The background of the book cover features a photograph of four people in a modern office or study environment. A woman with curly hair is leaning over a table, pointing at a tablet screen. Two other women are seated at the table, looking at papers and a laptop. A man is partially visible on the right. The scene is lit with warm, golden light.

SISTEMAS DE INFORMAÇÕES GERENCIAIS

Cristiano Barbosa

Ciclos de desenvolvimento de sistemas, metodologia ágil e scrum

Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Analisar as diferenças entre as principais metodologias de desenvolvimento de software e suas etapas.
- Diferenciar os papéis, os artefatos e as cerimônias da metodologia ágil para o desenvolvimento de software e projetos.
- Construir histórias para solicitar funcionalidades para o desenvolvimento de produtos.

Introdução

O mundo evoluiu de forma intensa nos últimos anos, em grande parte, devido aos sistemas de computador que automatizaram as rotinas e, em muitos casos, substituíram completamente a interação humana.

As softhouses, empresas que desenvolvem sistemas, cresceram e, hoje, fazem parte de uma fatia do mercado sólida e promissora. Elas são responsáveis pela criação e pelo desenvolvimento dos softwares e, para isso, utilizam várias metodologias.

Neste capítulo, você entenderá as diferenças entre as principais metodologias de desenvolvimento de software e suas etapas, compreenderá os diferentes papéis, artefatos e cerimônias da sua metodologia ágil e aprenderá a construir histórias para solicitar funcionalidades para o desenvolvimento de produtos.

Metodologias de desenvolvimento de software

Para se desenvolver um software, é preciso muito mais que apenas linhas de código, pois deve-se ter uma metodologia de trabalho.

Segundo o dicionário Webster's (1998), metodologia é um conjunto de métodos, regras e postulados empregados por uma disciplina: um procedimento particular ou conjuntos de procedimentos.

A metodologia de desenvolvimento de software possui a missão de definir de forma clara e objetiva os papéis de todos os envolvidos, por exemplo, usuários, desenvolvedores e administradores, e determinar como, quando e onde ele será realizado. Ela serve como uma ferramenta que harmoniza e coordena todas as áreas envolvidas no processo de desenvolvimento.

Modelo cascata ou sequencial linear

O modelo cascata ou sequencial linear é realizado sistemática e sequencialmente, em que, primeiro, é feito o levantamento de requisitos ou necessidades e, depois, a fase de planejamento na qual são definidas as estimativas, o cronograma e o acompanhamento. Em seguida, realiza-se a modelagem, com a análise e o projeto, depois a construção na qual são desenvolvidos o código e os testes. Logo após, são realizadas a implantação e entrega, o suporte e feedback do sistema terminado. Você verá na Figura 1 as etapas desse modelo.

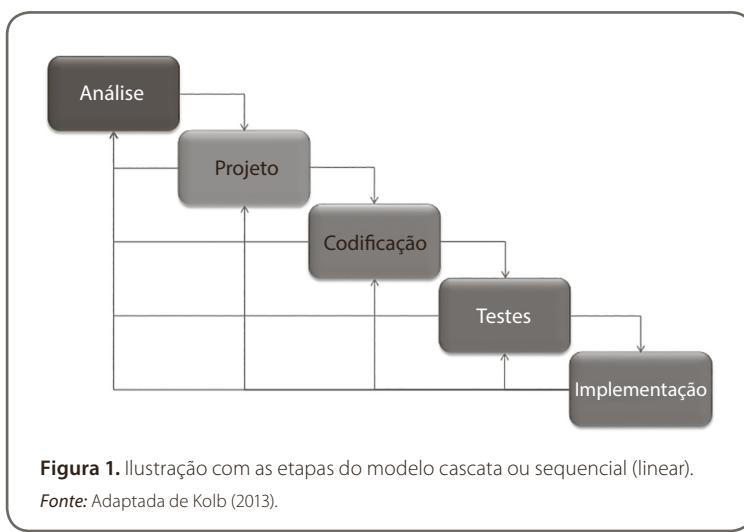


Figura 1. Ilustração com as etapas do modelo cascata ou sequencial (linear).

Fonte: Adaptada de Kolb (2013).



Fique atento

As metodologias de desenvolvimento servem para não tornar a tarefa, complexa por natureza, um verdadeiro caos.

Desenvolvimento por prototipação

A prototipação é um processo que objetiva o esclarecimento de todos os requisitos, apresentando conceitos e funcionalidades do sistema e possibilitando, assim, a proposta de uma solução adequada para o problema. Os protótipos podem ser divididos basicamente em três categorias, como você verá a seguir.

- *Wireframes & rascunhos*: são de baixa fidelidade e permitem alterações rápidas. Eles não mostram detalhes visuais ou interações, mas permitem validar regras de negócios de maneira muito eficiente.
- Protótipos visuais: são criados com softwares de edição gráfica. Embora possuam maior apelo visual, não possuem interações e demandam muito mais tempo para ajustes.
- Protótipos interativos: são muito completos, pois além da parte visual, englobam uma série de detalhes e efeitos de interação. Permitem também que os desenvolvedores identifiquem novos requisitos, oportunidades e futuros problemas. Embora tragam inúmeras vantagens, demandam desenvolvedores experientes e demoram mais tempo para serem criados.

Metodologia ágil ou scrum

Scrum é uma metodologia ágil para desenvolvimento e planejamento de projetos de sistemas, na qual esses projetos são divididos em ciclos, geralmente mensais, chamados de *sprints*. O *sprint* representa um *time box* dentro do qual uma série de rotinas devem ser executadas.

Todas as funcionalidades a serem desenvolvidas são mantidas em uma lista chamada *product backlog*. No início de cada *sprint*, é realizado um *sprint planning meeting*, uma reunião na qual o *product owner* (proprietário do projeto) prioriza os itens do *product backlog* e a equipe escolhe as atividades

que irá desenvolver, assim sendo, as tarefas escolhidas são transferidas do *product backlog* para o *sprint backlog*.

A cada dia de *sprint*, a equipe faz uma breve reunião chamada *daily scrum*, cujo objetivo é discutir e inteirar a equipe sobre o que foi desenvolvido no dia anterior, identificando impedimentos e priorizando o trabalho do dia.

Ao final de um *sprint*, os desenvolvedores apresentam as funcionalidades implementadas em uma *sprint review meeting*. Finalmente, é realizada uma *sprint retrospective*, e a equipe começa o planejamento do próximo *sprint*.

Papéis

O *product owner* é responsável por maximizar o valor de retorno do produto para o cliente e fornecer o trabalho ao *development team*. Já o *development team* são os profissionais, de três a cinco, que entregarão uma versão incrementada do produto feito, ao final de cada iteração. Por fim, o *scrum master* é responsável por assegurar que o *scrum* seja entendido e disseminado, resolver impedimentos e ajudar o *product owner* e o *development team*.

Artefatos

Os artefatos são documentos responsáveis pela transparência sobre os itens que precisam ser feitos e entregues de um produto ou incremento, e existem quatro deles.

- *Product backlog*: engloba todos os requisitos que precisam ser implementados, seja de negócio (*user stories*), melhorias, questões técnicas ou erros. Esse artefato é controlado pelo *product owner* e ordenado de acordo com o valor de negócio para o cliente.
- *Sprint backlog*: é uma lista de itens que define o objetivo da *sprint*. Retirados do *product backlog*, esses itens são os que estão na frente. Os itens que entram no *sprint backlog* são os de maior valor de negócio apresentado pelo *product owner*.
- *Definition of done*: é um documento que explica quais serão as definições de pronto para as *user stories*.
- *Burndown*: é um gráfico que mostra o andamento do desenvolvimento do produto. Pode ser usado para monitorar o andamento tanto do *product backlog* quanto do *sprint backlog*.

backlog como do *sprint backlog*, indicando se ambos serão completados de acordo com o planejamento.

Na Figura 2, você acompanhará as etapas do modelo ágil ou *scrum*.

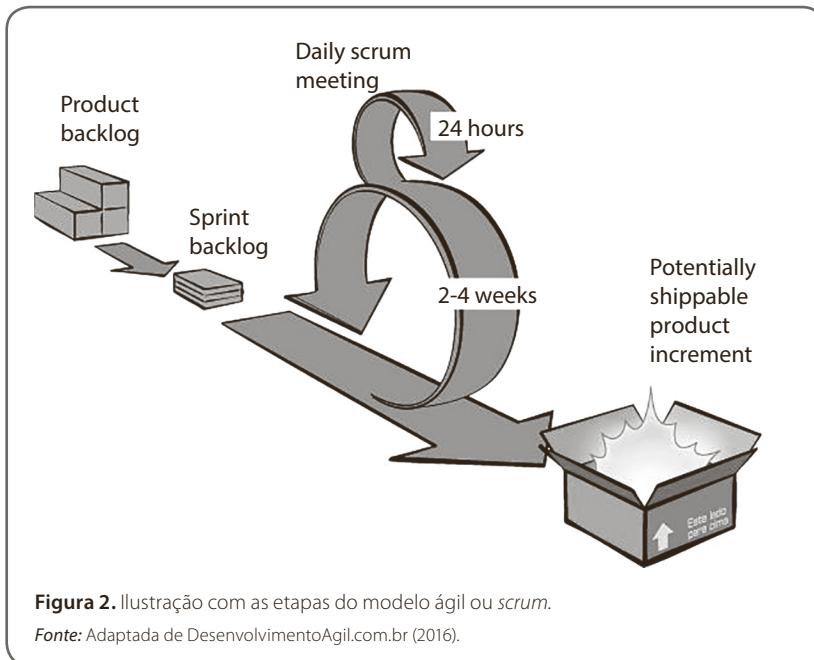


Figura 2. Ilustração com as etapas do modelo ágil ou *scrum*.

Fonte: Adaptada de DesenvolvimentoAgil.com.br (2016).

Cerimônias

As cerimônias são alguns eventos descritos no *scrum*, cada qual com o seu objetivo, a fim de evitar reuniões desnecessárias. Elas devem ter um tempo máximo de duração, chamado *time box*, permitindo a oportunidade de praticar a inspeção e adaptação.

Existem quatro cerimônias: *sprint planning*; *daily scrum*; *sprint review meeting*; e *sprint retrospective*. Esses eventos são o princípio que diz “o método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma conversa cara a cara”.

Você verá, detalhadamente, na Figura 3 as cerimônias e os papéis do *scrum*.

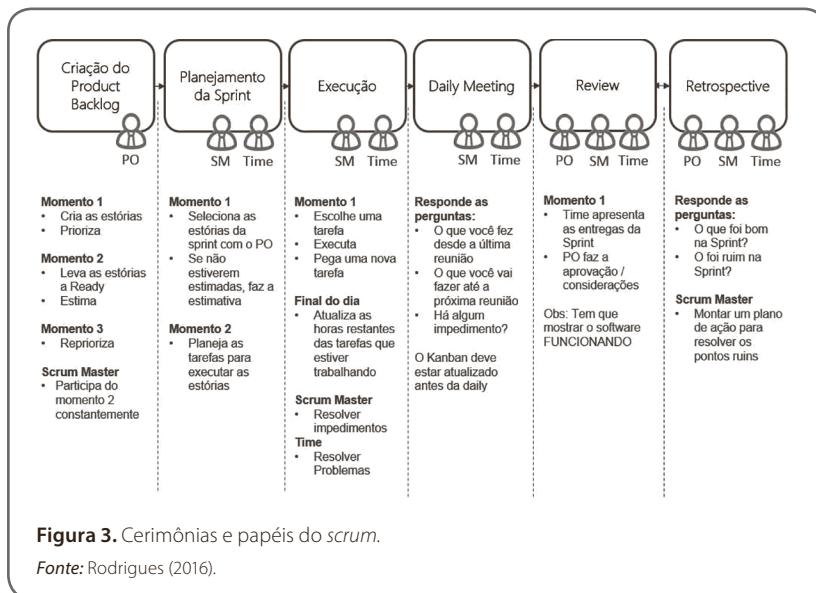


Figura 3. Cerimônias e papéis do *scrum*.

Fonte: Rodrigues (2016).

História de usuário

A história de usuário é uma curta e simples descrição da necessidade do cliente, geralmente contada a partir da perspectiva de quem precisa da nova necessidade. Ela deve explicar bem para quem, o que e por que está sendo criada. Quanto mais explicativa ela for, melhor os desenvolvedores poderão entender e construir softwares com menos dificuldade e mais qualidade.

INVEST

Para garantir a qualidade na criação de uma história de usuário, foi criado o acrônimo INVEST, o qual você verá o significado a seguir.

I de *independent* (independente): a história de usuário precisa ser independente e não depender de outras histórias.

N de *negotiable* (negociável): a história de usuário não é apenas uma narrativa com as características que o *product owner* espera, mas também um ponto de partida para uma discussão a fim de que a equipe proponha melhorias.

V de *valuable* (de grande valor): a história de usuário precisa possuir valor, se não for descrito o valor que o cliente terá com ela, não terá muita serventia.

E de *estimable* (estimável): a história de usuário precisa conter informações suficientes para permitir que os desenvolvedores façam uma estimativa, caso contrário, ela não poderá ser iniciada.

S de *small* (pequena): a história de usuário necessita ser pequena, pois deve caber na duração de um *sprint*.

T de *testable* (testável): a história de usuário precisa ser testável, pois caso não possua informações suficientes para testes, ela não poderá ser colocada no *backlog*.



Referências

KOLB, J. J. O modelo em cascata. *Compartilhando*, 7 nov. 2013. Disponível em <<http://jkolb.com.br/o-modelo-em-cascata/>>. Acesso em: 9 mar. 2018.

METHODOLOGY. In: WEBSTER'S ninth new collegiate dictionary. Springfield (MA), Merriam-Webster, 1998.

RODRIGUES, E. Cerimônias e papéis do Scrum: quem faz o que e quando. Gestão de projetos na prática, 30 set. 2016. Disponível em: <<https://www.elirodrigues.com/2016/09/30/cerimonias-e-papeis-do-scrum-quem-faz-o-que-e-quando/>>. Acesso em: 9 mar. 2018.

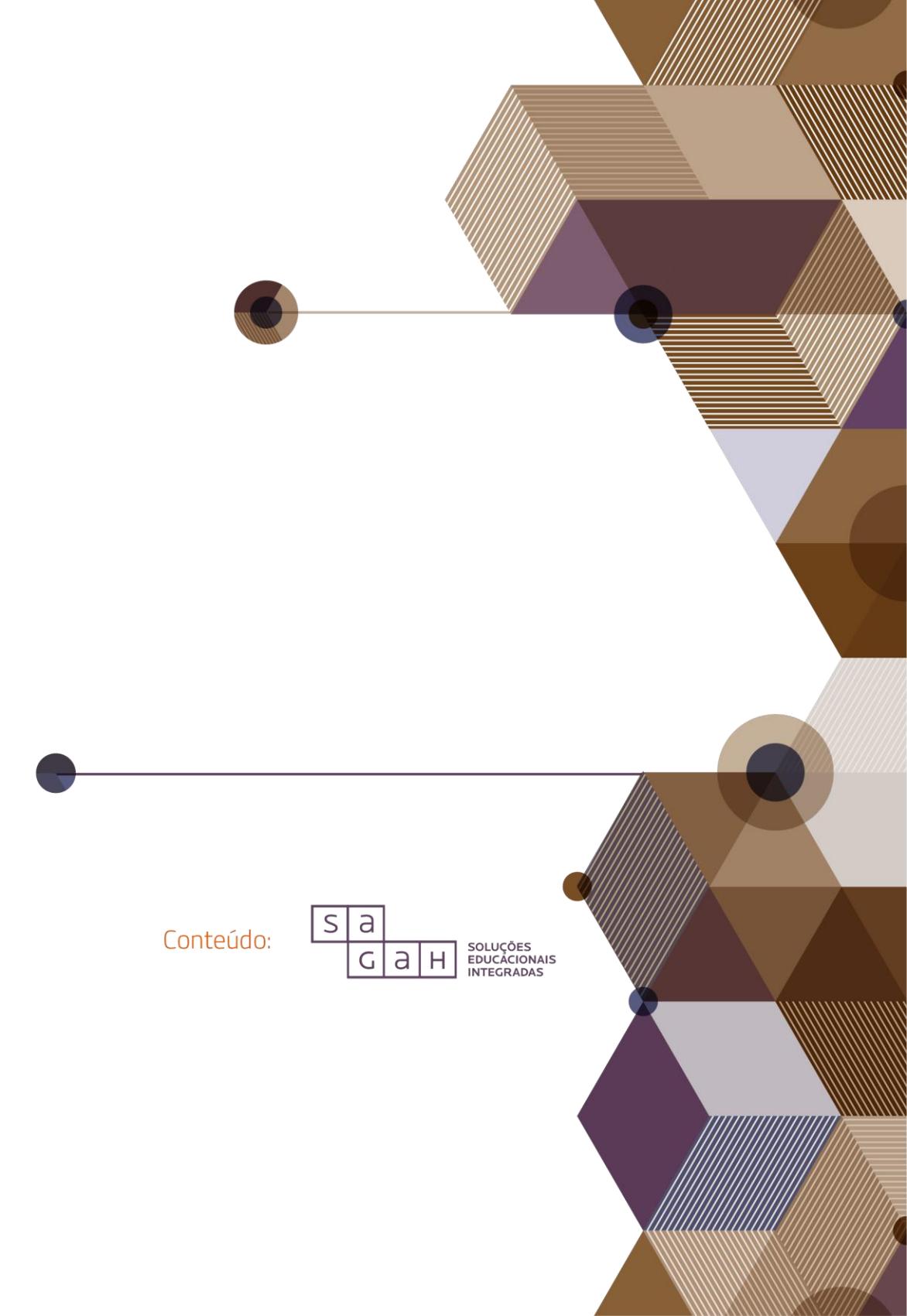
SCRUM. *DesenvolvimentoAgil.com.br*, 25 maio 2016. Disponível em: <<http://www.desenvolvimentoagil.com.br/scrum/>>. Acesso em: 22 fev. 2018.

Leituras recomendadas

BERNARDO, K. Estória de usuário: você saberia contar? *Cultura Ágil*, 8 dez. 2014. Disponível em <<https://www.culturaagil.com.br/estoria-de-usuario-voce-saberia-contar/>>. Acesso em: 23 fev. 2018.

MORALLES, F. Metodologia ágil e framework Scrum. *Matera*, 11 mar. 2013. Disponível em: <<http://matera.com.br/2013/03/11/metodologia-agil-framework-scrum-2/>>. Acesso em: 23 fev. 2018.

Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.



A large, abstract graphic in the background consists of a grid of hexagons in various shades of brown, tan, purple, and blue. Some hexagons have fine diagonal lines running from top-left to bottom-right, while others are solid or have different patterns. Small, dark circular dots are scattered across the white space between the hexagons, some connected by thin lines.

Conteúdo:



SOLUÇÕES
EDUCACIONAIS
INTEGRADAS