALHO BEM DEFINIDO VS. TRABALHO DE ALTA INCERTEZA



- › Um novo design, resolução de problemas e trabalho não-feito-antes, é **exploratório**.
 - › Requer especialistas no assunto para colaborar e resolver problemas para criar uma solução.
- › Exemplos de pessoas que encontram um trabalho de alta incerteza incluem engenheiros de sistemas de software, designers de produtos, médicos, professores, advogados e muitos tipos de engenheiros. À medida que o trabalho mais definível é automatizado, as equipes de projeto vão realizando mais projetos de complexidade elevada que requerem técnicas ágeis.

AGILE

TRABALHO BEM DEFINIDO VS. TRABALHO DE ALTA INCERTEZA



- › Projetos de alta incerteza têm altas taxas de mudança, complexidade e risco. Essas características podem apresentar problemas para as abordagens preditivas tradicionais que visam determinar a maior parte dos requisitos antecipadamente e controlar as mudanças por meio de um processo de solicitação de mudança. Em vez disso, abordagens ágeis foram criadas para explorar a viabilidade em ciclos curtos e se adaptar rapidamente com base em avaliação e feedback.

AGILE

Embora originados na indústria de software, estes princípios espalharam-se por muitas outras indústrias.

A integração de mentalidade, valores e princípios define o que constitui uma chamada “abordagem ágil”.

As várias abordagens ágeis em uso hoje em dia partilham raízes comuns com a mentalidade, valores e princípios Agile.

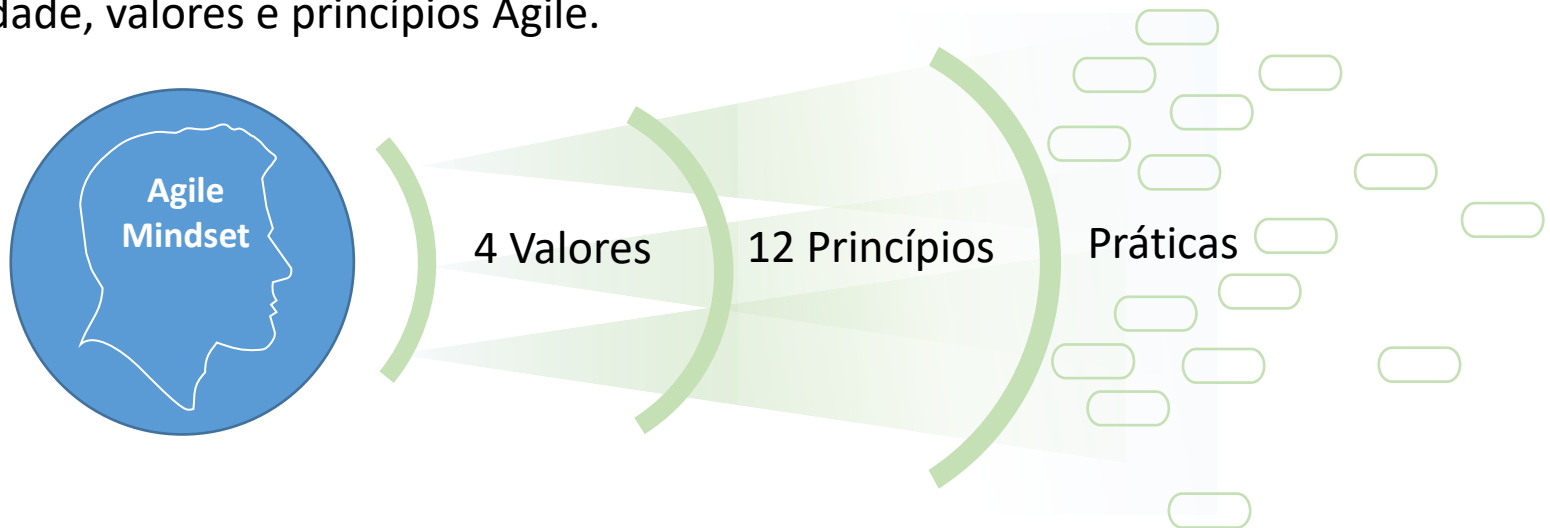


Figura 1: O relacionamento entre os valores, princípios e práticas comuns associadas

AGILE

Conforme mostrado na Figura 1, o modelo, (inspirado por Ahmed Sidky), articula o Agile como uma mentalidade definida pelos valores do Manifesto Agile, guiado pelos princípios do Manifesto Agile e abrangido por várias práticas.



- › É importante salientar que embora o termo “Agile” tenha se popularizado após o Manifesto, as abordagens e técnicas usadas pelas equipas de projeto hoje, existiam antes do Manifesto Agile há muitos anos e, em alguns casos, décadas.
- › Abordagens ágeis e métodos ágeis são termos gerais que cobrem uma variedade de estruturas e métodos. A Figura 1 coloca o Agile no contexto e nela ele é visualizado como um termo genérico, referindo-se a qualquer tipo de abordagem, técnica, estrutura, método ou prática que cumpra os valores e princípios do Manifesto Ágil.

AGILE

Em geral, existem duas estratégias para cumprir os valores e princípios ágeis.



Abordagem ágil formal,

intencionalmente projetada e comprovada para alcançar os resultados desejados.

Depois, reservar tempo para aprender e compreender as abordagens ágeis antes de alterá-las ou adaptá-las. A adaptação prematura e aleatória pode minimizar os efeitos da abordagem e, assim, limitar os benefícios.



Implementar mudanças nas práticas do projeto de uma maneira que se ajuste ao contexto do projeto

para alcançar o progresso em um valor ou princípio fundamental. Usar timeboxes para criar recursos ou técnicas específicas para refinar recursos iterativamente. Considerar-se dividir um grande projeto em várias releases, se isso funcionar para o contexto específico do projeto. Implementar mudanças que ajudem o projeto a ter sucesso - as mudanças não precisam fazer parte das práticas formais da organização. O objetivo final não é ser ágil por si só, mas sim entregar um fluxo contínuo de valor aos clientes e alcançar melhores resultados de negócios.

AGILE

Agile é um termo “guarda-chuva”



- › O termo ágil é um termo abrangente usado para descrever muitas estruturas diferentes. Uma estrutura é “ágil” quando apoia os valores e princípios do Manifesto Ágil.
- › As quatro frameworks ágeis que mais falaremos são Scrum, XP, Kanban e o Scaled Agile Framework (SAFe). Algumas delas são mais consagradas do que outras. Uma estrutura, como XP, tem práticas que são acompanhadas mais de perto, enquanto outra, como Kanban, pode ser mais generalizada.

AGILE

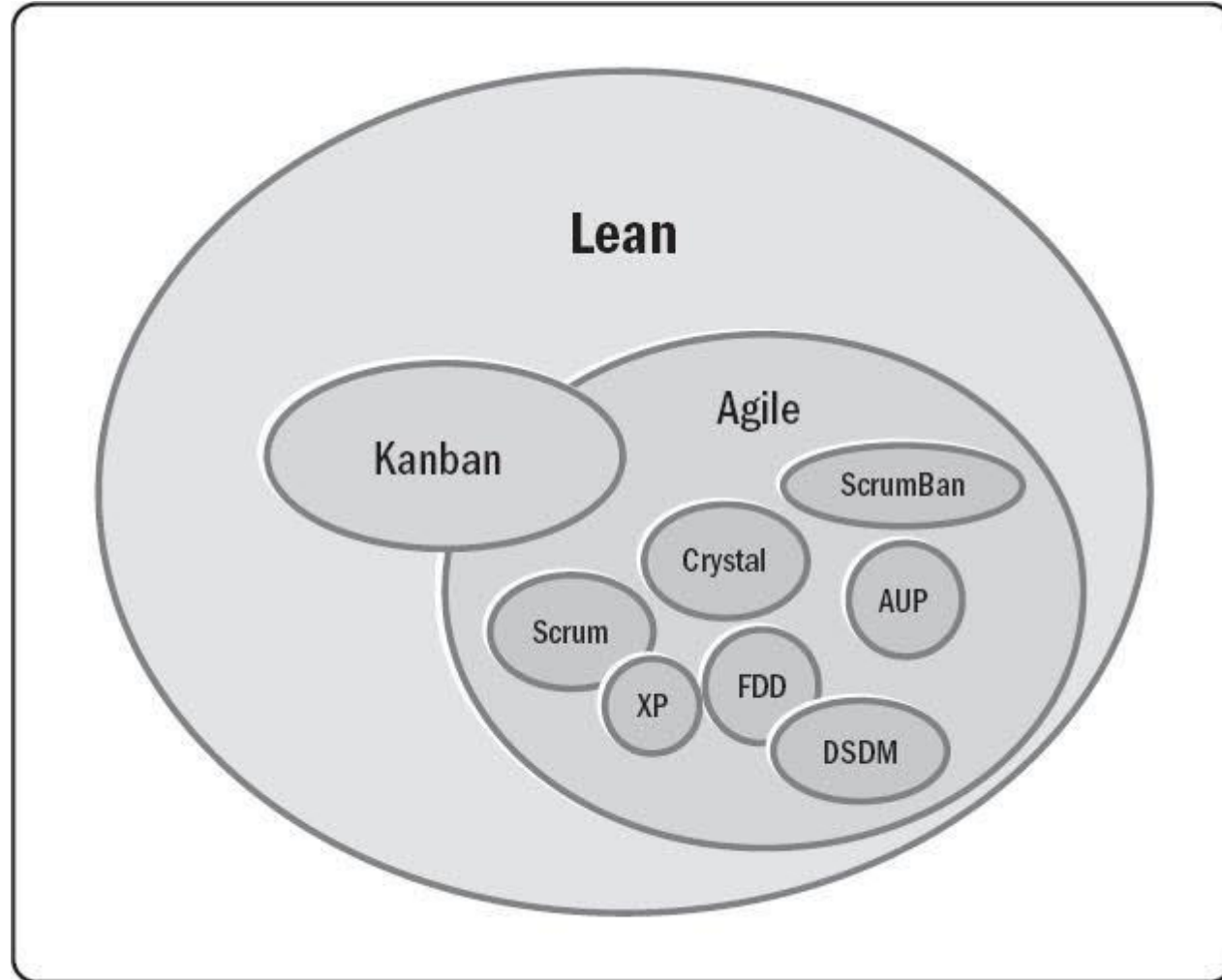


Figura 2: Agile é um termo genérico para muitas abordagens

AGILE

DICA

Muitas das estruturas ágeis também se colam umas às outras. Scrum, em particular, geralmente tem elementos-chave de XP. A maioria dos gestores pode ter dificuldade em distinguir entre Scrum e equipes de XP. Não fica claro simplesmente observando-os trabalhar. Para a maioria das organizações, o Scrum parece atingir o melhor equilíbrio entre estrutura e flexibilidade.

Às vezes, Agile é usado para descrever apenas uma delas. Em outras ocasiões, o termo Agile é usado para todas elas juntas. Podemos ver uma equipe Scrum chamada Agile. É ver-se uma equipe a usar Scrum, Lean ou Kanban e ser chamada de equipe Agile.

AGILE

EXTREME PROGRAMMING

DEFINIÇÃO

“Extreme Programming (XP) é uma estrutura ágil de desenvolvimento de software que visa produzir software de alta qualidade e maior qualidade de vida para a equipa de desenvolvimento. XP é a mais específica das estruturas ágeis em relação às práticas de engenharia apropriadas para o desenvolvimento de software.”

AGILE

EXTREME PROGRAMMING

Quando aplicar

As características gerais onde o XP é apropriado foram descritas por **Don Wells** (www.extremeprogramming.org):

- Requisitos de software que mudam dinamicamente
- Riscos causados por projetos de tempo fixo usando novas tecnologias
- Equipa de desenvolvimento estendida pequena e co localizada
- A tecnologia que você está usando permite testes unitários e funcionais automatizados

Devido à especificidade do XP quando se trata de seu conjunto completo de práticas de engenharia de software, existem várias situações em que podemos não conseguir praticar o XP completamente. O post “When is XP Not Appropriate” no C2 Wiki é provavelmente um bom lugar para começar a encontrar exemplos de onde não usar o XP.

Embora não se possa usar toda a estrutura XP em muitas situações, isso não deve impedir de usar o maior número possível de práticas, considerando seu contexto.

Principais Características

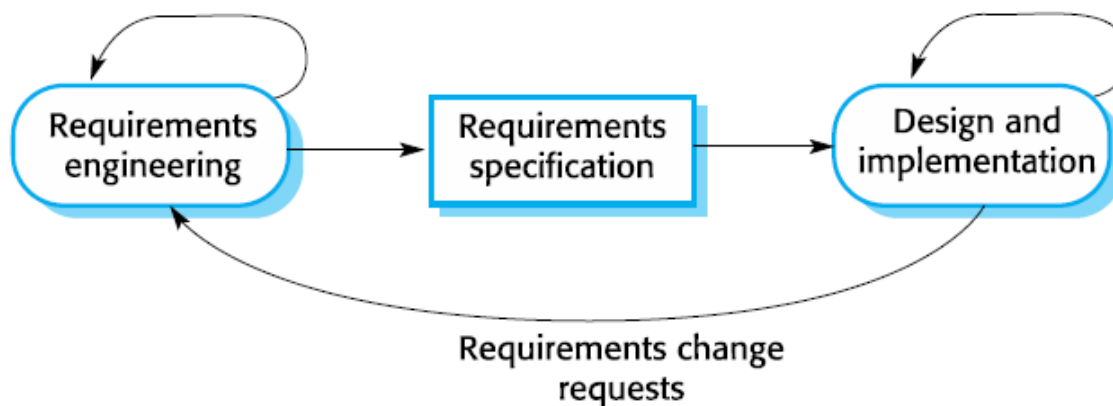


- › A especificação, design e implementação do programa são intercaladas
- › O sistema é desenvolvido ao longo de várias versões ou incrementos em conjunto com os Stakeholders
- › Entregas frequentes de novas versões para avaliação
- › Amplo suporte em ferramentas (por exemplo, ferramentas de teste automatizadas) usadas para apoiar o desenvolvimento.
- › Foco mínimo de documentação – Foco no desenvolvimento de código

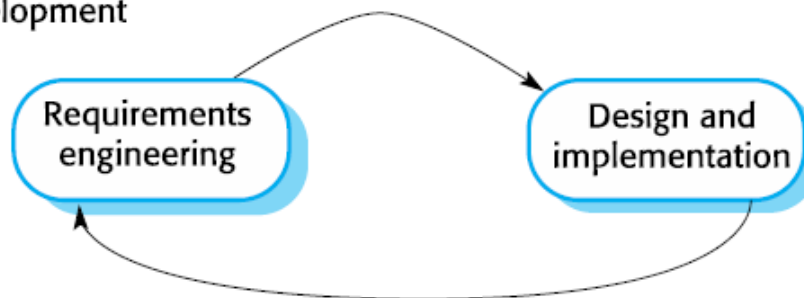
AGILE

DESENVOLVIMENTO AGILE VS ORIENTADO AO PLANO

Plan-based development



Agile development



AGILE

DESENVOLVIMENTO AGILE VS ORIENTADO AO PLANO

Desenvolvimento orientado ao plano

- Uma abordagem orientada a planos para a engenharia de software é baseada em estágios de desenvolvimento separados, com os artefactos a serem produzidos em cada um desses estágios planeados com antecedência.
- Não necessariamente direcionado ao plano do modelo em cascata, o desenvolvimento incremental é possível
- A iteração ocorre dentro das atividades.

Desenvolvimento Agile

- Especificação, design, implementação e teste são intercalados e os artefactos de desenvolvimento são decididos por meio de um processo de negociação durante o processo de desenvolvimento de software.

AGILE

DESENVOLVIMENTO AGILE VS ORIENTADO AO PLANO

TÓPICOS

PRINCÍPIOS ORIENTADOS A PLANOS

PRINCÍPIOS AGILE

Similaridade entre desenvolvimento e fabricação

Ambos seguem um processo definido.

Desenvolvimento não é fabricação; o desenvolvimento cria a receita do produto.

Estrutura do processo

O desenvolvimento é baseado em fases e são sequenciais.

O desenvolvimento deve ser iterativo e incremental.

Grau de variabilidade de processo e produto

Tenta eliminar a variabilidade do processo e do produto.

Aproveitar a variabilidade por meio de inspeção, adaptação e transparência.

Gestão de incertezas

Elimina a incerteza final primeiro, e depois o significado de risco.

Reduzir as incertezas (riscos) em simultâneo.

Tomada de decisão

Tomar cada decisão apenas na fase adequada.

Manter todas as opções em abertos.

Fazer certo da primeira vez

Presume que temos desde logo todas as informações corretas para criar os requisitos e planos.

Evolui-se por tentativa e erro

AGILE

DESENVOLVIMENTO AGILE VS ORIENTADO AO PLANO

TÓPICOS

PRINCÍPIOS ORIENTADOS A PLANOS

PRINCÍPIOS AGILE

Exploração versus exposição

Explorar o que é conhecido atualmente e prever o que não é conhecido.

Favorece uma abordagem adaptativa e exploratória.

Mudanças / emergências

Mudanças são prejudiciais aos planos e caras, devem ser evitadas.

Abraça a mudança de uma forma economicamente sensata.

Preditivo versus adaptativo

O processo é altamente preditivo.

Equilibra o trabalho antecipado preditivo com o trabalho just-in-time adaptável.

Suposições (conhecimento não validado)

O processo é tolerante com suposições duradouras.

Valida suposições importantes rapidamente.

Feedback

A aprendizagem ocorre após grande ciclo de análise-projeto-código-teste.

Estimula as Lesson Learned dos vários loops

Feedback rápido

O processo é tolerante com o feedback tardio.

Organiza os fluxos de trabalho para obter feedback rápido.

AGILE

DESENVOLVIMENTO AGILE VS ORIENTADO AO PLANO

TÓPICOS

PRINCÍPIOS ORIENTADOS A PLANOS

PRINCÍPIOS AGILE

Batch size (quanto trabalho é concluído antes que a próxima atividade possa começar)

São grandes, 100% - todos em simultâneo. Economias de escala devem ser aplicadas.

Usa tamanhos de batches menores e economicamente viáveis.

Pessoas versus desperdício de trabalho

Alocar pessoas para alcançar altos níveis de utilização.

Concentra-se no trabalho inativo, não nos trabalhadores inativos.

Foco

Centrado no processo - siga o processo.

Centrado no valor - entregue o valor.

Custo do atraso

O custo do atraso raramente é considerado.

Sempre considera o custo do atraso.

Conformidade com o plano

A conformidade é considerada o meio principal de alcançar um bom resultado.

Adapta e replaneia em vez de seguir um plano.

Progresso

Demonstra o progresso progredindo por estágios ou fases.

Mede o progresso validando ativos de trabalho.

AGILE

DESENVOLVIMENTO AGILE VS ORIENTADO AO PLANO

TÓPICOS

PRINCÍPIOS ORIENTADOS A PLANOS

PRINCÍPIOS AGILE

Rapidez

Acompanhe o processo; faça as coisas certas da primeira vez e vá rápido.

Ir rápido, mas nunca com pressa.

Quando obtemos alta qualidade

A qualidade vem no final, após uma extensa fase de teste e conserto.

Construir qualidade desde o início.

Formalidade (cerimônia)

A formalidade (procedimentos e pontos de verificação bem definidos) é importante para uma execução eficaz.

Usar formalidade minimamente suficiente.

SAFe

SAFe

(Scaled Agile Framework)

SAFe

SAFe®



- › O SAFe foi projetado para ajudar as empresas a entregar valor de forma contínua e mais eficiente de uma forma regular e previsível.
- › Ele fornece uma base de conhecimento de princípios e práticas comprovadas e integradas para apoiar a agilidade empresarial.

SAFe

SAFe®



- › Uma chave importante para o sucesso no apoio a uma transformação Lean-Agile é o envolvimento da liderança combinado com educação e treino. O currículo baseado de funções do Scaled Agile consta ajudar as empresas a desbloquear resultados de negócios com o SAFe.
- › Foi projetado para ajudar as empresas a vincular a estratégia à execução, treinando líderes técnicos e de negócios, arquitetos e programadores em práticas Lean-Agile.

SAFe



SAFe®

(Scaled Agile Framework®)

Capacita organizações complexas
a alcançar os benefícios do
software Lean-Agile e no
desenvolvimento de sistemas em
escala.

O desenvolvimento de software ágil tem se tornado cada vez mais popular à medida que empresas de software e não software fazem a transição de metodologias de desenvolvimento tradicionais, como o modelo em cascata, para uma abordagem Agile orientada por valor.

A base Agile

Agile é um termo abrangente para vários métodos e ferramentas de desenvolvimento de software, como Scrum e Scaled Agile Framework (SAFe), que partilham um sistema de valores comum.

Desenvolvido em 2001, o Manifesto Agile fornece um conjunto de princípios fundamentais que as equipes Agile e seus líderes adotam para desenvolver software com sucesso e agilidade.

As empresas que adotaram práticas de desenvolvimento Agile reconhecem a urgência de se adaptar rapidamente às mudanças de tecnologia e entregar software de qualidade num curto espaço de tempo; caso contrário, correm o risco de se extinguir.

SCALING AGILE METHODS

- › Os métodos ágeis provaram ser bem-sucedidos para projetos de pequeno e médio porte que podem ser desenvolvidos por uma pequena equipa local.



- › Por vezes, argumenta-se que o sucesso desses métodos vem devido à melhoria das comunicações, que é possível quando todos estão a trabalhar juntos.



- › Aumentar a escala de métodos ágeis envolve alterá-los para lidar com projetos maiores, mais longos e mais complexos, onde existem várias equipas de desenvolvimento, talvez a trabalhar em locais muito diferentes.



SAFe

Muitas organizações grandes executam vários grandes projetos em simultâneo. É por isso que é muito comum que estas organizações terem um grupo de projetos a que chamam portfólio de projetos. Este portfólio de projetos é então subdividido em “programas”.



- › SAFe é uma combinação de pensamento Lean, Agile ao nível de sistemas. Isso dá ao SAFe uma vantagem considerável para transformações *ágeis*. A maioria das organizações já possui departamentos e processos ao nível de sistemas.
- › O SAFe divide um projeto em três níveis: um nível de **equipa**, um nível de **programa** e um nível de **portfólio**. O SAFe começa com uma estrutura no estilo Scrum para o nível de equipa. O nível de equipa também pode usar estruturas híbridas, como ScrumBan ou AgileFall. Eles podem até usar a framework Lean como Kanban.

Welcome to
SAFe®
for Lean Enterprises

SAFe®

Overview in 5 Minutes

SCALED AGILE®
Provider of SAFe

<https://youtu.be/aW2m-BtCJyE>

Q&A

