

Engenharia de
Software



Anhanguera

AVALIE
SUA PROFISSÃO

QUANDO APARECER EM SEU
PORTAL UMA AVALIAÇÃO SOBRE
SEU CURSO, RESPONDA:



NOTAS

9 ou 10

SIGNIFICA QUE VOCÊ INDICA

NOTAS

7 ou 8

SIGNIFICA QUE VOCÊ NÃO INDICA



Anhanguera



Anhanguera



MODELO ÁGIL PARA A GESTÃO DE PROJETOS DE SOFTWARE

As metodologias ágeis são conjuntos de práticas para gerenciar projetos mais adaptável às mudanças. Elas são estruturadas em pequenos ciclos de trabalho, que geram entregas, essas entregas acrescentam um incremento ao produto final, assim, a cada novo ciclo, é entregue um conjunto de funcionalidades no produto.



Anhanguera

CONCEITO-CHAVE

Aquilo que não é bom não dura por muito tempo. Se a seleção natural é infalível na natureza, ela parece atuar com rigor semelhante no mundo do desenvolvimento de software ao permitir que apenas as metodologias mais aptas sobrevivam ao teste do tempo. Foi por isso que o Modelo em Cascata atravessou algumas décadas oferecendo aos engenheiros de software um modo razoavelmente seguro e efetivo de criar seus produtos, em substituição ao desenvolvimento especulativo e caótico praticado antes da estruturação de metodologias de desenvolvimento de software.



Anhanguera

Mas, se o Modelo em Cascata é realmente bom, talvez o termo “razoavelmente” tenha sido mal colocado na sentença, não é mesmo? Bem, nesta seção teremos oportunidade de entender alguns elementos desse modelo que justificaram a mudança de paradigma iniciada no ano de 2001 e vivenciada até hoje. O Ciclo de Vida Tradicional – outra forma como o Modelo em Cascata é conhecido – apresenta falhas de concepção que o acabaram tornando obsoleto diante das demandas de tempo e agilidade atuais de produção de software.



Anhanguera

Mas, afinal, quais são os problemas da metodologia tradicional? Qual o motivo da dificuldade dela em acompanhar as inevitáveis mudanças de requisitos de um produto durante seu ciclo de desenvolvimento? Por que o cliente tem dificuldade em reconhecer valor no que está sendo desenvolvido? Mais do que responder a essas questões, esta seção tem a intenção de apontar soluções. Antes de abordarmos diretamente as Metodologias Ágeis, convém tratarmos ainda de alguns aspectos do processo tradicional de desenvolvimento para que as comparações sejam feitas de forma adequada.



Anhanguera

Conforme estudamos na primeira seção, o processo tradicional de desenvolvimento baseia-se na construção linear do sistema, seguindo uma sequência definida de fases, com a particularidade de uma etapa do processo utilizar o resultado obtido pela etapa anterior para criar seu artefato (WAZLAWICK, 2013). Há também a possibilidade de que o fluxo retorne para etapas anteriores sem havendo necessidade de ajustes. Essa construção linear da metodologia baseia-se na ideia de que todas as coisas podem ser construídas como se estivessem em uma linha de montagem: a matéria-prima é inserida na linha de produção e, após algumas etapas determinadas, o produto final está pronto.



A Figura 1.2 ilustra a ideia de utilização dos princípios de uma linha de produção no processo de desenvolvimento de software. Observe que na primeira etapa os requisitos do produto de software representam as matérias-primas. Depois, como em uma linha de montagem tradicional, os especialistas em cada área do processo atuam na linha de montagem até que um produto de software acabado esteja disponível.





Anhanguera

É essa fundamentação nas formas tradicionais de fabricação que nos permite atribuir ao Modelo em Cascata três características bem particulares segundo Teles(2006): o determinismo, a especialização e o foco na execução. Para explicar esses conceitos, usaremos a analogia com o processo de montagem de veículos.

Vejam os:

Determinismo: materiais alimentam um processo de fabricação e, ao final da linha de produção, temos um automóvel terminado. As alterações pelas quais os materiais passam são determinísticas e devem sempre gerar um resultado conhecido. Um processo assim estruturado tende a gerar segurança, redução de tempo e de custo.



Anhanguera

Especialização: o processo tradicional de manufatura divide a montagem desse carro em atividades especializadas, desenvolvidas por trabalhadores igualmente especializados. Assim, numa linha de montagem automotiva, haverá a etapa de soldagem do chassi, de colocação do motor, da montagem do interior e assim por diante, todas elas executadas por especialistas em cada função.

Foco na execução: se as transformações na linha de montagem já estão determinadas e se cada etapa será executada por especialistas, então não há muito em que pensar. Basta executar.



Anhanguera

Se as metodologias tradicionais de desenvolvimento se baseiam em modelos de construção que não se adequam especificamente ao produto que desejam entregar, as metodologias ágeis nasceram ajustadas para a criação de software. Os métodos ágeis enfatizam menos as definições de atividades e mais os fatores humanos inerentes ao desenvolvimento de programas de computador (WAZLAWICK, 2013). Eles são, portanto, claramente mais adequados à natureza do trabalho de profissionais de TI, já que preveem a necessidade de sucessivas revisões na obra. Atividades intelectuais não são executadas de forma linear e não são determinísticas.



Anhanguera

Durante o processo de criação de um software, é mais do que natural que os requisitos sejam revistos, decisões sejam alteradas e detalhes sejam resgatados. Afinal, ao explicar pela primeira vez as funcionalidades que deseja para o produto, o cliente ainda não as conhece por completo e a visão global do que necessita ainda não está formada. Que tal um procedimento que dê a ele oportunidade de aprender e de mudar de ideia ao longo do desenvolvimento? Você certamente não notou essa atenção com o cliente ao longo do conteúdo do Modelo em Cascata.



Anhanguera

No ano de 2001, um documento contendo a declaração dos doze princípios que fundamentam o desenvolvimento ágil foi divulgado por um grupo de profissionais de TI. Através desse documento, chamado Manifesto Ágil, seus autores declaravam que indivíduos e interações importavam mais do que processos e ferramentas; que softwares em funcionamento importam mais do que documentação; que colaboração com o cliente importa mais do que negociação de contratos e que responder a mudanças é melhor do que seguir um plano (BECK et al., 2001). Você pode conferir a íntegra do manifesto e conhecer seus autores na página Agile Manifesto, na internet (BECK et al., 2001).

<https://agilemanifesto.org/iso/ptbr/manifesto.html>



Anhanguera

VISÃO GERAL DO EXTREME PROGRAMMING(XP)

O XP é uma metodologia adequada para empreitadas em que os requisitos mudam com certa constância, além de ser ajustado para equipes pequenas e para o desenvolvimento de programas orientados a objetos (TELES, 2006). Ao contrário das metodologias tradicionais, ele é indicado também para ocasiões em que se desejam partes executáveis do programa logo no início do desenvolvimento e que ganhem novas funcionalidades conforme o projeto avança.



Anhanguera

Embora a especialização não seja estimulada nas metodologias ágeis, ainda assim existe a necessidade de se estabelecer funções para os participantes do projeto. Uma típica equipe de trabalho no XP tem a seguinte configuração (TELLES, 2004):

Gerente do projeto: trata-se do responsável pelos assuntos administrativos do projeto e do relacionamento com o cliente. Duas de suas funções mais importantes incluem o estabelecimento de contato entre a equipe e o cliente e o cuidado para que a equipe fique livre de pressões externas e se dedique integralmente ao desenvolvimento.



Anhanguera

Coach: este é o nome do responsável técnico pelo projeto, que deve ser tecnicamente bem preparado e experiente. Se, por exemplo, uma nova tecnologia fica disponível no mercado, é função dele sugerir seu uso no produto em desenvolvimento.

Analista de teste: ajuda o cliente a escrever os testes de aceitação e fornece feedback para a equipe interna de modo que as correções no sistema possam ser feitas. Ao contrário do que é feito nas metodologias tradicionais, no desenvolvimento ágil em geral (e no XP em particular), o cliente participa ativamente dos testes no produto.



Anhanguera

Redator técnico: ajuda a equipe de desenvolvimento a documentar o sistema, permitindo que os desenvolvedores estejam plenamente focados na construção do programa propriamente dito. À propósito, a metodologia recomenda que o código deve ser claro o suficiente para permitir uma documentação mínima do sistema.

Desenvolvedor: realiza análise, ajuda a criar o projeto e codifica o sistema. No XP, não há divisão entre as especialidades de engenheiro de requisitos, projetista e o desenvolvedor propriamente dito. No modelo tradicional, o desenvolvedor apenas programa, via de regra.



Anhanguera

Para atingir seus objetivos, o Extreme Programming apoia-se em quatro pilares, comumente chamados de valores.

O primeiro deles é o **feedback**, cuja aplicação pretende alcançar a troca de informações entre cliente e equipe. Assim, quando o cliente aprende com o sistema que utiliza e reavalia suas necessidades, ele gera feedback para sua equipe de desenvolvimento.

O segundo valor é chamado de **comunicação** e pretende estabelecer contato proveitoso entre equipe e cliente, evitando que os desenvolvedores realizem o trabalho de forma especulativa.



Anhanguera

O terceiro valor é a **simplicidade**, cuja formulação serve para orientar a equipe a desenvolver apenas o que for suficiente para atender a necessidade do cliente, sem “sobrecarregar” o sistema com funções quase sempre inúteis.

Por fim, o último valor estabelece a **coragem** como um dos pilares do XP, pois ela será necessária no impulsionamento da equipe para manter sempre o cliente presente, para propor melhorias no que já foi testado e colocado em funcionamento e, de modo geral, para sempre levar adiante as práticas da metodologia.



Anhanguera

Planejamento: essa atividade, comumente identificada como jogo do planejamento, tem início com o esclarecimento de requisitos do produto e é executada de modo bem diferente daquele proposto pelo modelo tradicional: aqui, o cliente escreve – de próprio punho – as funcionalidades do produto em uma ficha. A essa ficha dá-se o nome de História e cada uma delas é avaliada pela equipe em termos de custo e tempo de execução. A qualquer momento, o cliente pode escrever novas histórias.

Projeto: representações complexas da solução certamente não são características desta tarefa, muito menos a implementação de funcionalidades extras no produto não solicitadas pelo cliente. A intenção é que o projeto sirva como um guia de implementação de cada história e que seja compreendido também pelo cliente. Caso a equipe se depare com um problema cuja representação seja muito difícil, a metodologia recomenda que se construa um protótipo executável dessa parte do projeto, reduzindo, assim, o risco de se construir uma versão final equivocada daquela história.



Anhanguera

Codificação: mesmo depois de desenvolvidas as histórias e de ter sido feito o trabalho de projeto, a codificação ainda não é iniciada. Ao invés de procurarem codificar uma versão final do produto, os desenvolvedores criarão testes de unidade para cada uma das histórias e só então, após feita e validada essa atividade, o desenvolvimento será orientado a uma versão completa e final.

Testes: os testes de unidade criados durante a codificação devem ser aptos à automatização, de modo a permitir que sejam feitos rapidamente e repetidos quando necessário. O conjunto de testes de unidade podem ser usados a qualquer momento para testes de integração e de validação do produto.

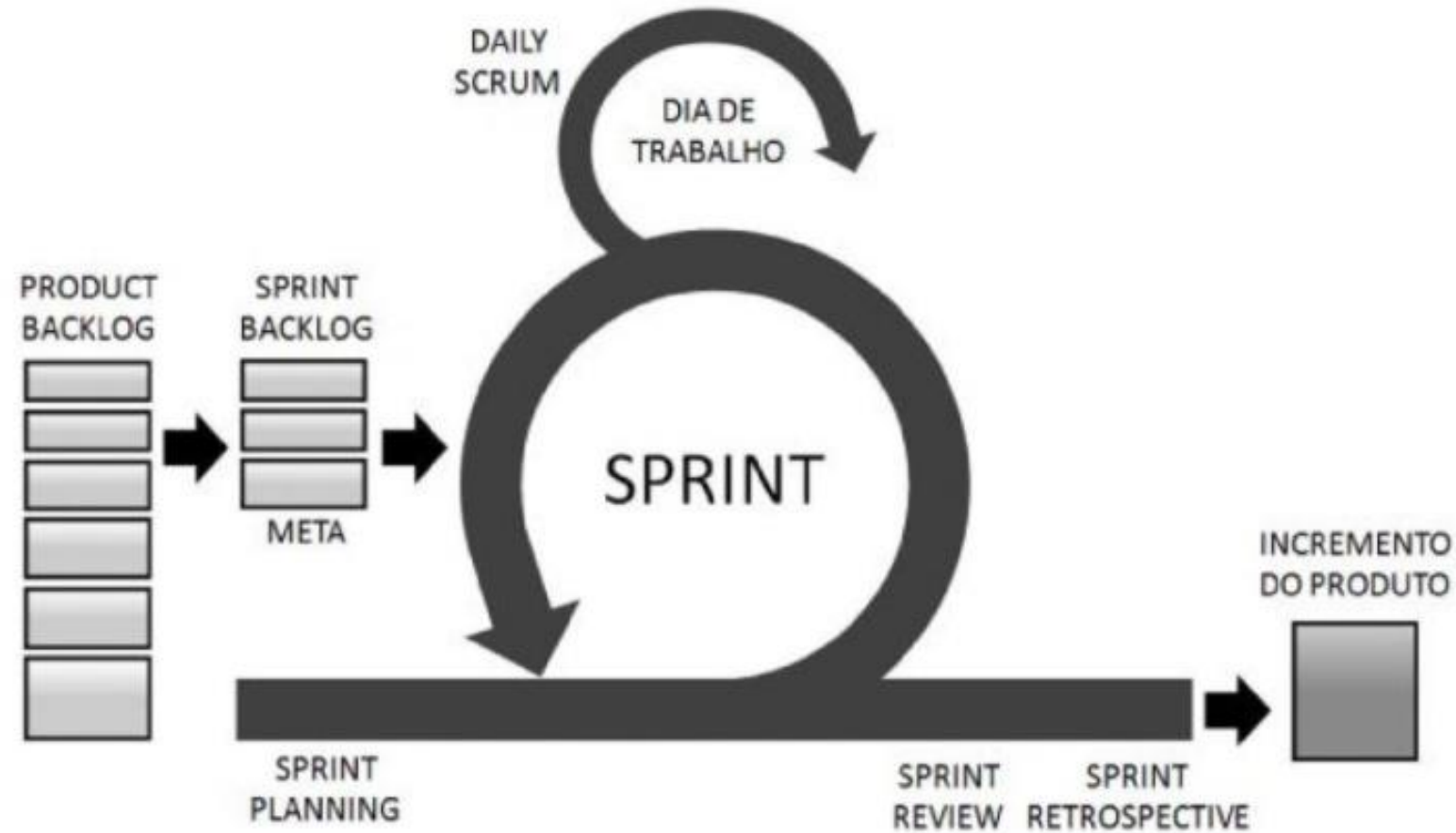


VISÃO GERAL DO SCRUM

Outro método, também bastante adaptado às demandas atuais de tempo e qualidade, tem sido alternativa para as organizações desenvolvedoras. Trata-se do Scrum, um modelo ágil para a gestão de projetos de software, o qual tem, na reunião regular dos seus desenvolvedores para criação de funcionalidades específicas, sua prática mais destacada. Suas práticas guardam semelhança com as próprias do XP, mas possuem nomes e graus de importância diferentes nos dois contextos. Iniciaremos nossa abordagem com o ciclo tradicional do Scrum, ilustrado na Figura 1.3.



Anhanguera





Cada um dos elementos do ciclo é abordado na sequência:

Product Backlog: trata-se da lista que contém todas as funcionalidades desejadas para o produto. O Scrum defende que tal lista não precisa estar completa logo na primeira vez em que é feita. “Pode-se iniciar com as funcionalidades mais evidentes [...] para depois, à medida que o projeto avançar, tratar novas funcionalidades que forem sendo descobertas” (WAZLAVICK, 2013, p.56).

Sprint Backlog: lista de tarefas que a equipe deverá executar naquele Sprint. Tais tarefas são selecionadas do Product Backlog, com base nas prioridades definidas pelo Product Owner.



Anhanguera

Sprint: o Scrum divide o processo de efetiva construção do software em ciclos regulares, que variam de duas a quatro semanas (ou em períodos maiores, a depender da complexidade das tarefas). Trata-se do momento em que a equipe se compromete a desenvolver as funcionalidades previamente definidas e colocadas no Sprint Backlog. Se alguma funcionalidade nova for descoberta, ela deverá ser tratada na Sprint seguinte. Cabe ao Product Owner manter o Sprint Backlog atualizado, apontando as tarefas já concluídas e aquelas a serem concluídas.



Anhanguera

Embora o modelo Scrum defenda que as equipes sejam auto organizadas, ainda assim apresenta três perfis profissionais de relevância:

Scrum Master: trata-se de um facilitador do projeto, um agente com amplo conhecimento do modelo e que preza pela sua manutenção durante todas as etapas do projeto. Deve atuar como moderador ao evitar que a equipe assuma tarefas além da sua capacidade de executá-las.



Anhanguera

Product Owner: é a pessoa responsável pelo projeto propriamente dito. Ele tem a missão de indicar os requisitos mais importantes a serem tratados nos sprints.

Scrum Team: é a equipe de desenvolvimento, composta normalmente por um grupo de seis a dez pessoas. A exemplo do Extreme Programming, não há divisão entre programador, analista e projetista (WAZLAVICK, 2013).



Anhanguera

A ênfase no gerenciamento tradicional de projetos diz respeito à condução de um planejamento detalhado do projeto com ênfase na fixação do escopo, do custo e do cronograma e no gerenciamento desses parâmetros. O gerenciamento tradicional de projetos pode às vezes levar a uma situação em que o plano foi bem-sucedido, mas o cliente não está satisfeito.

Em relação às características dos modelos ágeis, analise as afirmações que seguem:

- I. Estimulam o desenvolvimento incremental e com intervalos curtos de feedback entre equipe e cliente.
- II. Foram criados com base nas ideias da produção linear, desenvolvidas para bens de consumo comuns.
- III. Apresentam determinismo e especialização de funções como marcas próprias.
- IV. São mais bem adaptadas às mudanças de requisitos durante o projeto do que os modelos tradicionais.

Considerando o contexto apresentado, é correto o que se afirma em:

- a. I, II, III e IV.
- b. I e IV apenas.
- c. I, II e IV apenas.
- d. III e IV apenas.
- e. I, III e IV apenas.



Anhanguera

O Scrum é um dos métodos Agile mais populares. É uma estrutura adaptativa, iterativa, rápida, flexível e eficaz projetada para entregar valor significativo de forma rápida e durante todo o projeto. O Scrum garante transparência na comunicação e cria um ambiente de responsabilidade coletiva e de progresso contínuo.

No contexto do modelo Scrum, preencha as lacunas a seguir.

- I. O _____ contém as funcionalidades que compõem o sistema.
- II. O _____ responde pelo projeto e tem a responsabilidade de indicar os itens que compõem a lista de requisitos.
- III. O _____ é a equipe de desenvolvimento. Nela, não existe necessariamente divisão funcional em papéis tradicionais.

Assinale a alternativa cujos termos completam corretamente as lacunas nas frases anteriores.

- a. Product Backlog, Product Owner, Scrum Team.
- b. Rol de requisitos, Scrum Team, conjunto de desenvolvedores.
- c. Sprint Backlog, sponsor, Scrum Team.
- d. Levantamento de requisitos, sponsor, Scrum Master.
- e. Product Backlog, Scrum Team, Scrum Master.